A hybrid retrieval strategy for case-based reasoning using soft likelihood functions

Yameng Wang[1] · Liguo Fei[1] · Yuqiang Feng[1] · Yanqing Wang[1] · Luning Liu[1]

Abstract
According to characteristics of new problems, the process of finding one or more similar cases from the existing cases to get a new solution is called case-based reasoning (CBR). The kernel idea of CBR is similar in cases having similar solutions. CBR can play its best role only by finding cases that are most similar to new problems through some retrieval methods. Currently, commonly used case retrieval algorithms are basically based on mean operator method. Although the difficulty of calculation is low, the accuracy is limited, and if a certain local similarity is low, the overall result can be affected. We introduce the soft likelihood functions into case retrieval, combine it with KNN, and propose a hybrid retrieval strategy, which is a new and softer way to calculate case similarity. The core of our hybrid retrieval strategy is to aggregate the local similarity and feature similarity of cases by soft likelihood functions, so as to obtain the global similarity. And at the same time, take into account the different attitudinal characteristics of the decision-maker, whether optimistic or pessimistic. The accuracy of this strategy is more than 81% in simulation experiments on real data sets, which verifies its effectiveness.

Keywords Case-based reasoning · Ordered weighted average · Soft likelihood function · Case retrieval · Attitudinal character

1 Introduction

The proposal of case-based reasoning (CBR) can be traced back to the late 1970s (Schank 1983). Roger et al. from Yale University in the USA proposed to represent knowledge by means of script, which is regarded as the beginning of CBR research. Since then, CBR has experienced from simple basic application research to theoretical perfection (Kolodner and Simpson 1989; Navarro et al. 2011; Müller and Bergmann 2015; Homem et al. 2020; Le et al. 2020). It originated in the fields of cognitive science (CS) and artificial intelligence (AI). Typically, target cases are used to represent current problems or situations, and source or historical cases are used to represent problems or situations that have occurred. CBR refers to recalling previous successful cases, comparing the similarities and differences between source cases and the target case, finding successful cases that are similar to the current situation, then adapting and applying its solutions to solve the current problem Armengol et al. (2001). In particular, CBR acts a pivotal part in the field of application where there is no known standard, no known cycle, and no complete domain theory (Schmidt and Gierl 2005). CBR can simplify knowledge acquisition, improve problem solving efficiency, improve solving quality, and accumulate knowledge. It provides a method which is quite similar to human solving problems (Jian et al. 2015).

At present, CBR has been widely used in AI, and it has become a new methodology of problem solving and learning (Liu et al. 2019). With the gradual maturity of theories and methods, the applications of CBR have been extended to various fields, including medical treatment (Holt et al. 2005; Georgopoulos and Stylios 2008; Begum et al. 2010; Ramos-González et al. 2017; Torrent-Fontbona et al. 2019), planning (Pinto et al. 2018; Jiang et al. 2019), assessment (Liang et al. 2012; Hong et al. 2015), forecast (Kwon et al. 2020; Xu et al. 2021), game (Catalá et al. 2014), recommendation system (Alshammari et al. 2017), management (González-Briones et al. 2018) and so on (Floyd et al. 2015; Le Ber et al. 2018).

Plenty of scholars developed different CBR models with the intention of providing a better understanding of CBR process. One of the representative models is introduced by

✉ Luning Liu
liuluning@hit.edu.cn

1 School of Management, Harbin Institute of Technology,
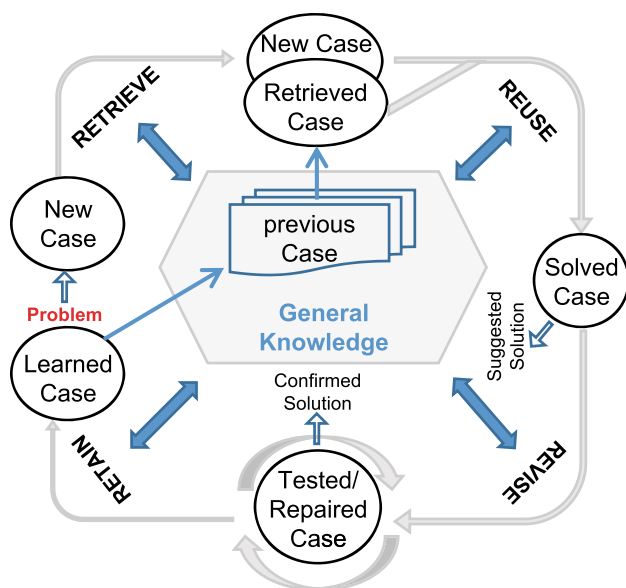Harbin 150001, China

Fig. 1 The 4-R lifecycle model of CBR from Aamodt and Plaza (Aamodt and Plaza 1994)

(Aamodt and Plaza 1994), in which they propose a process to sovle a new problem. Before reasoning, we need to choose the appropriate method to build a case base Bergmann et al. (2005). For a problem to be solved, we need retrieving one or more similar cases from the existing case base according to the characteristics. Solutions to cases retrieved are employed to create solutions to the new problem, and the solutions will be tested, modified, and evaluated to determine their effectiveness. Solutions that satisfy the user are learned and stored to case bases. The model of the CBR cycle is illustrated in Fig.1, which is called the 4-R lifecycle model.

From the model of CBR cycle, the CBR reasoning process is mainly divided into four stages: retrieval (R-1), reuse (R-2), revise (R-3), and retain (R-4) (De Mantaras et al. 2005).

- R-1: RETRIEVE information from source case base and select potentially available source cases.
- R-2: REUSE the solutions of retrieved source cases in new problems or cases.
- R-3: REVISE the proposed solutions.
- R-4: RETAIN the solutions to the problem in favor of subsequent reasoning.

The 4-R cycle model is summarized as follows: analyze the features of existing problems, *retrieve* one or more similar cases, try to *reuse* cases, and *retain* new cases in case base in light of their importance after the solution is *revised* and applied.

From the 4-R cycle model, we can get the fact that the quality of case retrieval strategy can largely determine whether a CBR system can play a strong superiority (Kang et al. 2013).

The retrieval method directly affects the retrieval speed and accuracy rate (Petrovic et al. 2016), and whether the retrieval strategy is reasonable or not directly affects the realization effect of the whole case system. So case retrieval is the key to problem solving. In the aspect of retrieval strategy, there are knowledge guidance strategy (Rallabandi and Sett 2008), genetic algorithm strategy (Abualigah and Hanandeh 2015; Abualigah and Khader 2017), text metaheuristic strategy (Abualigah 2019), iterative methodology strategy (Marcos-Pablos and García-Peñalvo 2020), and nearest neighbor strategy (Cover and Hart 1967; Guo et al. 2014).

From the research status of case retrieval at home and abroad (Greene et al. 2010), the K-nearest neighbor (KNN) retrieval strategy is widely used at present (Schmidt et al. 2001). KNN means that each sample can be represented by its closest K neighboring values (Cover and Hart 1967). In the feature space, if most of the K samples closest to the sample belong to a certain category, then the sample is also classified into that category. KNN retrieval strategy calculates the similarity between the target case to be solved and the source cases according to the attribute weight and its eigenvalue (Li et al. 2009) and then selects one or some source case solutions with high similarity as the basis of case reuse (Lin and Chen 2011). In the calculation of similarity, the weight distribution will have a significant influence on the calculation results and the quality of the solution. Attributes that generally play a major role are assigned greater weight; conversely, less weight is given. KNN generally uses the average weight method. Although it is simple and easy to operate, it is sensitive to noise or irrelevant data, which will affect the reliability of the calculation results. The settlement of this problem usually relies on the reasonable allocation of the weight of characteristic attributes, so the allocation of weight has become an important research direction.

In CBR, although the current retrieval method based on similarity has attracted the attention of researchers and been widely used, it is not completely in keeping with the actual reasoning process. It is easily disturbed by small probability events, and the whole result is easily affected by a certain term. For another thing, as CBR systems are developed to facilitate decision-making by decision-makers (DMs), it is inevitable that they need to be able to reflect DMs' personal attitudes in different situations. However, the attitude characteristics of DMs are often neglected in similarity calculation, which is illogical. Therefore, it is necessary to further explore the mechanism of optimal weight allocation for the sake of improving the quality of problem solving.

