Research Article

Usability Evaluation Approach of Educational Resources Software Using Mixed Intelligent Optimization

Jiaze Sun

School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an 710121, China

Correspondence should be addressed to Jiaze Sun; sunjiaze@126.com

Aiming at the problems of strong subjectivity and uncertain fuzziness of attribute weights in the software usability evaluation approach, an evaluation approach based on mixed intelligent optimization was proposed, which combines subjective and objective methods to measure software usability for educational resources software. Firstly, the usability evaluation index system of educational resources software was established, and the basic probability assignment was generated by the interval method from the historical sample data. Then the weight optimization problem was adapted to the smooth optimization problem by the maximum entropy function method, and the hybrid social cognitive optimization (HSCO) algorithm was introduced to solve the optimal weights of evidence. Finally, the software usability level was fused by DS evidence theory. The experimental results show that the educational resources software usability evaluation approach can objectively and truly reflect the usability of the software. It provides an efficient way to evaluate the usability of the software.

1. Introduction

In the era of user-centered product design, a good user experience is the direct way to keep the user's viscosity, for the usability is an important software feature. Software usability evaluation is an important part in the process of software quality assurance. Therefore, usability as product quality evaluation indicator gradually becomes a key factor to be considered in the design of software development.

Software usability evaluation is an important part of software credibility evaluation. Software credibility evaluation is mainly based on the related standard of software quality by the establishment of evaluation model and tool. In the past, the usability of software systems was evaluated subjectively and the process was not well defined, and there was no mature evaluation method that is accepted by most people [1]. Although usability evaluation and analysis methods and methodologies were being developed, software usability evaluation method research is still in its early infancy. At the same time, the usability of software is closely related to the specific applications and users, and software usability evaluation research has all kinds of different methods in different software systems [2].

With the rapid development of the mobile applications, usability of mobile applications is in the focus of the software usability researches. Harrison et al. [3] introduced the PACMAD (People at the Centre of Mobile Application Development) usability mode to solve the limitations of existing usability models when they are applied to mobile applications. Field studies and lab experiments are two methodologies most often applied by researchers; Kaikkonen et al. [4] found that field testing is worthwhile when combining usability tests with a field pilot or contextual study. Chen [5] proposed evaluation method for mobile B2C interface program which is based on adaptation touch interaction and natural gesture interaction and includes usability evaluation index system and analytic hierarchy process (AHP) method for optimizing the weights. Zhang et al. [6] proposed quality indicators system of software products and presented a simple usability evaluation method from the perspective of users and experts. Liu et al. [7] proposed a usability assessment method based on Usability Maturity Model for organizational human-centeredness at some software enterprises, which gives a feasible strategy for introducing usability engineering in the industry. Guo et al. [8] presented a usability evaluation model for application

software based on user emotion; however, the evaluation process is entirely subjective assessment that was not easily convincing.

Van Nuland et al. [9] found that usability testing, particularly within the anatomical sciences teaching online, should be employed during the design and development phases, as well as during its delivery. Zhao et al. [10] used fuzzy analytic hierarchy process to quantitatively evaluate software usability, but the result of this method was still greater uncertainty. Li et al. [11] constructed a weighted sum of software usability evaluation method, but the weight of the distribution was not very objective. Li et al. [12] proposed a method of using the AHP method to determine the weight of the weighted DS evidence theory; the initial weight and the initial probability distribution were given by the experts.

Mainly based on subjective evaluation by experts and users, these traditional methods adopt the method of analytic hierarchy process (AHP) and the weighted average to evaluate the software usability. Software usability evaluation problem is ultimately a multiple attribute decision-making problem, which depends on how to establish the availability index system, how to assign attribute weights, and how to fuse the multiple attributes.

DS (Dempster-Shafer) evidence theory uses quantitative and qualitative data to establish the assessment model with great advantage under the unified recognition framework, which helps obtain more accurate results for multiple attribute decision-making (MADM) problems [13]. Literature [14] proposed a hybrid approach to develop the partner evaluation model for tourism partner selection problem, by applying the DS evidence theory and satisfactory principle as alternative framework. Literature [15] proposed a MADM method based on evidential reasoning approach with unknown attribute weights in intuitionistic fuzzy environment. When DS evidence theory is applied for MADM problems, there are two main key problems [16]: basic probability assignment (BPA) generation and attribute weight optimization problems.

At present, there are two kinds of BPA generating methods: expert subjective set method and automatically generating method according to the historical knowledge. Multiple experts independent set methods always have high conflicts. Literature [17] proposed Dynamic Belief Fusion (DBF) method to assign probabilities to individual detectors, which optimally fused information from all detectors. Literature [18] automatically generated BPA which used the history sample data to identify results. Literature [19] generated random numbers based on set theory and presented evidence fusion strategy based on distance. So far, basic probability assignment is no good way to generate. Traditional BPA generation needs complete information to support. But the usability evaluation indexes are different, since software usability metric in different software varies greatly. And because usability testing data are poor and available empirical knowledge is scarce, usability evaluation is very subjective and uncertain. Interval number theory [20] only requires the upper and lower bounds of the information scope, so it is very suitable for the application field in which characteristic information is poor, fuzzy, and imprecise. In literature [21],

the basic probability assignment (BPA) is generated based on the distance between interval numbers to improve belief Markov chain model.

To reduce the negative impact of single inaccurate attribute and improve accuracy and stability of the determining system, it is very necessary to fuse the property from multiple aspect sources. But the importance of each attribute in judgment system is different, so the attribute fusion should consider the influence of the weight of the different characteristics. To deal with different weights, the weighted synthesis technology widely uses similarity weighted method and the weighted average method [22]. Literature [23] put forward a kind of evidence synthesis method based on practical experience. But, at present, the DS evidence theory rarely discusses optimization weight acquisition method. The traditional weight acquisition methods mainly have expert subjective weights determining method and history statistical method, which are all difficult to obtain the optimal weight value. Aimed at the shortage of the weighted method in determining the weights of evidence theory, literature [24] uses particle swarm algorithm combining historical data value to obtain the optimal weights in the weighted information fusion problem, but the particle swarm optimization (PSO) algorithm is easy to premature and cannot guarantee the global convergence. Literature [25] proposes a weighted classifier combination method to minimize the distances between fusion results obtained by Dempster's rule to enhance the classification accuracy.

We apply DS evidence theory to software usability evaluation and establish the software usability evaluation index system in view of the education resources, and a software usability evaluation method based on DS evidence reasoning is proposed. In the new method, the basic probability assignment is produced by using the method of interval number in combination with historical data; the weight optimization problem is transmitted to a smooth optimization problem through maximum entropy function method; the hybrid social cognitive optimization (HSCO) algorithm [26] is adopted to solve the optimal weights problem. As a multiple attribute decision-making problem, software usability evaluation is eventually better solved.

