The Research on General Case-Based Reasoning Method Based on TF-IDF

Lin Zhang
Institute of Robot Engineering
Anhui Sanlian University
Anhui, China
zhanglin9292@163.com,sirenrabbit@sina.com

***Abstract*—With the continuous expansion of the application field of Case-Based Reasoning (CBR) technology, it is increasingly difficult for programmers to acquire and express professional knowledge. Therefore, the demand for general case retrieval model based on Case-Based Reasoning is rising. This paper first gives a structured expression of professional knowledge, and combines the Case-Based Reasoning method with the scientific measurement of keyword weight (Term Frequency-Inverse Document Frequency, TF-IDF) to design the case organization, case retrieval and case retaining in CBR technology. It provides an effective method for general case retrieval model.**

Keywords—Case-based reasoning, retrieval model, text frequency

I. INTRODUCTION

In recent years, with the rapid development of computer science and Information Technology, people find it difficult to obtain knowledge by using traditional Rule-based reasoning (RBR) system in the face of disorganization, information and many experiences, service difficulties, such as: rules are difficult to obtain, professional knowledge is difficult to express clearly, programming, people can not clearly understand the problem, etc. . In this Case, Yale University professor Roger Schank proposed a Case-Based Reasoning (CBR) method in his book Dynamic Memory in 1982, which uses Case-Based Reasoning (CBR) to replace rules, with the rapid development of artificial intelligence, the case-based reasoning (CBR) technique has been paid more and more attention, and the problem of obtaining knowledge from RBR has been avoided, and has been widely used in various fields.

With the development of the research of CBR and the application field of CBR, it has been involved in machine fault diagnosis, medicine/medical diagnosis, business consulting and decision-making, legal case assessment and weather forecast etc. As a result, it is difficult for programmer to obtain and express professional knowledge, which is increasingly prominent problem. So, the demand of the general case retrieval model based on case-based reasoning is rising.

Case-based Reasoning is a branch of artificial intelligence, and is a kind of reasoning method of artificial intelligence based on empirical knowledge [1-9]. CBR structure and form a putted forward with rich subject, adopts case based reasoning in the problem solving mechanism of analogy in the process of strategy and imitating human decision understanding way, effectively solve the problem of unstructured, poor knowledge [10].

II. THE GENERAL CASE RETRIEVAL MODEL FRAMEWORK BASED ON CASE-BASED REASONING

Through comparing the Case-Based Reasoning technology which used in different areas, researchers put forward the general case retrieval model framework based on Case-Based Reasoning, which is shown as follows in Figure 1:

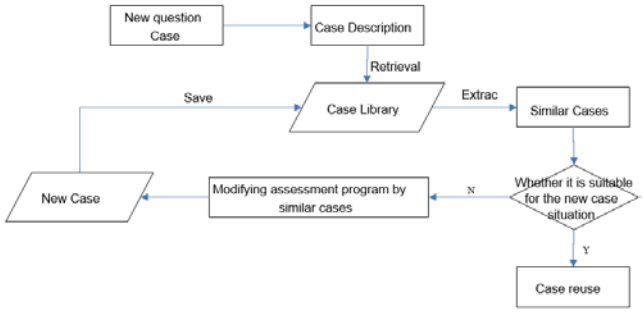

Fig. 1. The general case retrieval model framework

In the general case retrieval method based on case-based reasoning, we first give the description of different attribute criteria according to the characteristics of each attribute of the case, and then express these attributes in the form of feature vector, finally, the matching cases which are similar to the new problem are searched in the case base. If we find the same or similar cases within the margin of error allowed by the threshold, the knowledge of the old cases will be reused directly. Otherwise, the assessment process will be modified from the most similar cases to form new cases and saved in the corresponding case base.

III. FOUR KEY TECHNOLOGIES OF GENERAL CASE RETRIEVAL MODEL FRAMEWORK BASED ON CASE-BASED RESONING

A. Case Knowledge Indication

With the development of social economy, the research of CBR and the rapid increase of amount of data, the application field of CBR is more and more widely. But before using CBR, we should clean and clear up data at first. From all walks of life in our country every institution has available data. But due to differences in local, many data is scattered in time and space. In addition, there are differences in storage structure, evaluation content and properties characteristics. So a lot of data are difficult to compare on the same platform.