On account of the above analysis, inspired by the soft likelihood functions (SLFs) introduced by Yager et al. (Yager et al. 2017), a new case retrieval algorithm using SLFs (abbreviated as CBR-SLFs) is proposed in this study, which offers a new perspective to case retrieval. SLFs allow the ordered weighted average (OWA) aggregation to soften the strong

likelihood constraint requirements of all information and, at the same time, provide weight for attitude features, allowing optimistic or pessimistic possible results. SLFs are more flexible than general algorithms, so they are called soft likelihood functions. The basic idea of case retrieval by the proposed method is as follows: Firstly, calculate the local similarity between different attributes of target and source case; then, the CBR-SLFs come up with in this paper is used to calculate the overall similarity, and some potential available source cases with high similarity are obtained; finally, the source case solution that is closest to the target case is obtained through KNN and reuses it. As a flexible method to calculate global similarity, this strategy has stronger robustness and practicability in case retrieval (Tian et al. 2020). Furthermore, SLFs-based case retrieval algorithm is developed introducing an attitudinal characteristic to reflect the subjective preference of DMs, which allows different types of DMs to make more flexible choices.

The rest part of the article is organized as shown below: Section 2 introduces likelihood function in case retrieval, some basic calculations of OWA aggregation operator, and local similarity measurement method for heterogeneous information. Section 3 introduces the application of soft likelihood function in case retrieval, then takes feature similarity into consideration, and gives some examples. Section 4 furnishes some simulated experiments on benchmark data sets. In the end, Section 5 summarizes this article and puts forward the future research direction.

2 Preliminaries

This part first presents likelihood functions in case retrieval and OWA aggregation and then introduces local similarity measurement methods for case information.

2.1 Using likelihood functions in case retrieval

In a CBR system, existing knowledge or experience needs to be represented as a case library typically includes multiple cases. Each case is generally composed of two parts, the problem description and the corresponding solution. For convenience of description, the symbol is given below.

$$C_i = \{D_i, S_i\}, i = 1, 2, \ldots, n \tag{1}$$

$C = \{C_1, C_2, \ldots, C_n\}$ is n historical cases in case base, C_i represents the ith case ($i \in \{1, 2, \ldots, n\}$) including problem description D_i and corresponding solution S_i. $\mathcal{C}*$ is the target case, and the problem description for the target case is represented as $\mathcal{D}*$. Suppose SIM_i represents the similarity between Ci and the target case. $Sim_j(\mathcal{D}*, D_i)$ represents the similarity of the problem description $\mathcal{D}*$ of target case and

the problem description D_i of the historical case C_i about the characteristic attribute j.

In case reasoning, our goal is to find some order of historical cases, that is, the similarity between historical and target cases, so as to support the selection of source cases with the highest similarity as candidate cases for further revision and use. In other words, the more similar the historical case is, the more willing we are to reuse the case. One way to calculate the similarity of a case is to take the product of the local similarity of different attributes.

$$SIM_i = \prod_{j=1}^{q} sim_{ij} \tag{2}$$

We can see that each additional feature can only reduce the probability that the case C_i is the most optimal candidate case. If any $sim_{ij} = 0$ for $j = 1 \ldots q$, then $SIM_i = 0$. For any case C_i, as long as there is a low local similarity value, the overall similarity of the case C_i will be greatly reduced. This is a kind of logical "anding" for a given C_i. The expression of this possibility is too strong, because it requires the premise that all the local similarity of C_i is consistent and high, so that we can think of this suspect's historical case as similar. Therefore, this paper will consider adopting OWA to determine the candidate case similarity of the softer formula. In the following text, we set λ_i as the index function and $\lambda_i(k)$ as the kth probability index of great compatibility of C_i. Here, $sim_{i\lambda_i(k)}$ is the kth largest local similarity of the case C_i. We let

$$Prod_i(j) = \prod_{k=1}^{j} sim_{i\lambda_i(k)} \tag{3}$$

Here, $Prod_i(j)$ is the product of the j largest probabilities. $Prod_i(j)$ is monotonically decreasing as a function of j. At the same time, every $sim_{i\lambda_i(k)} \in [0, 1]$, so $Prod_i(j) \in [0, 1]$. From the above equation, we find the likelihood function can now be expressed as $SIM_i = Prod_i(q)$.

2.2 Ordered weight averaging aggregation

Below, we will consider using OWA aggregation operator to provide a category of SLFs. In order to do this, OWA needs to be briefly described.

Ordered weight averaging aggregation was first proposed by Yager (1988). An OWA aggregator operator of n dimension is a mapping: $R^n \rightarrow R$. $OWA_w(a_1, a_2, \ldots, a_i, \ldots, a_n) = \sum_{j=1}^{n} w_j a_{\lambda(j)}$, where $W = (w_1, w_2, \ldots, w_n)^T$ is the weighted vector associated with the function OWA with $w_j \in [0, 1]$ and $\sum_j w_j = 1$ ($j \in \{1, 2, \ldots, n\}$); $a_{\lambda(j)}$ is the jth largest element in $a_1, a_2, \ldots, a_n$ in order from largest to

smallest. Then, we called function OWA as ordered weight averaging operator.

The characteristic of OWA is to rearrange the given data $(a_1, a_2, \ldots, a_i, \ldots, a_n)$ into $(a_{\lambda(1)}, a_{\lambda(2)}, \ldots, a_{\lambda(i)}, \ldots, a_{\lambda(n)})$ in order from large to small, and aggregate $(a_{\lambda(1)}, a_{\lambda(2)}, \ldots, a_{\lambda(i)}, \ldots, a_{\lambda(n)})$ by the given weight vector. Furthermore, element a_i has nothing to do with weight w_j, and weight w_j is only connected with the jth position in the assembly process, so we can also call the weighted vector W a position weighted vector.

Let's notice some special operators (Yager 1988):

1. $W^* = (1, 0, \ldots, 0)$, the OWA is reduced to the max operator, $OWA(a_1, \ldots, a_n) = a_{\lambda(1)} = max_i(a_i)$.
2. $W_* = (0, 0, \ldots, 1)$, the OWA is reduced to the min operator, $OWA(a_1, \ldots, a_n) = a_{\lambda(n)} = min_i(a_i)$.
3. $W_n = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$, the OWA is reduced to a simple arithmetic average operator,
   $OWA(a_1, \ldots, a_n) = \frac{1}{n}\sum_{i=1}^{n} a_i$.
4. $W_{n-2} = \left(0, \frac{1}{n-2}, \frac{1}{n-2}, \ldots, \frac{1}{n-2}, 0\right)$, the OWA is reduced to an arithmetic average operator that removes the extremum, $OWA(a_1, \ldots, a_n) = \frac{1}{n-2}(\sum_{i=1}^{n} a_i - max_i(a_i) - min_i(a_i))$
5. $W_k = (0, \ldots, 1, \ldots, 0)$, $OWA(a_1, \ldots, a_n) = a_{\lambda(k)}$.

When w_j near the top of W allocates more weight, the total value is larger; while w_j near the bottom of W allocates more weight, the total value is smaller. Weighted vector W can reflect the tendency of the DMs to be optimistic or pessimistic, and it determines how OWA is aggregated. Now, we define attitudinal character (Yager 1996):

$$AC(W) = \sum_{j=1}^{n} \frac{n-j}{n-1} w_j \qquad (4)$$

$AC(W) \in [0, 1]$ and the numerical value of $AC(W)$ determines the degree of optimism. In other words, the more optimistic the DM is, the greater the attitudinal eigenvalue is and the higher the aggregated value is.

We use a method to get OWA weights, w_j. Assume a monotonic function f: $[0, 1] \to [0, 1]$; when $x > y$, $f(x) > f(y)$; $f(0) = 0$ and $f(1) = 1$. We obtain

$$w_j = f\left(\frac{j}{n}\right) - f\left(\frac{j-1}{n}\right) \qquad (5)$$

We get $w_j \in [0, 1]$ and $\sum_{j=1}^{n} w_j = 1$; w_j satisfies all attributes required by OWA weights Yager (1996).