The primary contributions of the paper are as follows:

(1) A software usability evaluation index system of the educational resources software based on the ISO/IEC25000 series standards is introduced, which fully considers the characteristics of the educational resources software and the characteristics of the target user. Usability evaluation index system is flexible and is more in line with the needs of educational resources software usability evaluation.

(2) The basic probability assignment is generated through the interval number theory from the history of the sample data. Interval number theory is very suitable for poor evaluation information and fuzzy imprecise characteristics.

(3) To obtain better fusion effect, the hybrid social cognitive optimization algorithm is used to optimize

the different weights of evidence, in which global convergence is guaranteed.

2. Dempster-Shafer Theory

2.1. Basic Conceptions. A belief structure as introduced by Shafer provides an approach to represent nonspecific forms of uncertainty. Formally DS belief structure on space X consists of a collection of nonempty crisp subsets of X called focal elements: $A_1, \ldots, A_q$. This represents the value of a variable V whose domain is called the frame of discernment [13].

Let Ω stand for a domain set for every possible value of X, and every component in Ω is incompatible. And then we call Ω the differentiation frame of X; let 2^Ω stand for the power set of Ω. Let ϕ denote the empty set.

Definition 1. A basic probability assignment (BPA) is a function $M : 2^\Omega \to [0, 1]$, which satisfies the following conditions: (1) $\sum_{A \subseteq \Omega} M(A) = 1$ and (2) $M(\Omega) = 0$; $M(A)$ is called basic probability number, which represents the proportion of all relevant and available evidence that supports the claim that a particular element of Ω belongs to the set A but to no particular subset of A.

Definition 2 (Dempster combination rules). According to Dempster's orthogonal rule of evidence combination [13], for k basic probability assignment functions $M_1, M_2, \ldots, M_k$ in the same frame of discernment Ω, the combination function of $M_1, M_2, \ldots, M_k$ is $M = M_1 \oplus M_2 \oplus \cdots \oplus M_k$.

$$M(A)$$
$$= \frac{\sum_{A_1 \cap A_2 \cap \cdots \cap A_k = A} M_1(A_1) M_2(A_2) \cdots M_k(A_k)}{1 - \sum_{A_1 \cap A_2 \cap \cdots \cap A_k = \Phi} M_1(A_1) M_2(A_2) \cdots M_k(A_k)}, \quad (1)$$

or

$$M(A) = K^{-1} \times \sum_{\cap A_i = A} \prod_{1 < i < n} M_i(A_i); \quad (2)$$

here, $K = \sum_{\cap A_i \neq \Phi} \prod_{1 < i < n} M_i(A_i)$, $\oplus$ indicates that Dempster's combination rule is used, and $A_1, A_2, \ldots, A_k$ are focus elements. $\sum_{A_1 \cap A_2 \cap \cdots \cap A_k = \Phi} M_1(A_1) M_2(A_2) \cdots M_k(A_k)$ is the conflict weight that reflects the information conflict. If the conflict weight is 1, the evidences completely conflict with each other.

2.2. Weighted Transformation. In uncertain information fusion, not all the evidences have the same importance. Some evidences are more important than others. Traditional DS evidence theory does not differentiate the importance of different evidences. To differentiate the importance of different evidences, the weighted value should be processed in the following [27].

Suppose that there are r features $F_1, F_2, \ldots, F_r$. For the sake of simplicity, we assume that all features are independent of one another. Their weighted values are $w_1, w_2, \ldots, w_r$. Let $w_f = \max(w_1, w_2, \ldots, w_r)$.

$$\beta_i = \frac{w_i}{w_f} \quad (i = 1, 2, \ldots, r),$$

then $M_i^{\beta_i}(\Phi) = 0$,

$$M_i^{\beta_i}(\Omega) = \beta_i M(\Omega) + (1 - \beta_i),$$
$$M_i^{\beta_i}(A) = \beta_i M(A),$$
$$\quad (3)$$

$$\forall A \subset 2^\Omega, \ A \neq \Omega, \ A \neq \Phi.$$

In formula (3), M_i are the basic probability values of the evidence Fi; $M_i^{\beta_i}$ denotes the transferred probability assignment function. Through formula (3) transferring, the weight value β_i of the evidence Fi is reflected in the basic probability values of the proposition, which makes the weights of each evidence be transformed equally. Then we can synthesize the basic probability values transformed by DS evidence theory.

3. Hybrid Social Cognitive Optimization (HSCO) Algorithm

By introducing human social intelligence based on social cognitive theory to artificial system, Xie et al. [28] proposed social cognitive optimization (SCO) algorithm in 2002. In SCO optimization procedure, a knowledge library with symbolizing capability consists of many knowledge points and learning agents, which act observational learning via the neighborhood local searching by observing the selected model from tournament selection. Because SCO algorithm fully makes use of the interactions and shares of the entire social swarm, it greatly improves the convergence speed and accuracy of the swarm intelligence algorithm and makes it better than many other well-used intelligent optimization methods, such as Genetic Algorithm (GA), particle swarm optimization (PSO), and Ant Colony Optimization (ACO), in many applications [29].

To improve the global convergence speed and stability of SCO algorithm without increasing the computation, a hybrid social cognitive optimization (HSCO) algorithm, based on elitist strategy and chaotic optimization, was proposed to solve constrained NLPs in literature [26]. Learning agents are partitioned into three groups in proportion: elite learning agents, chaotic learning agents, and common learning agents. The common agents in major proportion work in the search way of traditional SCO. The elite learning agents in a little proportion search via elitist selection to improve the global searching performance. The chaotic learning agents in a little proportion search via chaotic search algorithm to avoid the premature convergence. HSCO algorithm is guaranteed to converge to global optimal solution with probability of one because of the elite learning strategy and the chaotic learning strategy [30].

To select better algorithm to optimize the weight values, we compare HSCO with traditional PSO algorithm in the weight values optimum process in evaluating the usability of
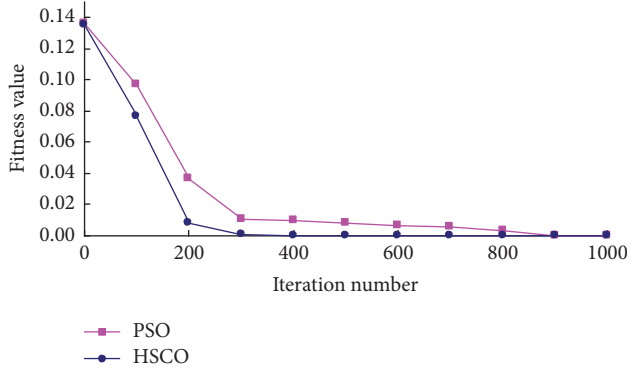
FIGURE 1: Fitness versus iteration by PSO and HSCO.