Here, we use the Boolean characteristic vector to represent case knowledge. Because different areas have different assess priorities, and most data are unstructured. Therefore, we first establish an attribute statistics, which is to integrate the various agencies in some area of evaluation index and break it down for each option. And establish the statistics of the attributes are shown in table I below

TABLE I ATTRIBUTE STATISTICS INDICATORS TABLE

Attribute number	Attribute content	Attribute values
1	Attribute 1	0
2	Attribute 2	1
3	Attribute 3	0
……	……	……
i	Attribute i	1
……	……	……
j	Attribute j	0
……	……	……
n	Attribute n	1

Similar to the concept of Inverse Document Frequency(IDF), the more frequently a word appears in all articles, such as "is" , the less it should weigh in a search, and vice versa, the less frequently a word appears in all articles, such as some technical terms, the more weight it should have in the search. By the same token, if an attribute appears less frequently in all cases, this means that the attribute is typical in the evaluation of the case. Therefore, in case retrieval, its weight should be larger. On the contrary, if an attribute appears more frequently in all cases, it is difficult to judge the attribution of cases by this attribute, then the weight in case retrieval should be relatively small. Therefore, it is not reasonable to set all the indicators that appear in the case to 1. Therefore, we can use the method of information entropy to set different weights for each attribute. Assuming that the attribute Ai appears ni times in n cases in the case base, the AI's weight is set to log (n /ni).

If there are 800 cases in the case base, 400 cases have the attribute ni and 200 cases have the attribute nj, then n=800, ni=400 and nj=200. Therefore, the weight of the attribute ni is log(n/ni)=log(800/400)=log(2)=1, and the weight of the attribute nj is log(n/ni)=log(800/200)=log(4)=2. It is easy to count the number of times each attribute appears in all cases in the case base, so we can get the weight of all attributes. We can then set up a table of weighted attribute statistics according to the different weights of the attributes as shown in Table II:

From this we can get the weight vectors $b=(b1,b2,...,bn)^T=(9.12,0.86，...，6.34)^T$.

TABLE II WEIGHTED ATTRIBUTE STATISTICS INDICATORS TABLE

Attribute number	Attribute content	Attribute values	Weight
1	Attribute 1	0	9.12
2	Attribute 2	1	0.86
3	Attribute 3	0	
……	……	……	……
i	Attribute i	1	1
……	……	……	……
j	Attribute j	0	6.69
……	……	……	……
n	Attribute n	1	6.34

Although IDF shows a good application effect in the distribution of information feature value weights, the introduction of IDF is intended to suppress the negative influence of meaningless high frequency words in a document, but when the ratio of total document to keyword contained document is higher, the low frequency words will be highlighted. There is a question here that deserves to be discussed: common words are not equal to meaningless words, such as some public figures, hot events, etc. Similarly, the occasional appearance of low-frequency words will be treated as high-weight keywords, this transition enlarges the importance of rare words. In view of the inadequacy of IDF, mainly the frequency of the occurrence of the i-th keyword in different classes will directly affect whether the keyword can become the characteristic word of the document. T Therefore, an item between the original document classes can be added to indicate the distribution of feature words among the classes, that is, the inter-class dispersion of the feature word distribution. By the same token, we can add the same inter-class dispersion in the case-based reasoning process.

Inter-class dispersion is a description of the distribution of feature attributes in different categories of cases. Features that are concentrated in a certain type of case often have strong class distinguishing ability. Therefore, the feature attribute has strong inter-class dispersion. Suppose all cases can be divided into n categories, f(i) represents the frequency of occurrence of feature attribute i in a certain type of case, and $\overline{f(i)}$ represents the average frequency of feature i appearing in all types of cases. Therefore:

$$\overline{f(i)} = \frac{1}{n}\sum_{k=1}^{n} f_k(i) \tag{1}$$

And the overall inter-class dispersion is:

$$D(i) = \frac{\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(f_k(i) - \overline{f(i)})^2}}{\overline{f(i)}} \tag{2}$$