This method of obtaining OWA weights is called the function method, and the function itself and cardinality n jointly determine w_j and the associated attitudinal character. Then, we define the attitudinal character (Yager 1996):

$$Opt(f) = \int_0^1 f(x)dx \qquad (6)$$

When n gets really big, $Opt(f)$ is really just $AC(W)$.

It is easy to find out $f(x) = x^m$ for $m \geq 0$, and for this function,

$$\alpha = \int_0^1 x^m dx = \frac{x^{m+1}}{m+1}\bigg|_0^1 = \frac{1}{m+1} \qquad (7)$$

We have $m = \frac{1-\alpha}{\alpha}$, and $\alpha \in [0, 1]$. We can see that the larger the α, the more optimistic the attitude of users. $m = 1$ when $\alpha = 0.5$; $m = 0$ when $\alpha = 1$; $m \to \infty$ when $\alpha \to 0$.

Using the function form described above, we can get

$$w_j = f\left(\frac{j}{n}\right) - f\left(\frac{j-1}{n}\right) = \left(\frac{j}{n}\right)^m - \left(\frac{j-1}{n}\right)^m \qquad (8)$$

Then, α once given, we can obtain

$$w_j = \left(\frac{j}{n}\right)^{\frac{1-\alpha}{\alpha}} - \left(\frac{j-1}{n}\right)^{\frac{1-\alpha}{\alpha}} \qquad (9)$$

Then, we will next consider using OWA to determine softer formulas for computing similarity.

2.3 Local similarity measurement methods for case information

CBR is very similar to the way humans solve problems. When a new problem is encountered, CBR retrieves and selects possible source cases from the case bases by some retrieval method (Cunningham 2008). CBR can not only give full play to the advantage of the immediacy of computer processing information, but also improve the scientific nature and effectiveness of decision-making (El-Sappagh et al. 2019). In the CBR system, whether all the follow-up work can play its due role largely hinges on the quality of the cases retrieved, so case retrieval is very critical.

The information or data in a CBR system are usually heterogeneous, and heterogeneity indicates a difference in the type and nature of information or data (Yu et al. 2017). The key link in the decision-making process is to process heterogeneous information (Yahong and Xiuli 2018; Wan et al. 2016). As case events are usually characterized by risk, complexity, and uncertainty (Nikpour and Aamodt 2021), plus the imprecision of the environment, decision information is often not always expressed as accurate numbers (Fei et al. 2021), including Boolean values, interval values, fuzzy values, and so on. Furthermore, because of the fuzziness of human mind, sometimes in the decision-making process, expressing all decision information with quantitative values is very hard,

and qualitative language is also applied to describe attributes (Fei and Deng 2020; Fei and Feng 2021).

Suppose $Sim_j(\mathcal{D}*, D_i)$ represents the similarity between the target case $\mathcal{D}*$ and the historical case D_i about the characteristic attribute j. Heterogeneous decision information contains many types of attribute information such as numerical features, Boolean features, symbolic features with orders, symbolic features without orders, string features, fuzzy features, and interval features, and its similarity is calculated as follows (Tan et al. 2020):

- For numerical features, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = 1 - \frac{|\mathcal{D}* - D_i|}{\max} \quad (10)$$

- For Boolean features, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = \begin{cases} 0 & \mathcal{D}* \neq D_i \\ 1 & \mathcal{D}* = D_i \end{cases} \quad (11)$$

- For symbolic features with orders, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = 1 - \frac{|\mathcal{D}* - D_i|}{g} \quad (12)$$

where g is the number of value levels.
- For symbolic features without orders, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = \frac{num(\mathcal{D}* \wedge D_i)}{num(\mathcal{D}* \vee D_i)} \quad (13)$$

- For string features, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = \frac{t \times l}{\max(len(\mathcal{D}*), len(D_i))} \quad (14)$$

where t is the matching number, l is the matching length, and len is the string length.
- For fuzzy features, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i) = 1 - \{(n_i - n_i')^2$$
$$+ \frac{1}{9}[(m_i - m_i')^2 + (r_i - r_i')^2 - (m_i - m_i')(r_i - r_i')] \quad (15)$$
$$- \frac{1}{2}(n_i - n_i')[(m_i - m_i') - (r_i - r_i')]\}^{\frac{1}{2}}$$

$\mathcal{D}*, D_i$ are triangular fuzzy number, $\mathcal{D}* = (n_i, m_i, r_i)$, $D_i = (n_i', m_i', r_i')$

- For interval features, the similarity between $\mathcal{D}*$ and D_i can be obtained as

$$Sim_j(\mathcal{D}*, D_i)$$
$$= \frac{len(\mathcal{D}* \bigcap D_i)}{len(\mathcal{D}*) + len(D_i) - len(\mathcal{D}* \bigcap D_i)} \quad (16)$$

where len is the interval length and $\mathcal{D}* \bigcap D_i$ is the overlapping interval.

3 Case retrieval strategy

We first give a global similarity calculation method-based soft likelihood function that integrates the similarity of each attribute, and then, considering the feature similarity, we give a SLFs case retrieval algorithm combining the feature similarity. Our retrieval strategy is to combine case retrieval algorithm based on SLFs with KNN, thus improving the performance of case retrieval.

3.1 Case retrieval method based on SLFs

In the previous section, we have obtained local attribute similarity between target case and historical cases under a variety of heterogeneous information environments. The global similarity is then calculated to retrieve the historical cases that are most similar to the target case from the case base. We apply SLFs based on OWA to case retrieval process and propose an original global similarity calculation method to improve the previous case retrieval strategy.

Let's consider using SLFs-based OWA as a retrieval strategy for CBR. For each source case C_i that we denote global similarity as $SIM_{i,W}$, we use W and $Prod_i(j)$ to calculate it. Here, W is the weighting vector, $W = \{w_1, \ldots, w_q\}$, $w_j \in [0, 1]$, $\sum_{j=1}^n w_j = 1$. We define

$$SIM_{i,W} = \sum_{j=1}^q w_j Prod_i(j) \quad (17)$$

It has been pointed out above that $Prod_i(j) = \prod_{k=1}^j sim_{i\lambda_i(k)}$. Here, λ_i is index function hence $\lambda_i(k)$ is an index of the local similarity of attribute with the kth largest probability of compatibility of case C_i.

For each C_i, $Prod_i(j) = Prod_i(j - 1)sim_{i\lambda_i(k)}$, as $sim_{i\lambda_i(k)} \leq 1$, so $Prod_i(j)$ is decreasing in j. Therefore, the $Prod_i(j)$ using W based on OWA aggregation is

$$SIM_{i,W} = \sum_{j=1}^q w_j Prod_i(j)$$
$$= OWA_W\{Prod_i(1), \ldots, Prod_i(q)\} \quad (18)$$

We can see that the SLFs are determined by weighting vector W which is only related to the location. For some of the special weighting vector,

(1): $W^* = \{w_1 = 1, w_j = 0 | j = 2, \ldots, q\}$

$$SIM_{i,W^*} = Prod_i(1) = sim_{i\lambda_i(1)} \tag{19}$$

This is the maximum possible value, which is equal to the maximum probability in the property C_i.

(2): $W_* = \{w_q = 1, w_j = 0 | j = 1, \ldots, q-1\}$

$$SIM_{i,W_*} = Prod_i(q) = \prod_{j=1}^{q} sim_{ij} \tag{20}$$

This is the form of a strong likelihood function that requires all properties of D_j to be compatible with the target case C_i.

(3): $W_n = \{w_j = \frac{1}{q} | j = 1, \ldots, q\}$

$$SIM_{i,W_n} = \frac{1}{q}\sum_{j=1}^{q} Prod_i(j) = \frac{1}{q}\sum_{j=1}^{q}\left(\prod_{k=1}^{j} sim_{i\lambda_i(k)}\right) \tag{21}$$

This is the simple average.