(1) Establishing the evaluation index system of educational resources software usability based on multidimensional attribute, and the value of metric element is obtained

(2) Generating the basic probability assignment by the interval number theory based on the historical data

(3) DS weighted value optimization model for swarm intelligence and its solution

(4) Hierarchical multiattribute synthetic evaluation and software final usability evaluation

the convenient educational resource management platform software (http://222.24.63.99:8080/). The experiments setting is as follows: for HSCO, $N_{\text{pop}} = 350$, $N_c = 70$, and $T = 1000$; for SCO, the number of particles is also 350. The two algorithms are executed 30 times. We calculate the mean solution by means of having a statistical computation for each run of the HSCO and PSO algorithm.

Figure 1 shows the mean fitness values of 30 times' iteration, which is performed by PSO and HSCO. The HSCO algorithm shows higher convergence velocity and higher sustainable evolutionary capability at the process of evolution than traditional PSO algorithm. HSCO algorithm can meet the requirement of numerical value in 400 iterations, while the PSO algorithm needs 900 to achieve the same fitness. The HSCO algorithm has higher efficiency in solving weight values optimum problem. Hence, we choose the HSCO algorithm to optimize the weight values in the paper.

4. Usability Evaluation Model of Educational Resources Software

4.1. Model Building. The usability of software is often influenced by many factors, which are called the available attributes of software. Different software focuses on different available attributes. To get the software's availability, you can firstly get the level of availability of the software on its available properties and then get the software availability comprehensively. When an available attribute can be subdivided into more detailed subattributes, the availability of this available property can be obtained by synthesizing the availability of these subattributes. The indivisible attribute is called metric element. The most important aspect of software usability evaluation is to find availability of the metric and to integrate the metric and subattributes evaluation into software availability [31]. Software availability measurement model is the process of subattribute synthesis, which is a multiple attribute decision-making model. Figure 2 shows a framework for the usability evaluation of educational resources software based on evidence reasoning.

As seen from Figure 2, usability evaluation of educational resources software based on evidential reasoning includes the following key processes:

4.2. Usability Evaluation Index System of Educational Resources Software. The ISO/IEC25000:2014 [32] is an important international standard for software quality and software evaluation, and a quantitative evaluation method for software usability is provided in the standard. This paper proposes a usability evaluation index system for assessing the availability of educational resources software, which combines the characteristics of educational resources software and target user characteristics. The usability evaluation index system of educational resources software mainly considers several aspects related to the influence factors of usability in Figure 3.

(1) Functionality. Functionality is the most direct aspect of influencing usability. For the educational software based on B/S architecture, the compatibility of browser and the operability of software basic function will be directly related to the usability of the software.

(2) Effectiveness of Information. Information validity is that the software can provide users with valid value resources information. In educational software, it specifically refers to the integrity and accuracy of information resources, the speed of software access and download, and the security of software.

(3) Usability. Usability mainly involves operability, learnability, and memorability. In educational software, the reasonable arrangement and classification of various multimedia data resources can also help users find required teaching resources quickly and accurately.

(4) User Interface Design. Most educational resources software users are teachers and students, which are professional. They will consider the design of the user interface of software products from aesthetic pleasure, functional completeness, and operation convenience.

Based on the above aspects factors and ISO/IEC25000 software quality standards, this paper builds the usability evaluation index system of educational resources software as shown in Figure 4. Of course, in the specific evaluation process, we can increase or decrease evaluation indexes in Figure 4 according to the actual requirements, so the index system has certain flexibility and extensive adaptability.

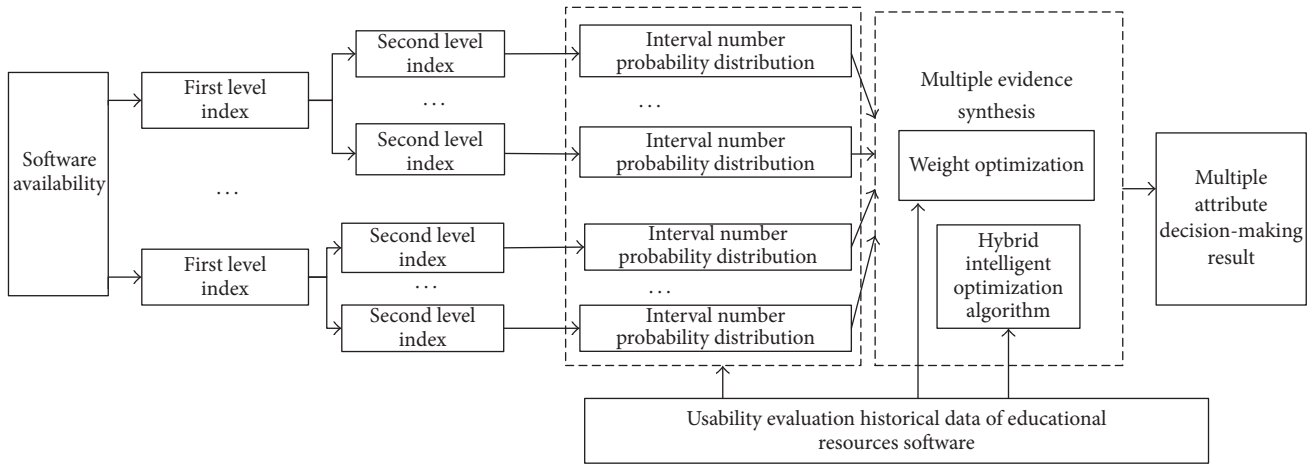The meaning of each attribute and subattribute is presented as follows.

FIGURE 2: Usability evaluation frame of educational resources software based on evidential reasoning.
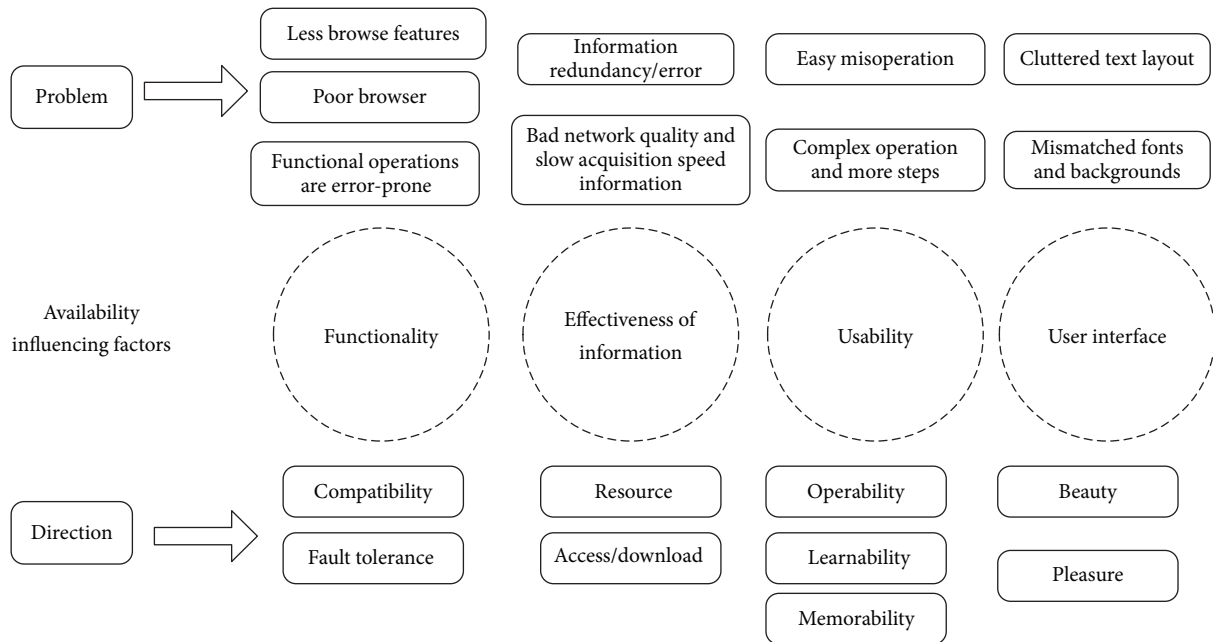