Substituting (1) into (2):

$$D(i) = \frac{\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(f_k(i) - \frac{1}{n}\sum_{k=1}^{n} f_k(i))^2}}{\frac{1}{n}\sum_{k=1}^{n} f_k(i)} \tag{3}$$

Combined with the original main idea of IDF, if the feature attribute in formula (3) appears only in a certain type of case, then D(i) is 1, and the term has the strongest classification ability. And if the frequency of occurrence in the case of each class is equal, the term is considered to have no classification ability, and thus D(i) is 0. This feature is useless and can be discarded. It can be seen that the value of D(i) is between [0, 1]. The IDF algorithm variant that considers the inter-class dispersion is:

$$W_i = IDF * D(t) = (\log\frac{D}{D_i}) * (\frac{\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(f_k(i) - \frac{1}{n}\sum_{k=1}^{n} f_k(i))^2}}{\frac{1}{n}\sum_{k=1}^{n} f_k(i)}) \tag{4}$$

However, although the degree of inter-class dispersion is considered here, if two feature attributes are basically the same in the same case, we still cannot accurately determine

the distribution of these two features. Therefore, we define the information entropy in similar cases, so that we can reflect the distribution of feature attributes in similar cases. For example, the more uniform the distribution of a feature attribute i in a similar case, the larger the information entropy in the case, the more the feature attribute i can reflect the feature information of the case, and the calculation formula of the case information entropy within the class is:

$$E(t,C_k) = -\sum_{j}^{D} \frac{Nd_j}{NC_k} \lg \frac{Nd_j}{NC_k} \tag{5}$$

Where Ndj represents the frequency at which the j-th value (0 or 1) of the feature attribute i in the Ck class case occurs, and NCk represents the total frequency at which the feature attribute i appears in the Ck class case.

Finally, the integrated inter-class dispersion and intra-class information entropy, the improved IDF algorithm is used to calculate the case-class distinction, so that the feature attributes can be determined relatively accurately. So there are:

$$\begin{aligned} W_i &= IDF * D(i) * E(t,C_k) \\ &= (\log \frac{D}{D_i}) * (\frac{\sqrt{\frac{1}{n-1}\sum_{k=1}^{n}(f_k(i) - \frac{1}{n}\sum_{k=1}^{n} f_k(i))^2}}{\frac{1}{n}\sum_{k=1}^{n} f_k(i)}) * (-\sum_{j}^{D} \frac{Nd_j}{NC_k} \lg \frac{Nd_j}{NC_k}) \end{aligned} \tag{6}$$

According to formula (6), the improved IDF algorithm is used to select the feature attributes, the weight of each feature attribute is calculated, and then the N cases with the largest weight are selected as the feature vector of case reasoning.

B. Case Retrieval

The core of Case-Based Reasoning is case retrieval, which aims to find cases with reference value and as few as possible from a large number of cases. The commonly used case retrieval strategies include knowledge guiding strategy, template retrieval strategy, nearest neighbor strategy and inductive index strategy. Since we have previously represented cases as weighted vectors of attributes, the similarity between new cases and existing cases can be calculated using a simple law of cosines. Since the weights of all attributes are positive, the cosine of the attribute's weighted vector is between 1 and 0. If the cosine value between the weighted vectors of two cases is 1, then the two cases are identical. If the cosine value between the weighted vectors of two cases is 0, then the two cases are completely different. Therefore, we only need to calculate the cosine value between the existing case and the weighted vector of the attribute of the new case and sort it, then we can find the most similar case between the existing case and the new case, that is, the cosine value between the weighted vector and the new case attribute.

As we know the cosine of $\angle A$ is: $\cos A = \frac{b^2 + c^2 - a^2}{2bc}$

At this time, if b and c are two vectors starting from A, the above formula can be equivalent to: $\cos A = \frac{<b,c>}{|b| \cdot |c|}$, among which <b,c> represents the inner product of vectors, and |b| and |c| represents the length of vector.