(4): $W_n = \{w_1 = 0, w_j = 0, w_j = \frac{1}{q-2} | j = 2, \ldots, q-1\}$

$$
\begin{aligned}
SIM_{i,W_n} &= \frac{1}{q-2}\left(\sum_{j=1}^{q} Prod_i(j) - Prod_i(1) - Prod_i(q)\right) \\
&= \frac{1}{q-2}\left(\sum_{j=1}^{q}\left(\prod_{k=1}^{j} sim_{i\lambda_i(k)}\right) - sim_{i\lambda_i(1)} - \prod_{j=1}^{q} sim_{ij}\right)
\end{aligned} \tag{22}
$$

This is an arithmetic mean minus the extreme value.

DMs who are more optimistic about the likelihood will assign more weight to w_j that has a smaller index; DMs who are more pessimistic about the likelihood will assign more weight to w_j that has a larger index. Because $SIM_{i,W}$ is depending on W, we discover that the likelihood functions rest with α which can impact weighting vector. If the user is more positive, then the α is near to 1 and SIM_{i,W_N} is larger; if the user is more negative, then the α is closer to 0 and SIM_{i,W_N} is smaller.

This has been discussed above that $w_j = f\left(\frac{j}{q}\right) - f\left(\frac{j-1}{q}\right)$ and $f(x) = x^m$. In addition, we use $m = \frac{1-\alpha}{\alpha}$ to show the desired degree of optimum α. As a result, we can express users' attitude by a softer likelihood function which is more in line with the reality. We can get

$$SIM_{i,\alpha} = \sum_{j=1}^{q}\left(\left[\left(\frac{j}{q}\right)^{\frac{1-\alpha}{\alpha}} - \left(\frac{j-1}{q}\right)^{\frac{1-\alpha}{\alpha}}\right]\prod_{k=1}^{j} sim_{i\lambda_i(k)}\right). \tag{23}$$

Table 1 Probability products

Ordered probability	$Prod_i(j)$
$sim_{i\lambda_i(1)} = sim_{i4} = 1$	$Prod_i(1) = 1$
$sim_{i\lambda_i(2)} = sim_{i3} = 0.9$	$Prod_i(2) = 1 \times 0.9 = 0.9$
$sim_{i\lambda_i(3)} = sim_{i6} = 0.8$	$Prod_i(3) = 0.9 \times 0.8 = 0.72$
$sim_{i\lambda_i(4)} = sim_{i1} = 0.7$	$Prod_i(4) = 0.72 \times 0.7 = 0.504$
$sim_{i\lambda_i(5)} = sim_{i5} = 0.5$	$Prod_i(5) = 0.504 \times 0.5 = 0.252$
$sim_{i\lambda_i(6)} = sim_{i2} = 0.4$	$Prod_i(6) = 0.252 \times 0.4 = 0.1008$

Because of the physiological and cognitive limitations of the DMs, he is bounded rational in reality (Simon 1955). DMs' reasoning is not only influenced by the information of historical cases, but also implies their personal wisdom, emotion, attitude, cognition, etc. Psychological characteristics make a difference to decision-making process of DMs (Mi et al. 2021). Therefore, attitude characteristics take a significant role in CBR, so it is necessary to keep a watchful eye on DMs' attitude characteristics in case retrieval. On the one hand, the use of attitude characteristics is subjective and highly dependent on users. An optimistic decision-maker and a pessimistic decision-maker tend to make different judgments about the same issue. On the other hand, if description of the target case is accurate and the calculation of similarity is accurate, an optimistic attitude should be adopted. If there is reason to doubt the accuracy of the similarity between cases, a pessimistic attitude should be adopted. Therefore, the attitude characteristics of users can be considered as finding a balance between risks and benefits.

Next, we give an example to illustrate our case retrieval algorithm.

Example 1 Let's have $q = 6$ primary attributes. Local similarity with the 6 attributes between source case and target case is: $C = \{sim_{i1} = 0.7, sim_{i2} = 0.4, sim_{i3} = 0.9, sim_{i4} = 1, sim_{i5} = 0.5, sim_{i6} = 0.8\}$. We can get $\lambda_i(1) = 4, \lambda_i(2) = 3, \lambda_i(3) = 6, \lambda_i(4) = 1, \lambda_i(5) = 5, \lambda_i(6) = 2$. Then, we can compute $Prod_i(j) = \prod_{k=1}^{j} sim_{i\lambda_i(k)}$ and these results are given in Table 1.

The value of α is different for different users, and we can calculate some typical $SIM_{i,\alpha}$. For $q = 6$, $w_j = \left(\frac{j}{6}\right)^{\frac{1-\alpha}{\alpha}} - \left(\frac{j-1}{6}\right)^{\frac{1-\alpha}{\alpha}}$ and $SIM_{i,\alpha} = \sum_{j=1}^{6} w_j Prod_i(j)$.

(1) For an optimistic attitude, $\alpha = 0.8$: $m = \frac{1-\alpha}{\alpha} = 0.25$ and $w_j = \left(\frac{j}{6}\right)^{0.25} - \left(\frac{j-1}{6}\right)^{0.25}$. The results are given in Table 2.

So $SIM_{i,\alpha} = 0.8553$ when $\alpha = 0.8$.

(2) For a neutral attitude, $\alpha = 0.5$: $m = \frac{1-\alpha}{\alpha} = 1$ and $w_j = \left(\frac{j}{6}\right) - \left(\frac{j-1}{6}\right) = \frac{1}{6}$. We can get: $SIM_{i,\alpha} = $

Table 2 The numerical example of $\alpha = 0.8$

	$\left(\frac{j}{6}\right)^{0.25}$	$\left(\frac{j-1}{6}\right)^{0.25}$	w_j	$Prod_i(j)$	$w_j Prod_i(j)$
$j=1$	0.6389	0	0.6389	1	0.6389
$j=2$	0.7598	0.6389	0.1209	0.9	0.1088
$j=3$	0.8409	0.7598	0.0811	0.72	0.0584
$j=4$	0.9036	0.8409	0.0627	0.504	0.0316
$j=5$	0.9554	0.9036	0.0518	0.252	0.0131
$j=6$	1	0.9554	0.0446	0.1008	0.0045
			$\sum_j w_j = 1$		$\sum_j w_j Prod_i(j) = 0.8553$

Table 3 The numerical example of $\alpha = 0.2$

	$\left(\frac{j}{6}\right)^4$	$\left(\frac{j-1}{6}\right)^4$	w_j	$Prod_i(j)$	$w_j Prod_i(j)$
$j=1$	0.0008	0	0.0008	1	0.0008
$j=2$	0.0123	0.0008	0.0116	0.9	0.0104
$j=3$	0.0625	0.0123	0.0502	0.72	0.0361
$j=4$	0.1975	0.0625	0.1350	0.504	0.0681
$j=5$	0.4823	0.1975	0.2847	0.252	0.0718
$j=6$	1	0.4823	0.5177	0.1008	0.0522
			$\sum_j w_j = 1$		$\sum_j w_j Prod_i(j) = 0.2393$

$\frac{1}{6}\sum_{j=1}^{6} Prod_i(j) = \frac{1}{6}(1 + 0.9 + 0.72 + 0.504 + 0.252 + 0.1008) = 0.579$. So $SIM_{i,\alpha} = 0.579$ when $\alpha = 0.5$.

(3) For a pessimistic attitude, $\alpha = 0.2$: $m = \frac{1-\alpha}{\alpha} = 4$ and $w_j = \left(\frac{j}{6}\right)^4 - \left(\frac{j-1}{6}\right)^4$. The results are given in Table 3. So $SIM_{i,\alpha} = 0.2393$ when $\alpha = 0.8$.

We can find from these examples that as α increases, so does $SIM_{i,\alpha}$. We see that the order of C_i basically depends on the order of sim_{ij}.

3.2 SLFs case retrieval algorithm combined with feature similarity

When CBR is carried out, the attributes of target cases and source cases are not necessarily identical (Li et al. 2006), that is, we need to consider the feature similarity (McSherry 2011). To solve the global similarity, both local similarity and feature similarity should be taken into consideration. In case retrieval, feature similarity is represented by different reliability of each attribute. Therefore, the reliability of each attribute should be taken into consideration in the case retrieval algorithm of SLFs.