FIGURE 3: Availability influencing factors.

(1) Functionality

Completeness. The software products can provide users with various functions for performing a task.

Compatibility. The software products in system and browser environment can work steadily without exclusion phenomenon.

Fault Tolerance. When users make some illegal or incorrect input and operation, the software products can still work properly without crashes.

(2) Effectiveness

Resource Accuracy. The software products can provide users with useful data resources.

Resource Integrity. Data resources should ensure the integrity of the content.

Access Speed. It is the total time from clicking the website page to completely opening the page.

Download Speed. It is the speed through which users download documents, pictures, and multimedia resources.

Safety. Software products should protect data resources and users' information.

(3) Usability

Learnability. A user who has never used the new software can get quickly familiar with the basic functions of the software.
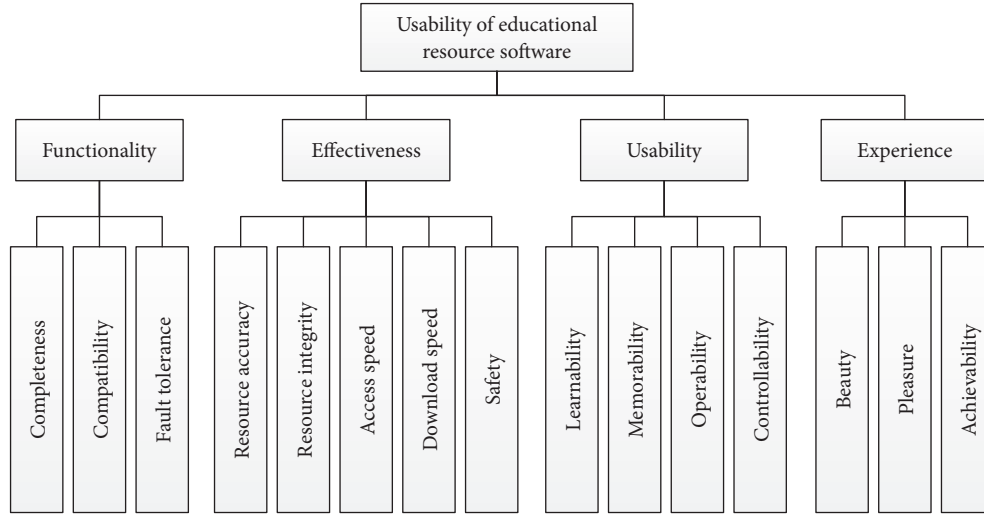
FIGURE 4: Usability index system.

Memorability. A software user at the second-time use can quickly recall the first use of software function.

Operability. The software should make users operate easily in the software interface design, button layout, link form, and so forth.

Controllability. Skilled users of software can give feedback on defects in a timely manner, and new improvements can be explained.

(4) Experience

Beauty. The users have a comfortable feeling for the software design features.

Pleasure. The users enjoy the overall process of software operation.

Achievability. The users have a psychological balance feeling resulting between the desire and the reality.

4.3. Basic Probability Assignment Generation Based on Interval Number. In this paper, the interval number, which evaluates the similarity between intervals by using the interval distance, is used to generate the BPA, which is based on the historical sample data [19].

Let interval number $A = [a^-, a^+]$ and interval number $B = [b^-, b^+]$; then the distance $D(A, B)$ between interval numbers A and B is given in the following formula [33]:

$$D(A, B) = \sqrt{\int_{-1/2}^{1/2} \left(\left(\frac{a^- + a^+}{2} + x(a^+ - a^-) \right) - \left(\frac{b^- + b^+}{2} + x(b^+ - b^-) \right) \right)^2 dx}$$

$$= \sqrt{\left(\frac{a^- + a^+}{2} - \frac{b^- + b^+}{2} \right)^2 + \frac{((a^+ - a^-) + (b^+ - b^-))^2}{12}}.$$

(4)

The similarity degree $S(A, B)$ between the interval numbers A and B is given in the following formula [33]:

$$S(A, B) = \frac{1}{1 + \alpha D(A, B)}.$$

(5)

α is support coefficient. The function of α is to adjust the dispersion degree of similarity degree, and the dispersion of data can be increased by adjusting α, which can avoid excessive data concentration caused by identification error increases. Support coefficient is usually obtained by experience.

BPA generation algorithm with interval numbers includes the following four steps, described as follows:

(1) Interval number model is constructed by the maximum and minimum historical value of the characteristic attributes of the collected samples.

(2) The distance between attribute value of sample and interval number of model to be identified is calculated according to formula (4).

(3) According to formula (5), the corresponding similarity degree is calculated.

(4) BPA is generated by similarity degree normalization.

4.4. DS Weighted Values Optimum Model. To reduce influence of the inaccuracy of a single feature to the system's
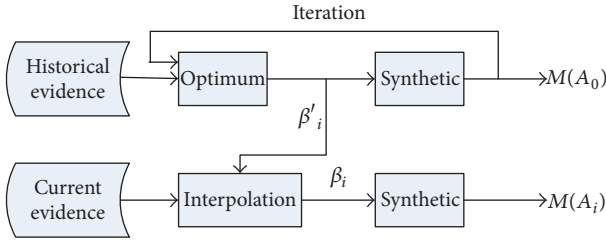
FIGURE 5: DS weight values optimum algorithm.

judgment and improve accuracy and stability of the system's judgment, it is necessary to integrate the features of various aspects. However, the importance of each feature in system judgment is different, so the influence of weight of different features on judgment result should be considered in the process of feature fusion [15]. To obtain objective weighted values of different features from the historical decision sample evidence data, we should build DS weighted values optimum model.

4.4.1. DS Weight Values Optimum Algorithm. Figure 5 shows the fundamental principle of *DS* weight values optimum algorithm. The process includes two phases, which are weight values optimum procedure and synthetic decision procedure [24].