Suppose the attribute vector of case X is (x1, x2, ... , xn), where any xi is Boolean, and the value of xi is “true” or “1”, which means the i-th attribute value of case x exists, otherwise it does not exist. The weighted attribute vector is $Y=(y1, y2, ... , yn)^T$, so the case's weighted attribute vector is $(x1, x2, ..., xn) \times (y1, y2, ..., yn)^T = (x1y1, x2y2, ..., xnyn)$.

Therefore, suppose that the weighted attribute vectors A and B of the two cases are (a1, a2, ... , an) and (b1, b2, ... , bn) respectively, well, the cosine of the angle between A and B is:

$$\cos\theta = \frac{a_1b_1 + a_2b_2 + \cdots + a_nb_n}{\sqrt{a_1^2 + a_2^2 + a_n^2} \cdot \sqrt{b_1^2 + b_2^2 + b_n^2}}$$

That is, the greater the cosa value of the two vectors, the greater the similarity of the vectors, and vice versa, the smaller the cosa value, the smaller the similarity of the two vectors.

C. Case reuse / Case modification

When a new case appears, we only need to calculate the cosine value between each case in the case base and the weighted attribute vector of the new case. If the cosine value is 1, then the current case and the new case are exactly the same, that the current case can be reused directly. If there is no case in the case base that has a cosine value of 1 between the weighted attribute vector of the new case, then according to the result of the computed cosine value, some cases that are closest to the new case are selected, or the error value in the threshold range of all cases, and according to similar cases to give a new case processing program. Because case-based reasoning often involves some professional knowledge, it can't be modified automatically by computer, so it needs some artificial intervention.

D. Case Study

The result of a new case after artificial intervention is not necessarily correct, this needs to be verified, as long as the new case is proved to be correct can be added to the case base. Of course, after adding a new case to the case base, the weight vector of the attribute needs to be adjusted, and the weight value of each attribute should be log(n'/ni'). Where n' is the total number of cases in the new case base, that is, n+1. The number of times the i-th attribute appears in the new case, ni' is also adjusted according to the attribute data in the new case. If the i-th attribute value in the new case is "true" or 1, then ni'=ni + 1, otherwise, ni=ni.

IV. METHOD TEST

In order to verify the accuracy of the general case retrieval model based on Case-Based Reasoning put forward in this paper, Visual Studio 2010 is used as the development platform, C# as the development language, and the cases are saved in the SQL Server 2000 to test the accuracy of the method in a simple way.

In the process of test, we used 240 cases of elderly health assessment files which are researched from the community hospital in developing economic district in HeFei in AnHui province for test. In the all 240 cases, 200 cases are used for training and 40 cases are uesd for test.

In the test, firstly, according to the researched data to establish elderly healthy assessment table show in table III:

TABLE III: HEALTH ASSESSMENT FORM FOR THE ELDERLY

1.Marital status	(1)Single (2)Married (3)Remarriage (4)Widowed (5)Other
2.Residence type	(1)Alone (2)Spouses together (3)Children together (4)Spouse and children together (5)Other (specify)
3.Housing type	(1)Building (floor) (2)Lift: a. Yes b. No (3)Bungalow (4)Other
……	……
23.Body index	(1)High (2) Higher (3)Normal (4)Lower (5)Low
24.Blood pressure	(1)High (2) Higher (3)Normal (4)Lower (5)Low

And then, according to the training cases and the corresponding evaluation standard, the SQL Server 2000 is used to establish a healthy statistics indicators table which contains 576 attributes (each attribute for a healthy index) and 240 tuples (each tuple is a feature vector Xi) , which is shown in table IV:

In table IV, the type of each healthy indicators in healthy statistics indicators table is Boolean . At last, we establish the healthy statistics indicators table, the former 200 tuple(training cases) are used as the foundation to calculate the weight of each health indicators in the table, and then a one-dimensional array containing 576 elements is established to hold the weight.

After the preparations all above, we use the later 20 tuple as the test cases, and calculate each vector Angle with the first 100 tuple respectively, and then screen out the cases which meet the threshold. The test shows that the case retrieval accuracy is more than 90%.