The reliability of each attribute is represented by $R_{ij} = \{r_{i1}, r_{i2}, \dots r_{iq}\}$, $R_{ij} \epsilon [0, 1]$, and $r_{ij} (j \epsilon 1, 2, \dots, q)$ represents the reliability of attribute j of the historical case i. In a case search, the reliability of each attribute does not change. So in this case, the value of r_{ij} depends only on j, not on i. Next, we give a description of SLFs case retrieval algorithm considering reliability (Yager et al. 2017).

The total reliability is $R_i = \sum_{j=1}^{q} R_{ij}$, and then we use this to obtain the normalized reliability $r_{ij} = \frac{R_{ij}}{R_i}$. Obviously, $\sum_{j=1}^{q} r_{ij} = 1$.

We need to consider the products of the probability and the normalized reliability associated with target case C_i. We define an index function σ_i and $\sigma_i(k)$ is the kth largest index of these products. $sim_{i\sigma_i(k)} \times r_{i\sigma_i(k)}$ is the kth largest of the $sim \times r$, where $sim_{i\sigma_i(k)}$ is the probability corresponding to the kth largest of the $sim \times r$ products and $r_{i\sigma_i(k)}$ is its associated reliability.

The order of local similarity for a certain C_i is depending on the product of compatible probability of the local similarity of each attribute and the reliability of each attribute. Either a small compatible probability or a small reliability can lead to a lower ordering. If reliability of all the attributes is identical, then index $\sigma_i(k)$ depends only on the probabilities. We have

$$Prod_i(j) = \prod_{k=1}^{j} sim_{i\sigma_i(k)} \tag{24}$$

where $Prod_i(j)$ is the product of the first j ordered probabilities and σ_i induces the order.

$$N_{ij} = \sum_{k=1}^{j} r_{i\sigma_i(k)} \tag{25}$$

where N_{ij} is the sum of the normalized reliability associated with the j largest $sim \times r$ products for the target case C_i.

Table 4 Probability reliability

j	Reliability r_{ij}	Probability reliability $sim_{ij} \times r_{ij}$	Index order
1	0.244	0.171	1
2	0.171	0.068	6
3	0.097	0.088	5
4	0.122	0.122	2
5	0.220	0.110	4
6	0.146	0.117	3

Table 5 Probability products

Ordered probability	$Prod_i(j)$
$sim_{i\sigma_i(1)} = sim_{i4} = 0.7$	$Prod_i(1) = 0.7$
$sim_{i\sigma_i(2)} = sim_{i1} = 1$	$Prod_i(2) = 0.7 \times 1 = 0.7$
$sim_{i\sigma_i(3)} = sim_{i6} = 0.8$	$Prod_i(3) = 0.7 \times 0.8 = 0.56$
$sim_{i\sigma_i(4)} = sim_{i5} = 0.5$	$Prod_i(4) = 0.56 \times 0.5 = 0.28$
$sim_{i\sigma_i(5)} = sim_{i3} = 0.9$	$Prod_i(5) = 0.28 \times 0.9 = 0.252$
$sim_{i\sigma_i(6)} = sim_{i2} = 0.4$	$Prod_i(6) = 0.252 \times 0.4 = 0.1008$

We define f(x) as the weight generating function, then for $j = 1 \ldots q$ we calculate the OWA weights:

$$w_{ij} = f(S_{ij}) - f(S_{i(j-i)}) \tag{26}$$

Then, the soft likelihood function of the target case C_i considering reliability is

$$SIM_{i,f} = \sum_{j=1}^{q} w_{ij} Prod_i(j) \tag{27}$$

If the reliability of $r_{i\sigma_i(k)}$ is 0, $S_{ij} = S_{i(j-1)}$ and $w_{ij} = S_{ij} - S_{i(j-1)} = 0$. If all the reliabilities are $r_{ij} = \frac{1}{q}$, $S_{ij} = \frac{j}{q}$ and $w_{ij} = f\left(\frac{j}{q}\right) - f\left(\frac{j-1}{q}\right)$. This is the same situation as not considering reliability.

When $f(x) = x^m$ and $m = \frac{1-\alpha}{\alpha}$, we get $f(x) = x^{\frac{1-\alpha}{\alpha}}$ and the weight is

$$w_{ij} = S_{ij}^{\frac{1-\alpha}{\alpha}} - S_{i(j-1)}^{\frac{1-\alpha}{\alpha}} \tag{28}$$

Next, we give an example to illustrate our case retrieval algorithm.

Example 2 Let's have $q = 6$ primary attributes. Local similarity with the 6 attributes between source case and target case is (the same as Example 1): $C = \{sim_{i1} = 0.7, sim_{i2} = 0.4, sim_{i3} = 0.9, sim_{i4} = 1, sim_{i5} = 0.5, sim_{i6} = 0.8\}$. The associated non-normalized evidence reliability is: $R = \{R_{i1} = 1, R_{i2} = 0.7, R_{i3} = 0.4, R_{i4} = 0.5, R_{i5} = 0.9, R_{i6} = 0.6\}$. The normalized reliability is: $r_{ij} = \frac{R_{ij}}{\sum_{k=1}^{q} R_{ik}} = \frac{R_{ij}}{4.1}$

Table 6 Sum of normalized probabilities

$r_{i\sigma_i(j)}$	N_{ij}
$r_{i\sigma_i(1)} = 0.244$	$N_{i1} = 0.244$
$r_{i\sigma_i(2)} = 0.122$	$N_{i2} = 0.366$
$r_{i\sigma_i(3)} = 0.146$	$N_{i3} = 0.512$
$r_{i\sigma_i(4)} = 0.220$	$N_{i4} = 0.732$
$r_{i\sigma_i(5)} = 0.097$	$N_{i5} = 0.829$
$r_{i\sigma_i(6)} = 0.171$	$N_{i6} = 1$

We calculate the probability reliability products, as given in Table 4.

Then, the index function $\sigma_i(k)$ is: $\{\sigma_i(1) = 1, \sigma_i(2) = 4, \sigma_i(3) = 6, \sigma_i(4) = 5, \sigma_i(5) = 3, \sigma_i(6) = 2\}$.

We can calculate $Prod_i(j) = \prod_{k=1}^{j} sim_{i\sigma_i(k)} = Prod_i(j-1)sim_{i\sigma_i(j)}$ as shown in Table 5.

We can use $N_{ij} = \sum_{k=1}^{j} r_{i\sigma_i(k)} = N_i(j-1) + r_{i\sigma_i(j)}$ and calculate the normalized reliability based on the index σ_i as shown in Table 6.

For different α, we can use $SIM_{i,\alpha} = \sum_{j=1}^{q} w_{ij} Prod_i(j)$ to calculate the $SIM_{i,\alpha}$ with different reliabilities associated with the attribute and $w_{ij} = S_{ij}^{\frac{1-\alpha}{\alpha}} - S_{i(j-1)}^{\frac{1-\alpha}{\alpha}}$. Now, we calculate some typical $SIM_{i,\alpha}$.

(1) For an optimistic attitude, $\alpha = 0.8$: $m = \frac{1-\alpha}{\alpha} = 0.25$. We can get Table 7. So $SIM_{i,\alpha} = 0.617$ when $\alpha = 0.8$.

(2) For a neutral attitude, $\alpha = 0.5$: $m = \frac{1-\alpha}{\alpha} = 1$. We can get Table 8. So $SIM_{i,\alpha} = 0.441$ when $\alpha = 0.5$.

(3) For a pessimistic attitude, $\alpha = 0.2$: $m = \frac{1-\alpha}{\alpha} = 4$. We can get Table 9. So $SIM_{i,\alpha} = 0.202$ when $\alpha = 0.2$.

It can be clearly observed from Table 10 that soft likelihood value increases with the increase in attitude value α.