In the weight values optimum procedure, upper values $\beta_i^{\max}$ and lower values $\beta_i^{\min}$ are generally given by experts

based on historical decision sample evidence data. The purpose of the evidence synthesis is to identify the goal as accurately as possible. Because the target A_0 is known, the weight value optimization is to find the best weight that maximizes A_0 synthesis BPA $M(A_0)$ and has the biggest difference from other targets.

In the synthetic decision procedure, the weight is interpolated to obtain the weights of the online optimization. Through the weight optimization and the weight interpolation phase, expert subjective experience and historical decision data have been used in a comprehensive way, and the optimization weight is got, which greatly improves the fusion effect of multiple evidence.

4.4.2. Weighted Optimum Model. The objective function is as follows:

$$g_i(A) = \frac{1}{M(A_0) - M(A_i)} \quad i = 1, 2, \ldots, r, \quad (6)$$

where $M(A_i)$ is the basic probability assignment of A_i; obviously $M(A_0) - M(A_i) > 0$.

$$g(A) = \max_{1 \le i \le r} \{g_i(A)\}. \quad (7)$$

The objective function is a nonlinear mini-max optimization problem [34]:

$$\min \quad \left\{ \max \left\{ \frac{1}{M(A_0) - M(A_i)} \mid i = 1, 2, \ldots, r \right\} \right\} \quad (8)$$

$$\text{Subject to} \quad M(A) = \begin{cases} 0, & A = \Phi \\ \dfrac{\sum_{A_1 \cap A_2 \cap \cdots \cap A_k = A} M_1^{\beta_1}(A_1) M_2^{\beta_2}(A_2) \cdots M_k^{\beta_k}(A_k)}{1 - \sum_{A_1 \cap A_2 \cap \cdots \cap A_k = \Phi} M_1^{\beta_1}(A_1) M_2^{\beta_2}(A_2) \cdots M_k^{\beta_k}(A_k)}, & \forall A \subset \Omega, \ A \ne \Phi. \end{cases} \quad (9)$$

In formula (9), $M(A_0) - M(A_i) > 0, 0 \le \beta_i \le 1, \sum_{i=1}^{r} \beta_i = 1$, $\beta_i^{\min} \le \beta_i \le \beta_i^{\max}$, and $\beta_i^{\max}$ and $\beta_i^{\min}$ indicate the upper bound and lower bound on the weights of the evidence given by the expert i. Therefore, the weight optimization model is a constrained mini-max optimization problem.

Using maximum entropy function method, the nonlinear constrained mini-max problem can be transformed into a smooth optimization problem [34]. In formula (7), the maximum entropy function of $g(x)$ in $x \in R^n$ is

$$G_p(x) = \frac{1}{p} \ln \left\{ \sum_{i=1}^{n} \exp[p * g_i(x)] \right\}. \quad (10)$$

As long as p is large enough, the maximum entropy function can take the place of the objective function $g(x)$. So the weighted optimum problem can be transformed into an unconstrained optimization problem with a smooth function [20].

4.4.3. HSCO Algorithm Optimization Weight. The problem of formula (6) is transformed into a smooth unconstrained optimization problem through the maximum entropy function method, which is equivalent to solving the following extreme optimization problem:

Find: $\quad X = [x_1, x_2, \ldots, x_n]$

Min: $\quad G_p(x) = \frac{1}{p} \ln \left\{ \sum_{i=1}^{n} \exp[p * g_i(x)] \right\}. \quad (11)$

In formula (11), p is a larger positive number which generally is 10^5. We use the hybrid social cognitive optimization (HSCO) algorithm to search the optimal weight. The research result shows that the weighted algorithm of this evidence theory is effective and has better fusion result than the particle swarm optimization algorithm. When HSCO algorithm is used to solve this problem, $G_p(x)$ is used as the fitness evaluation function of the knowledge point.

5. Application Analyses

To verify the effect of proposed software usability evaluation model for educational resources software, we used the proposed hybrid intelligent optimization model to evaluate the usability of the convenient educational resources management platform (http://222.24.63.99:8080/), which was developed by our software team in 2015. The whole software usability level is divided into five grades. Let $A = \{A_1, A_2, A_3, A_4\}$ note primary index collection, respectively: "functionality," "effectiveness," "usability," and "experience." Let $A_1 = \{A_{11}, A_{12}, A_{13}\}$, $A_2 = \{A_{21}, A_{22}, A_{23}, A_{24}, A_{25}\}$, $A_3 = \{A_{31}, A_{32}, A_{33}, A_{34}\}$, and $A_4 = \{A_{41}, A_{42}, A_{43}\}$, note secondary index collection. So, $A_1 = \{$completeness, compatibility, fault tolerance$\}$, $A_2 = \{$resource accuracy, resource integrity, access speed, download speed, safety$\}$, $A_3 = \{$learnability, memorability, operability, controllability$\}$, and $A4 = \{$beauty, pleasure, achievability$\}$. Let $U = \{S_1, S_2, S_3, S_4, S_5\}$ note evaluation set, respectively: "very low," "low," "general," "high," and "extremely high."

5.1. BPA Calculation Based on Interval Number Model. Firstly, software usability interval model is constructed by the maximum values and the minimum values of the historical

sample data, which are evaluated by the experts. The experts are domain experts in the field of education who have a clear grasp of teaching contents, teaching rules, the level of acceptance of educational objects, and software availability. Taking primary index A3 as an example, the sample interval model is shown in Table 1.

Secondly, the BPA of all the secondary indexes is calculated on the distance between interval numbers and the evaluation values. Suppose that the educational resources software usability measurement value of secondary indexes (learnability, memorability, operability, and controllability) is 0.95, 0.91, 0.89, and 0.95, respectively. Take learnability BPA calculation as example; Table 2 presents the calculation process of the learnability similarity and the learnability BPA: the value of the learnability is 0.95, which we can take as an interval [0.95, 0.95]. Then, we can calculate the distance between interval [0.95, 0.95] and sample interval model as in Table 1. The calculation processes of the distance between [0.95, 0.95] and the sample intervals are as follows.