TABLE IV: HEALTH STATISTICS INDICATORS TABLE

Index number	Index content
1	Single
2	Married
……	……
6	Live alone
7	Live with spouses together
……	……
117	Skin dry
118	Skin rash
……	……

TABLE V: WEIGHTED HEALTHY STATISTICS INDICATORS TABLE

Index number	Index content	Weight
1	Single	9.12
2	Married	0.86
……	……	……
6	Live alone	6.34
7	Live with spouses together	1.76
……	……	……
117	Skin dry	1
118	Skin rash	6.96
……	……	……

V. CONCLUSION

This paper put forward the general case retrieval model based on Case-Based Reasoning, and presents several key technologies in the process of reasoning. As the general model, it only needs to transfer case attributes in the application process, and then correspond it to the relevant properties, thus we can use the model of Case-Based Reasoning. For example, in testing process, we correspond health indicators to the attribute, and transform the general model for the elderly health assessment model. In addition, the establishment method of general case retrieval mode in this paper is easy to understand and is relatively simple to achieve. Through experimental verification, the accuracy is also relatively high. However, the test just verifies the accuracy of the method. As for its effectiveness in practical system applications, it has still been verified, and the key technologies are still have some problems to be solved.

First, the attributes of each case are decomposed into Boolean type attribute vectors. This method is simple and suitable for most cases and their attributes. However, it does not apply to certain reference data such as the text type of non-standard statements. In this case, the applicability is not very good.

Secondly, when knowledge is represented by feature vector, the feature vector of an attribute is actually a sparse vector because an index can be decomposed into many items and only one item can be selected under the same condition, how to simplify the sparse vector algorithm, at the same time, to ensure its effectiveness and improve its efficiency is one of the direction of future research.

In addition, the thresholds mentioned in case reuse techniques need to be set by professionals, which will undoubtedly increase the level of human intervention. Therefore, in the practical application, how to reduce the degree of human intervention, improve its efficiency, are the direction of future research.

REFERENCES

[1] M. Han, L.H. Shen, "Case-based Resoning Based on FCM and Neural Network", Control and Decision, vol. 27, pp. 1421-1424, September 2012.

[2] J.Q Li, X.G. Li, D.X. Gu, S. Feng, "Case Based Reasoning ISP Knowledge Reuse Method", Computer Engineering, vol. 36, pp. 36-39, January 2010.

[3] M.D Peng, T. Peng, "Application of Case-based Resoning in TCM Case Record Distribution System", The World Science and Technology—Modernization of Traditional Chinese Medicine, vol. 11, pp. 698-701, May 2009.

[4] D.X. Gu, X.G. Li, Research of Ca se-ba sed Informa t ion System Bus iness Process Knowledge Reuse", Journal of Chinese Computer Systems, vol. 28, pp. 1439-1443, August 2007.

[5] Y.C Shen, Z.M. Shu, "Research of Case Representation and System Architecture Based on CBR", Journa of Southern Medical University, vol. 27, pp. 1114-1116, July 2007.

[6] Li Jian-yang, Zheng Han-yuan, Liu Hui-ting, "Case-based Reasoning Based on Multi-layered Feedforward Neural Network", Computer Engineering, vol. 32, pp. 188-190, July 2006.

[7] R. Li, H.T. Ren, K.Y. Liu, "The Study on Feature Weight Auto-learning Method for Case-based Reasoning", Journal of Shanxi University, vol. 27, pp. 245-248, March 2004.

[8] R. Li, H.T. Ren, K.Y. Liu, J.Y Liang, "The Application of CBR in Agriculture Expert System", Computer Engineering and Application, vol. 25, pp. 196-198,204, January 2004.

[9] J.H. Zhang, Z.Y. Liu, "Case-based Reasoning and Rule-based Reasoning for Emergency Preparedness Information System", Journal of Tongji University, vol. 30, pp. 890-894, July 2002.

[10] L.H. Jiang, B. Liu, "On the Application of Case-based Reasoning to Intelligent Forecasting Support System", Journal of Decision Marking and Decision Support Systems, vol. 6, pp. 63-69, June 1996.

[11] J. Wu, Beauty of Mathematics, BeiJing: People's Posts and Telecommunications Publishing, 2012.