Table 7 Numerical example of $\alpha = 0.8$

	$N_{ij}^{0.25}$	$N_{i(j-1)}^{0.25}$	w_{ij}	$Prod_i(j)$	$w_{ij}Prod_i(j)$
$j = 1$	0.703	0	0.703	0.7	0.492
$j = 2$	0.778	0.703	0.075	0.7	0.052
$j = 3$	0.846	0.778	0.068	0.56	0.038
$j = 4$	0.925	0.846	0.079	0.28	0.022
$j = 5$	0.954	0.925	0.029	0.252	0.0074
$j = 6$	1	0.954	0.046	0.1008	0.0046
			$\sum_j w_{ij} = 1$		$\sum_j w_{ij}Prod_i(j) = 0.617$

Table 8 Numerical example of $\alpha = 0.5$

	N_{ij}	$N_{i(j-1)}$	w_{ij}	$Prod_i(j)$	$w_{ij}Prod_i(j)$
$j = 1$	0.244	0	0.244	0.7	0.171
$j = 2$	0.366	0.244	0.122	0.7	0.085
$j = 3$	0.512	0.366	0.146	0.56	0.082
$j = 4$	0.732	0.512	0.220	0.28	0.062
$j = 5$	0.829	0.732	0.097	0.252	0.024
$j = 6$	1	0.829	0.171	0.1008	0.017
			$\sum_j w_{ij} = 1$		$\sum_j w_{ij}Prod_i(j) = 0.441$

Table 9 Numerical example of $\alpha = 0.2$

	N_{ij}^4	$N_{i(j-1)}^4$	w_{ij}	$Prod_i(j)$	$w_{ij}Prod_i(j)$
$j = 1$	0.0035	0	0.0035	0.7	0.0025
$j = 2$	0.018	0.0035	0.0144	0.7	0.0101
$j = 3$	0.069	0.018	0.051	0.56	0.028
$j = 4$	0.287	0.069	0.218	0.28	0.061
$j = 5$	0.472	0.287	0.185	0.252	0.047
$j = 6$	1	0.472	0.528	0.1008	0.053
			$\sum_j w_{ij} = 1$		$\sum_j w_{ij}Prod_i(j) = 0.202$

Table 10 As α increases

α	Soft likelihood value
0.2	0.202
0.5	0.441
0.8	0.617

The function representing the attitude characteristic α of the DMs is $\alpha = \int_0^1 f(y)dy$. The closer α is to 1, the more optimistic he/she is; the closer α is to 0, the more pessimistic he/she is, whereas $\alpha = 0.5$ for more general behavior.

Our retrieval strategy is to combine the case retrieval algorithm based on SLFs developed above with KNN, replacing the traditional KNN strategy combined with the ordinary mean algorithm or the weight average method, thereby improving the accuracy of case retrieval.

4 Experimental verification

In this section, the proposed algorithm is simulated experimentally to evaluate the effectiveness of this case retrieval method. We selected 10 classification data sets from UCI resource base for classification experiment. The UCI database is a machine learning database proposed by the University of California Irvine, which has a lot of real data and is a common standard test data set (Arthur and David 2007). Table 11 shows the abbreviations of names, sample size, class number, attribute number, and other information of each data set. Detailed descriptions of each data set are omitted here.

This study dedicates to develop a case retrieval algorithm and applies the proposed CBR-SLFs method to KNN to obtain a new CBR retrieval strategy. For making a fair and detailed comparison, it is able to contrast its performance with traditional retrieval strategies. At present, retrieval strategies generally use an average-based method.

Table 11 General information of used data sets

Data set	Instance	Class	Attribute	Area
Iris	150	3	4	Life
Balance	625	3	4	Social
Survival	306	2	3	Life
Endgame	958	2	9	Game
Blood	748	2	4	Business
Banknote	1372	2	4	Computer
Breast	116	2	9	Life
Vertebral	310	2	6	N/A
User	403	4	5	Computer
Wholesale	440	2	7	Business

The experimental process is as follows. We use a tenfold cross-validation method to divide data set into training set and test set. We use the training set as a case base and every case in the test set as a target case. Based on the case base, we use different case retrieval strategies to calculate solutions for each target case. If calculated result is consistent with the original corresponding solution, we consider that it is effective, otherwise invalid. We use the ratio of the number of cases with valid solutions to the number of elements in the test set to indicate the effectiveness of each retrieval strategy. For each of the test data sets, the above procedure is repeated 100 times and the simple average is recorded.

For the purpose of verifying the effect of case retrieval strategy of CBR-SLFs proposed in this paper on CBR classification accuracy, the following five case retrieval algorithms were used for comparative experiments:

(1)The KNN retrieval strategy based on mean operator is used to investigate the performance of case retrieval, denoted as KNN-Mean;

(2)The KNN retrieval strategy based on trim mean operator is used to investigate the performance of case retrieval, denoted as KNN-Trim;

(3)The KNN retrieval strategy based on weighted average operator is used to investigate the performance of case retrieval, denoted as KNN-Weight;

(4)The KNN retrieval strategy based on SLFs operator proposed in this paper is used to investigate the performance of case retrieval, denoted as KNN-SLFs;

(5)The KNN retrieval strategy based on SLFs operator considering attribute reliability proposed in this paper is used to investigate the performance of case retrieval, denoted as KNN-RESLFs.

Since the reliability of the attribute is not provided in the data set, we use a random method to generate the reliability of the attribute.

For the KNN, we study the case of k values between 5 and 20. As can be seen from Fig.2, the accuracy of retrieval results

with different K values fluctuates slightly, but is basically flat, indicating that the retrieval strategy is insensitive to K. In comparison test, take $k = 11$.

The SLFs involve the DMs' attitude parameter α. Figure3 shows the influence of the value of α from $0 \ldots 1$, that is, the DMs' attitude from negative to positive, on the correctness of the retrieval strategy. It can be seen that the selection of parameters and different data set types will have impact on the retrieval effect, and the value of α needs to be obtained on the basis of the characteristics of the actual decision-maker and the field in which the case is located. In the comparison test, take the DMs' attitude as neutral, i.e., $\alpha = 0.5$.

Table 12 shows the accuracy of these five retrieval strategies in each data set. To more clearly compare the performance of each case retrieval strategy, the average accuracy of each retrieval strategy across all data sets is listed separately to make the results more clear and intuitive. As can be seen from Table 12:

(1)In all data sets, the retrieval strategy trim mean-based algorithm is almost the worst;

(2)The retrieval strategies of KNN-SLFs and KNN-RESLFs are better than other retrieval strategies;

(3)The ranking of average retrieval efficiency based on all data sets can be obtained by various retrieval strategies: KNN-RESLFs $\approx$ KNN-SLFs >KNN-Weight >KNN-Mean >KNN-Trim.

The above analysis can illustrate the superiority of our retrieval strategy suggested in this paper. In the experiment, the performance of the retrieval strategy of KNN-SLFs is very similar to that of KNN-RESLFs. But in practical application, the reliability degree of each attribute is not random, but according to the importance of the attribute itself or given by experts. The accuracy of KNN retrieval strategy based on SLFs operator considering attribute reliability may be higher in practical application.