In the application interval number $A = [0.95, 0.95]$ and interval numbers $B = \{[0.1, 0.4], [0.3, 0.55], [0.5, 0.7], [0.7, 0.9], [0.9, 1.0]\}$; then the distance $D(A, B_i)$ between interval numbers A and B_i is calculated by formula (4):

$$D(A, B_1) = D([0.95, 0.95], [0.1, 0.4]) = \sqrt{\left(\frac{0.95 + 0.95}{2} - \frac{0.1 + 0.4}{2}\right)^2 + \frac{((0.95 - 0.95) + (0.4 - 0.1))^2}{12}}$$

$$= \sqrt{0.7^2 + 0.0075} = 0.705337,$$

$$D(A, B_2) = D([0.95, 0.95], [0.3, 0.55]) = \sqrt{\left(\frac{0.95 + 0.95}{2} - \frac{0.3 + 0.55}{2}\right)^2 + \frac{((0.95 - 0.95) + (0.55 - 0.3))^2}{12}}$$

$$= \sqrt{0.525^2 + 0.001875} = 0.529937,$$

$$D(A, B_3) = D([0.95, 0.95], [0.5, 0.7]) = \sqrt{\left(\frac{0.95 + 0.95}{2} - \frac{0.5 + 0.7}{2}\right)^2 + \frac{((0.95 - 0.95) + (0.7 - 0.5))^2}{12}} \qquad (12)$$

$$= \sqrt{0.35^2 + 0.003333} = 0.354730,$$

$$D(A, B_4) = D([0.95, 0.95], [0.7, 0.9]) = \sqrt{\left(\frac{0.95 + 0.95}{2} - \frac{0.7 + 0.9}{2}\right)^2 + \frac{((0.95 - 0.95) + (0.9 - 0.7))^2}{12}}$$

$$= \sqrt{0.15^2 + 0.003333} = 0.160728,$$

$$D(A, B_5) = D([0.95, 0.95], [0.9, 1.0]) = \sqrt{\left(\frac{0.95 + 0.95}{2} - \frac{0.9 + 1.0}{2}\right)^2 + \frac{((0.95 - 0.95) + (1.0 - 0.9))^2}{12}}$$

$$= \sqrt{0^2 + 0.000833} = 0.0288675.$$

Then we can obtain the distance between interval numbers as shown in the first column of Table 2.

The similarity of the intervals can be calculated by similarity formula (5) as follows. The results are shown in the

TABLE 1: Sample interval model.

Assessment level	Index			
	A_{31}	A_{32}	A_{33}	A_{34}
S_1	[0.1, 0.4]	[0, 0.3]	[0.1, 0.4]	[0.0, 0.3]
S_2	[0.3, 0.55]	[0.2, 0.55]	[0.3, 0.59]	[0.3, 0.55]
S_3	[0.5, 0.7]	[0.5, 0.7]	[0.5, 0.7]	[0.5, 0.76]
S_4	[0.7, 0.9]	[0.7, 0.9]	[0.7, 0.9]	[0.75, 0.9]
S_5	[0.9, 1.0]	[0.93, 1.0]	[0.9, 1.0]	[0.85, 1.0]

TABLE 2: Learnability BPA.

Level	Distance	Similarity	BPA
S_1	0.705337	0.586394	0.153843
S_2	0.529937	0.653622	0.171480
S_3	0.354730	0.738154	0.193658
S_4	0.160728	0.861529	0.226026
S_5	0.0288675	0.971942	0.254993

TABLE 3: Memorability BPA.

Level	Distance	Similarity	BPA
S_1	0.764918	0.566598	0.148773
S_2	0.544457	0.647477	0.170009
S_3	0.315331	0.760265	0.199624
S_4	0.124231	0.889497	0.233557
S_5	0.0585946	0.944649	0.248038

TABLE 4: Operability BPA.

Level	Distance	Similarity	BPA
S_1	0.645833	0.607595	0.155448
S_2	0.452806	0.688323	0.176101
S_3	0.295691	0.771789	0.197455
S_4	0.106927	0.903402	0.231127
S_5	0.0665833	0.937573	0.239869

TABLE 5: Controllability BPA.

Level	Distance	Similarity	BPA
S_1	0.655744	0.603958	0.150747
S_2	0.381881	0.723651	0.180622
S_3	0.185831	0.843290	0.210483
S_4	0.050000	0.952381	0.237712
S_5	0.132288	0.883168	0.220437

middle column of Table 2. The support coefficient α is set to 1.0 as usual.

$$
\begin{aligned}
S(A, B_1) &= \frac{1}{1 + D(A, B_1)} = \frac{1}{1 + 0.705337} \\
&= 0.586394; \\
S(A, B_2) &= \frac{1}{1 + D(A, B_2)} = \frac{1}{1 + 0.529937} \\
&= 0.653622; \\
S(A, B_3) &= \frac{1}{1 + D(A, B_3)} = \frac{1}{1 + 0.354730} \\
&= 0.738154; \\
S(A, B_4) &= \frac{1}{1 + D(A, B_4)} = \frac{1}{1 + 0.160728} \\
&= 0.861529; \\
S(A, B_5) &= \frac{1}{1 + D(A, B_5)} = \frac{1}{1 + 0.0288675} \\
&= 0.971942.
\end{aligned}
\tag{13}
$$

Then the BPAs are obtained by normalized operation as follows. The results are shown in the last column of Table 2.

$$
\begin{aligned}
\mathrm{BPA}(A, S_1) &= \frac{S(A, B_1)}{\sum_{i=1}^{5} S(A, B_i)} \\
&= \frac{0.586394}{0.586394 + 0.653622 + 0.738154 + 0.861529 + 0.971942} \\
&= \frac{0.586394}{3.811641} = 0.153843; \\
\mathrm{BPA}(A, S_2) &= \frac{0.653622}{3.811641} = 0.171480; \\
\mathrm{BPA}(A, S_3) &= \frac{0.738154}{3.811641} = 0.193658;
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{BPA}(A, S_4) &= \frac{0.861529}{3.811641} = 0.226026; \\
\mathrm{BPA}(A, S_5) &= \frac{0.971942}{3.811641} = 0.254993.
\end{aligned}
\tag{14}
$$

As the same processing, Table 3 presents BPA generation process of the easy memory interval number similarity, Table 4 presents BPA generation process of the operability interval number similarity, and Table 5 presents BPA generation process of the controllable interval number similarity. From the BPA generation process, it is seen that interval number generating method is simple and requires less decision-making object information.

5.2. Analysis of Weight Values Optimum Algorithm. According to the DS theory in Section 2.1, the equal weight usability evaluation fusion of the four attributes and their subattributes is as follows.

Set the matrix of the BPA obtained by interval number theory B:

$$
\begin{aligned}
&B \\
&= \begin{bmatrix} 0.153843 & 0.171480 & 0.193658 & 0.226026 & 0.254993 \\ 0.155448 & 0.170009 & 0.199624 & 0.233557 & 0.248038 \\ 0.155448 & 0.176101 & 0.197455 & 0.231127 & 0.239869 \\ 0.150747 & 0.180622 & 0.210483 & 0.237712 & 0.220437 \end{bmatrix},
\end{aligned}
$$

TABLE 6: Usability attributes fusion result.