5 Conclusion

We introduce the SLFs based on OWA operator into CBR and propose a retrieval strategy based on CBR process. It can reduce the interference of small probability events and consider the attitudinal characteristics of DMs, which is more with the actual decision-making process. We mainly present a method to define global similarity for retrieving the most similar case to target case. Global similarity includes local similarity and feature similarity. Similarity between variables under feature type is represented by local similarity, and the similarity between features is represented by feature similarity. CBR-SLFs are used to aggregate local similarity and feature similarity to obtain the global similarity between the
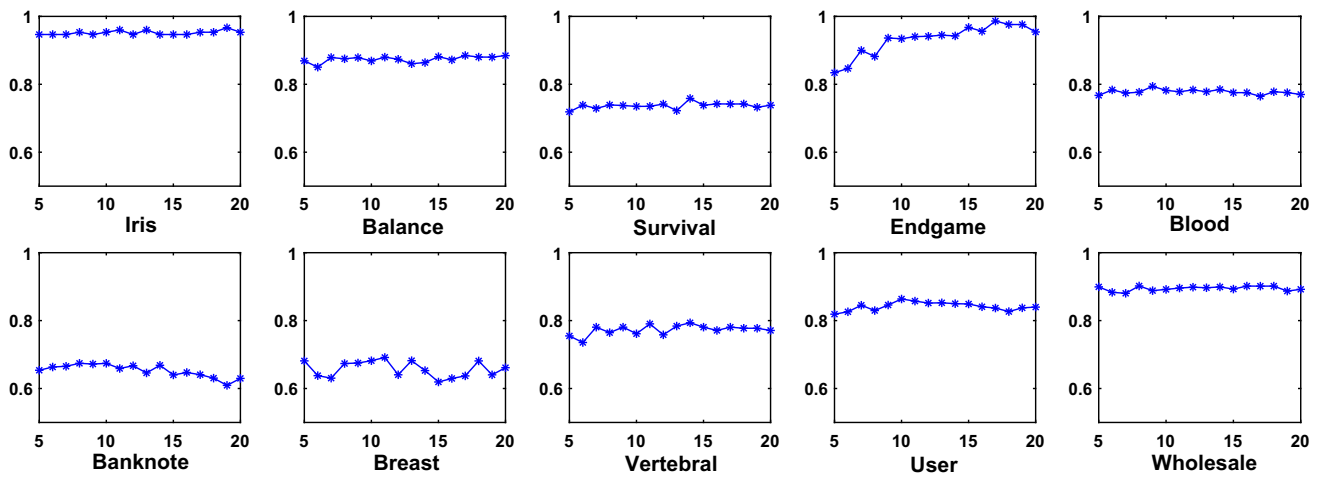
Fig. 2 Performance of the retrieval strategy with KNN-RESLF algorithm under different K
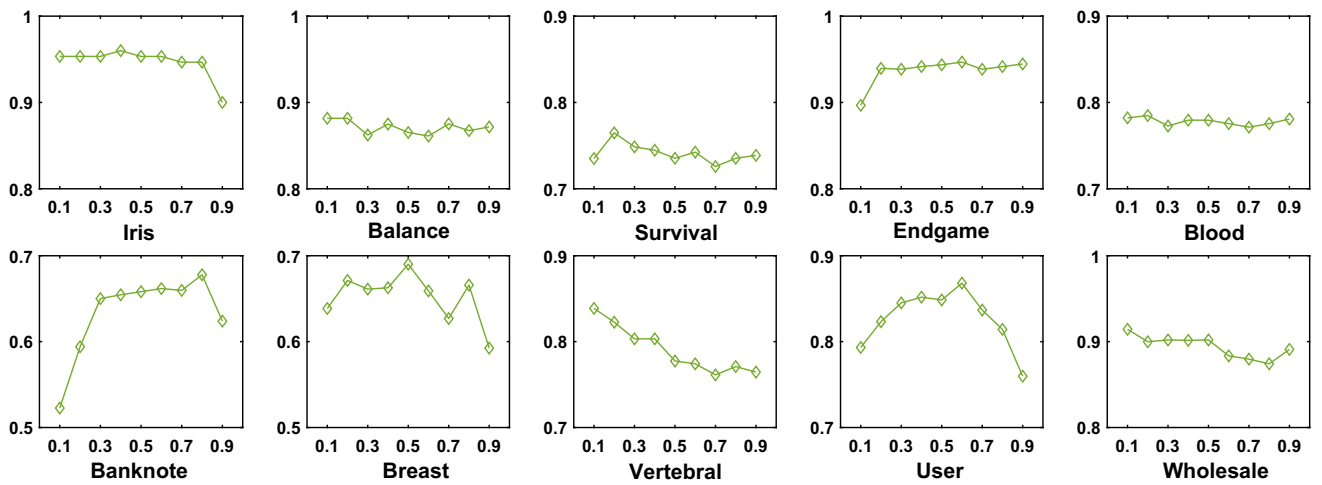

Fig. 3 Performance of the retrieval strategy with KNN-RESLF algorithm under different α

Table 12 Performance of CBR with different retrieval strategies	Data set	Knn-Mean	Knn-Trim	KNN-Weight	Knn-SLFs	Knn-RESLFs
	Iris	0.9545	0.9555	0.9541	0.9548	0.9526
	Balance	0.8104	0.7625	0.8313	0.8716	0.8725
	Survival	0.7273	0.7264	0.7334	0.7417	0.7429
	Endgame	0.8907	0.8584	0.9015	0.9373	0.9417
	Blood	0.7579	0.7589	0.7579	0.7753	0.7757
	Banknote	0.5698	0.6024	0.5728	0.6505	0.6507
	Breast	0.6634	0.6760	0.6798	0.6781	0.6726
	Vertebral	0.8126	0.7836	0.8086	0.8242	0.8284
	User	0.8258	0.8282	0.8234	0.8457	0.8462
	Wholesale	0.8634	0.8702	0.9037	0.8967	0.8934
	Average	0.7876	0.7822	0.7967	0.8176	0.8177

cases. Experimental results on real data sets show that the retrieval strategy proposed by us is superior to the traditional KNN method.

However, this paper also has some limitations: the method of this study is only put forward from the theoretical level and lacks practical application. Moreover, in the experimental verification of this paper, the reliability degree of attributes is generated by random method, which is very brief. In practice, this step is usually completed by decision-makers or experts.

In the future research, the CBR-SLFs retrieval strategy will be further improved. Firstly, the theoretical and experimental studies on the relevant parameters of the algorithm can be further improved to improve the adaptability and reliability of the method. Secondly, limited kinds of attribute types were included in this study. Given various data types may exist in the actual CBR process, further research can explore richer feature types. Next, the attributes of a case are not completely unrelated. We can combine the characteristics of specific research problems to study the interaction between attributes. And in the future, CBR can be applied to solve complicated problems in practice, for instance, disease diagnosis, image recognition, and so on.

Declarations

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

References

Aamodt A, Plaza E (1994) Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Commun 7(1):39–59

Abualigah LMQ et al (2019) Feature selection and enhanced krill herd algorithm for text document clustering. Springer, Heidelberg

Abualigah LMQ, Hanandeh ES (2015) Applying genetic algorithms to information retrieval using vector space model. Int J Comput Sci Eng Appl 5(1):19

Abualigah LM, Khader AT (2017) Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering. J Supercomput 73(11):4773–4795

Alshammari G, Jorro-Aragoneses JL, Kapetanakis S, Petridis M, Recio-García JA, Díaz-Agudo B (2017) A hybrid cbr approach for the long tail problem in recommender systems. International Conference on Case-Based Reasoning. Springer, Heidelberg, pp 35–45

Armengol E, Palaudaries A, Plaza E (2001) Individual prognosis of diabetes long-term risks: a cbr approach. Methods of Information in Medicine-Methodik der Information in der Medizin 40(1):46–51

Arthur A, David N (2007) University of california irvine machine learning repository, http://archive.ics.uci.edu/ml/index.php, Accessed

Begum S, Ahmed MU, Funk P, Xiong N, Folke M (2010) Case-based reasoning systems in the health sciences: a survey of recent trends and developments. IEEE Trans Syst Man Cybern Part C (Appl Rev) 41(4):421–434

Bergmann R, Kolodner J, Plaza E (2005) Representation in case-based reasoning. Knowl Eng Rev 20(3):209–214

Catalá L, Julián V, Gil-Gómez JA (2014) A CBR-based game recommender for rehabilitation videogames in social networks, in: International conference on intelligent data engineering and automated learning, Springer, pp 370–377

Cover T, Hart P (1967) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13(1):21–27

Cunningham P (2008) A taxonomy of similarity mechanisms for case-based reasoning. IEEE Trans Knowl Data Eng 21(11):1532–1543

De Mantaras RL, McSherry D, Bridge D, Leake D, Smyth B, Craw S, Faltings B, Maher ML, T COX MI, Forbus K (2005) Retrieval, reuse, revision and retention in case-based reasoning. Knowl Eng Rev 20(3):215–240

El-Sappagh S, Elmogy M, Ali F, Kwak K-S (2019) A case-base fuzzification process: diabetes diagnosis case study. Soft Comput 23(14):5815–5834

Fei L, Deng Y (2020) Multi-criteria decision making in pythagorean fuzzy environment. Appl Intell 50(2):537–561

Fei L, Feng Y (2021) A dynamic framework of multi-attribute decision making under pythagorean fuzzy environment by using dempster-shafer theory. Eng Appl Artif Intell 101:104213

Fei L, Feng Y, Wang H (2021) Modeling heterogeneous multi-attribute emergency decision-making with dempster-shafer theory. Comput Ind Eng 161:107633

Floyd MW, Drinkwater M, Aha DW (2015) Trust-guided behavior adaptation using case-based reasoning. Tech. rep, Naval Research Laboratory Washington United States

Georgopoulos VC, Stylios CD (2008) Complementary case-based reasoning and competitive fuzzy cognitive maps for advanced medical decisions. Soft Comput 12(2):191–199

González-Briones A, Prieto J, De La Prieta F, Herrera-Viedma E, Corchado JM (2018) Energy optimization using a case-based reasoning strategy. Sensors 18(3):865

Greene D, Freyne J, Smyth B, Cunningham P (2010) An analysis of current trends in CBR research using multi-view clustering. AI Mag 31(2):45–45

Guo X, Yuan J, Li Y (2014) Feature space k nearest neighbor based batch process monitoring. Acta Autom Sin 40(1):135–142

Holt A, Bichindaritz I, Schmidt R, Perner P (2005) Medical applications in case-based reasoning. Knowl Eng Rev 20(3):289–292

Homem TPD, Santos PE, Costa AHR, da Costa Bianchi RA, de Mantaras RL (2020) Qualitative case-based reasoning and learning. Artif Intell 283:103258

Hong T, Koo C, Kim D, Lee M, Kim J (2015) An estimation methodology for the dynamic operational rating of a new residential building using the advanced case-based reasoning and stochastic approaches. Appl Energy 150:308–322

Jiang Z, Jiang Y, Wang Y, Zhang H, Cao H, Tian G (2019) A hybrid approach of rough set and case-based reasoning to remanufacturing process planning. J Intell Manuf 30(1):19–32

Jian C, Zhe T, Zhenxing L (2015) A review and analysis of case-based reasoning research, in: 2015 International conference on intelligent transportation, big data and smart city, IEEE, pp 51–55

Kang Y-B, Krishnaswamy S, Zaslavsky A (2013) A retrieval strategy for case-based reasoning using similarity and association knowledge. IEEE Trans Cybern 44(4):473–487

Kolodner JL, Simpson RL (1989) The mediator: analysis of an early case-based problem solver. Cogn Sci 13(4):507–549

Kwon N, Song K, Ahn Y, Park M, Jang Y (2020) Maintenance cost prediction for aging residential buildings based on case-based reasoning and genetic algorithm. J Build Eng 28:101006

A hybrid retrieval strategy for case-based reasoning using soft...

3501

Le Ber F, Lieber J, Benoit M (2018) Case-based reasoning for forecasting the allocation of perennial biomass crops. ERCIM News 113(113):34–35

Le DV-K, Chen Z, Wong YW, Isa D (2020) A complete online-svm pipeline for case-based reasoning system: a study on pipe defect detection system. Soft Comput 24:16917–16933

Li H, Sun J, Sun B-L (2009) Financial distress prediction based on or-CBR in the principle of k-nearest neighbors. Expert Syst Appl 36(1):643–659

Liang C, Gu D, Bichindaritz I, Li X, Zuo C, Cheng W (2012) Integrating gray system theory and logistic regression into case-based reasoning for safety assessment of thermal power plants. Expert Syst Appl 39(5):5154–5167

Lin S-W, Chen S-C (2011) Parameter tuning, feature selection and weight assignment of features for case-based reasoning by artificial immune system. Appl Soft Comput 11(8):5042–5052

Li Y, Shiu SC, Pal SK (2006) Combining feature reduction and case selection in building CBR classifiers. IEEE Trans Knowl Data Eng 18(3):415–429

Liu W, Tan R, Cao G, Yu F, Li H (2019) Creative design through knowledge clustering and case-based reasoning. Eng Comput 36:527

Marcos-Pablos S, García-Peñalvo FJ (2020) Information retrieval methodology for aiding scientific database search. Soft Comput 24(8):5551–5560

McSherry D (2011) Conversational case-based reasoning in medical decision making. Artif Intell Med 52(2):59–66

Mi X, Tian Y, Kang B (2021) A hybrid multi-criteria decision making approach for assessing health-care waste management technologies based on soft likelihood function and D-numbers, Appl Intell 1–20

Müller G, Bergmann R (2015) Learning and applying adaptation operators in process-oriented case-based reasoning, In: International conference on case-based reasoning, Springer, pp. 259–274

Navarro M, Heras S, Julián V, Botti V (2011) Incorporating temporal-bounded CBR techniques in real-time agents. Expert Syst Appl 38(3):2783–2796

Nikpour H, Aamodt A (2021) Fault diagnosis under uncertain situations within a bayesian knowledge-intensive cbr system, Progress in Artif Intell 1–14

Petrovic S, Khussainova G, Jagannathan R (2016) Knowledge-light adaptation approaches in case-based reasoning for radiotherapy treatment planning. Artif Intell Med 68:17–28

Pinto T, Faia R, Navarro-Caceres M, Santos G, Corchado JM, Vale Z (2018) Multi-agent-based cbr recommender system for intelligent energy management in buildings. IEEE Syst J 13(1):1084–1095

Rallabandi VS, Sett S (2008) Knowledge-based image retrieval system. Knowl-Based Syst 21(2):89–100

Ramos-González J, López-Sánchez D, Castellanos-Garzón JA, de Paz JF, Corchado JM (2017) A CBR framework with gradient boosting based feature selection for lung cancer subtype classification. Comput Biol Med 86:98–106

Schank RC (1983) Dynamic memory: a theory of reminding and learning in computers and people. Cambridge University Press, Cambridge

Schmidt R, Gierl L (2005) A prognostic model for temporal courses that combines temporal abstraction and case-based reasoning. Int J Med Inform 74(2–4):307–315

Schmidt R, Montani S, Bellazzi R, Portinale L, Gierl L (2001) Cased-based reasoning for medical knowledge-based systems. Int J Med Inform 64(2–3):355–367

Simon HA (1955) A behavioral model of rational choice. Q J Econ 69(1):99–118

Tan R-P, Zhang W-D, Chen S-Q, Yang L-H (2020) Emergency decision-making method based on case-based reasoning in heterogeneous information environment. Control Dec 35:1966–1976

Tian Y, Mi X, Liu L, Kang B (2020) A new soft likelihood function based on d numbers in handling uncertain information. Int J Fuzzy Syst 22(7):2333–2349

Torrent-Fontbona F, Massana J, López B (2019) Case-base maintenance of a personalised and adaptive CBR bolus insulin recommender system for type 1 diabetes. Expert Syst Appl 121:338–346

Wan S-P, Xu J, Dong J-Y (2016) Aggregating decision information into interval-valued intuitionistic fuzzy numbers for heterogeneous multi-attribute group decision making. Knowl-Based Syst 113:155–170

Xu L, Huang C, Niu J, Li C, Wang J, Liu H, Wang X (2021) An improved case-based reasoning method and its application to predict machining performance. Soft Comput 25(7):5683–5697

Yager RR (1988) On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE Trans Syst Man Cybern 18(1):183–190

Yager RR (1996) Quantifier guided aggregation using owa operators. Int J Intell Syst 11(1):49–73

Yager RR, Elmore P, Petry F (2017) Soft likelihood functions in combining evidence. Inform Fus 36:185–190

Yahong P, Xiuli G (2018) A hybrid multiple attributes group decision making method based on vikor. Mach Design Res 34(1):177–182

Yu G, Li D, Ye Y, Qiu J (2017) Heterogeneous multi-attribute variable weight decision-making method considering regret avoidance. Comput Integr Manuf Syst (Chin) 23:154–161