Attribute	Level				
	S_1	S_2	S_3	S_4	S_5
Learnability	0.153843	0.171480	0.193658	0.226026	0.254993
Memorability	0.155448	0.170009	0.199624	0.233557	0.248038
Operability	0.155448	0.176101	0.197455	0.231127	0.239869
Controllability	0.150747	0.180622	0.210483	0.237712	0.220437
Equal weight fusion	0.060001	0.099292	0.1720404	0.310563	0.358098
PSO fusion	0.054702	0.100286	0.18258557	0.289597	0.372829
HSCO fusion	0.055892	0.099233	0.17202167	0.284289	0.388564

$$\sum_{1 \le j \le 5} \prod_{1 \le i \le 4} B_{ij}$$

$$= 0.153843 * 0.155448 * 0.155448 * 0.150747 + 0.171480$$

$$* 0.170009 * 0.176101 * 0.180622 + 0.193658$$

$$* 0.199624 * 0.197455 * 0.210483 + 0.226026$$

$$* 0.233557 * 0.231127 * 0.237712 + 0.254993$$

$$* 0.248038 * 0.239869 * 0.220437 = 0.009339,$$

$$\prod_{1 \le i \le 4} B_{i1} = 0.153843 * 0.155448 * 0.155448 * 0.150747$$

$$= 0.00056,$$

$$\prod_{1 \le i \le 4} B_{i2} = 0.171480 * 0.170009 * 0.176101 * 0.180622$$

$$= 0.000927,$$

$$\prod_{1 \le i \le 4} B_{i3} = 0.193658 * 0.199624 * 0.197455 * 0.210483$$

$$= 0.001606695,$$

$$\prod_{1 \le i \le 4} B_{i4} = 0.226026 * 0.233557 * 0.231127 * 0.237712$$

$$= 0.0029,$$

$$\prod_{1 \le i \le 4} B_{i5} = 0.254993 * 0.248038 * 0.239869 * 0.220437$$

$$= 0.003344,$$

$$M(S_1) = \frac{\prod_{1 \le i \le 4} B_{i1}}{\sum_{1 \le j \le 5} \prod_{1 \le i \le 4} B_{ij}} = \frac{0.00056}{0.009339} = 0.060006,$$

$$M(S_2) = \frac{0.000927}{0.009339} = 0.099292,$$

$$M(S_3) = \frac{0.001606695}{0.009339} = 0.172040401,$$

$$M(S_4) = \frac{0.0029}{0.009339} = 0.310563,$$

TABLE 7: Weighted values after optimization.

Attribute	Algorithm	
	PSO	HSCO
Learnability	0.301	0.313
Memorability	0.292	0.271
Operability	0.251	0.235
Controllability	0.156	0.181

$$M(S_5) = \frac{0.003344}{0.009339} = 0.358098.$$

$$(15)$$

Then we get the result as shown in the fifth line in Table 6.

According to literature [21], traditional PSO algorithm is not guaranteed to converge to global optimal solution with probability of one, because the traverse direction of particle iteration formula is too strong. Otherwise, HSCO algorithm is guaranteed to converge to global optimal solution with probability of one because of the elite learning strategy and the chaotic learning strategy.

To evaluate the performance of the HSCO, we compare HSCO algorithm with traditional PSO algorithm in the weight values optimum process of the convenient educational resources management platform software in Section 3. From Section 3 analysis, the HSCO algorithm has higher efficiency in solving weight values optimum problem. So, this paper chooses the HSCO algorithm to optimize the weight values.

To obtain objective weighted values of different features from the historical decision sample evidence data, we build DS weighted values optimum model according to Section 4.4 using traditional PSO algorithm and HSCO algorithm, respectively. The result of the weighted values after optimization is shown in Table 7. From Table 7, we can see that the weights of the four evidences are different. The knowledge from the historical evaluation samples is very helpful to more accurately carry out the usability evaluation of the new educational resources software.

Table 6 shows the BPA results of four usability sub-attributes, equal weight DS fusion result, weight fusion result optimized by using particle swarm optimization (PSO) algorithm, and weight fusion optimization results using the HSCO algorithm. It can be seen from Table 6 that the optimizing evidence synthesis results can more clearly recognize

TABLE 8: Four public educational resources sites.

Educational resources website name	Internet site	Availability of evaluation results	Evaluate the sort	Student experiment sort
Convenient education	http://222.24.63.99:8080/	S5	1	2
Software engineering course resources	http://222.24.63.100:9102/	S4	3	3
High-level language programming	http://gjyy.csxupt.com/gjyy/	S3	4	4
Data structure resource sharing course website	http://jpkc.nwu.edu.cn/datastr/index.htm	S5	1	1

the level of the usability than the standard DS synthesis results, because the optimizing evidence synthesis may reflect different importance of different evidence. Compared with traditional particle swarm optimization algorithm [22], the fusion results have stronger recognition ability and better solve the problem of the multiple attribute fusion. The BPA values of the primary index can be obtained from the evidence of the interval numbers, and the final multiattribute fusion evaluation can be produced by weighted DS evidence theory.

5.3. Evaluation Results and User Test Results Contrast. To judge the applicability of the usability evaluation method proposed in this paper, four public educational resource sites in Table 8 are evaluated by the method of this article. 60 students, who used the four sites to learn lessons, give four educational resources availability vote sorting result. From Table 8, we can see that evaluation results and the availability of 60 students sorting results are consistent, and the proposed method is an effective educational resources assessment approach, which has important guiding significance for educational resource availability design.

6. Conclusion

This paper proposes a novel educational resource software usability assessment method. The method uses the interval number theory to generate BPA, which is more simple and convenient. DS evidence theory fuses evaluation results of all the evidence, which greatly eliminates fuzziness of evidence synthesis. Meanwhile, evidence weight optimization makes the quantitative results more accurate. Experimental results show that the novel evaluation method is of accurate recognition rate even in the case of few sample data. Based on the clear evaluation results, we can trace back the software usability problems to a certain extent and put forward improvement scheme to enhance the usability of the software.

In the future, we will research further the following problems. Firstly, we will try to apply the proposed method to more practical educational resources software and import extra procedure to reduce attribute weights computational complexity. Furthermore, the possible dependencies among the attributes of the actual software system will be considered to improve the proposed method.

Conflicts of Interest

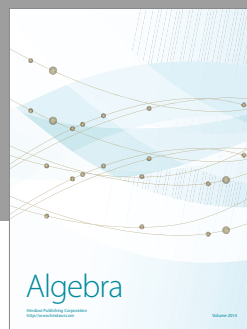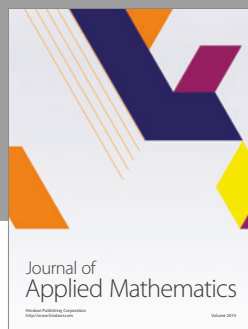The author declares that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] G. H. Shen, Z. Q. Huang, B. Xie et al., "Survey on software trustworthiness evaluation: standards, models and tools," *Journal of Software. Ruanjian Xuebao*, vol. 27, no. 4, pp. 955–968, 2016.

[2] F. Nayebi, J. M. Desharnais, and A. Abran, "The state of the art of mobile application usability evaluation," in *Proceedings of the 25th IEEE Canadian Conference on Electrical and Computer Engineering, CCECE 2012*, Montreal, Canada, May 2012.

[3] R. Harrison, D. Flood, and D. Duce, "Usability of mobile applications: literature review and rationale for a new usability model," *Journal of Interaction Science*, vol. 1, no. 1, article 1, 2013.

[4] A. Kaikkonen, A. Kekäläinen, M. Cankar et al., "Usability testing of mobile applications: A comparison between laboratory and field testing," *Journal of Usability studies*, vol. 1, no. 1, pp. 4–16, 2005.

[5] C. C. Chen, "Research of user experience oriented mobile B2C interface usability evaluation," *Chongqing: Southwest University*, 2014.

[6] L. P. Zhang, Z. J. Liu, H. X. Zhang et al., "Usability evaluation for IT products," *CEA*, vol. 39, no. 9, pp. 73–75, 2003.

[7] Z. J. Liu, J. L. Chen, L. P. Zhang et al., "A usability maturity assessment at Chinese software enterprises," *Computer Science*, vol. 31, no. 7, pp. 127–130, 2004.

[8] F. Guo, Z. Hao, and N. Xu, "A usability evaluating method for application software based on emotional experience," *Industrial Engineering and Management*, vol. 18, no. 2, pp. 146–152, 2013.

[9] S. E. Van Nuland, R. Eagleson, and K. A. Rogers, "Educational software usability: Artifact or Design?" *Anatomical Sciences Education*, vol. 10, no. 2, pp. 190–199, 2017.

[10] Q. Zhao, X. Zang, L. X. Wang et al., "Evaluation method of software usability process based on fuzzy analytic hierarchy process," *Application Research of Computers*, vol. 30, no. 9, pp. 2730–2735, 2013.

[11] J. G. Li, L. M. Shen, and C. X. Zhao, "Research on usability evaluation method of flexible point for user-oriented software," *Computer Applications and Software*, vol. 28, no. 1, pp. 61–64, 2011.

[12] J. Y. Li, S. Y. Wang, and J. Z. Sun, "Evaluation model of software usability based on weighted D-S," *Computer Engineering and Design*, vol. 37, no. 1, pp. 118–184, 2016.

[13] J. B. Yang and D. L. Xu, "On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 32, no. 3, pp. 289–304, 2002.

[14] N. Pongsathornwiwat, V.-N. Huynh, T. Theeramunkong, and C. Jeenananta, "Linguistic partner evaluation model in tourism supply chain networks," in *Proceedings of the 2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016*, pp. 1051–1058, Vancouver, Canada, July 2016.

[15] T. Bao, X. Xie, P. Long, and Z. Wei, "MADM method based on prospect theory and evidential reasoning approach with unknown attribute weights under intuitionistic fuzzy environment," *Expert Systems with Applications*, vol. 88, pp. 305–317, 2017.

[16] Y. C. Zhang, J. M. Pang, and R. C. Zhao, "Evidential reasoning method for decision of program maliciousness," *Journal of Software*, vol. 23, no. 12, pp. 3149–3160, 2012.

[17] H. Lee, H. Kwon, R. Robinson M et al., "Dynamic belief fusion for object detection," in *Proceedings of the 2016 IEEE Winter Conference*, pp. 1–9, Applications of Computer Vision (WACV), New York, NY, USA, 2016.

[18] M. Li, X. Lu, Q. Zhang, and Y. Deng, "Multiscale probability transformation of basic probability assignment," *Mathematical Problems in Engineering*, vol. 2012, 6, no. 40, Article ID 319264, pp. 1092–1096, 2012.

[19] Y. Deng, W. K. Shi, Z. F. Zhu et al., "Combining belief functions based on distance of evidence," *Decision Support Systems*, vol. 38, no. 3, pp. 489–493, 2004.

[20] S. P. Wan, "Interval number method for object threat assessmen," *CEA*, vol. 45, no. 6, pp. 32–34, 2009.

[21] Z. He and W. Jiang, "A new belief Markov chain model and its application in inventory prediction," https://arxiv.org/abs/1703.01963, 2017.

[22] H. Y. Guo, L. Zhang, and J. X. Zhou, "Identification of structural multiple damaged locations based on Dempster-Shafer theory of weighted balance of evidence," *Engineering Mechanics*, vol. 22, no. 1, pp. 235–240, 2005.

[23] H. Liu, Z. Zhao, and H. Ba, "Mutisensor target identification method based on weighted evidence combination," *Journal of PLA University of Science and Technology Natural Science Edition*, vol. 6, no. 6, pp. 521–524, 2005.

[24] B. Wang, G. Liang, and C. Wang, "D-S algorithm based on particle swarm optimizer," in *Proceedings of the 2007 8th International Conference on Electronic Measurement and Instruments, ICEMI*, pp. 2311–2315, Xi'an, China, August 2007.

[25] Z. Liu, Q. Pan, J. Dezert, and A. Martin, "Combination of classifiers with optimal weight based on evidential reasoning," *IEEE Transactions on Fuzzy Systems*, no. 99, pp. 1-1, 23 June 2017.

[26] J. Z. Sun, G. H. Gang, S. Y. Wang, and M. Q. Zhou, "Hybrid social cognitive optimization algorithm for constrained nonlinear programming," *Journal of China Universities of Posts and Telecommunications*, vol. 19, no. 3, pp. 91–99, 2012.

[27] O. Basir and X. H. Yuan, "Engine fault diagnosis based on multisensor information fusion using Dempster-Shafer evidence theory," *Information Fusion*, vol. 8, no. 4, pp. 379–386, 2007.

[28] X. F. Xie, W. J. Zhang, and L. Z. Yang, "Social cognitive optimization for nonlinear programming problems," in *Proceedings of International Conference on Machine Learning and Cybernetics*, vol. 2 of *783*, pp. 4–779, Beijing, China, 2002.

[29] J. Z. Sun, S. Y. Wang, and J. K. Zhang, "SCO algorithm based on entropy function for NCP," *CEA*, vol. 46, no. 21, pp. 40–42, 2010.

[30] J. Z. Sun, S. Y. Wang, and H. Chen, "A guaranteed global convergence social cognitive optimizer," *Mathematical Problems in Engineering*, vol. 2014, Article ID 534162, 8 pages, 2014.

[31] A. Solano, C. A. Collazos, C. Rusu, and H. M. Fardoun, "Combinations of methods for collaborative evaluation of the usability of interactive software systems," *Advances in Human Computer Interaction*, vol. 2016, Article ID 4089520, 2016.

[32] ISO/IEC 25000:2014 Preview Systems and software engineering—Systems and software Quality Requirements and Evaluation.

[33] J. Ding, D. Han, J. Dezert, and Y. Yang, "Comparative study on BBA determination using different distances of interval numbers," in *Proceedings of the 2017 20th International Conference on Information Fusion (Fusion)*, pp. 1–6, Xi'an, China, July 2017.

[34] H. W. Tang, L. W. Zhang, and X. H. Wang, "A maximum entropy method for a class of constrained nondifferentiable optimization problems," *Mathematica Numerica Sinica*, vol. 15, no. 3, pp. 268–275, 1993.