

# Acquisition of Knowledge in the Form of Fuzzy Rules for Cases Classification

Tatiana Avdeenko<sup>(✉)</sup> and Ekaterina Makarova

Novosibirsk State Technical University, Novosibirsk, Russia  
avdeenko@corp.nstu.ru, katmc@yandex.ru

**Abstract.** We consider an approach to automatic knowledge acquisition through machine learning based on integration of two basic paradigms of reasoning – case-based and rule-based reasoning. Case-based reasoning allows to use high-performance database technology for storing and accumulating cases, while rule-based reasoning is the most developed technology for creating declarative knowledge on the basis of strong logical inference. We also propose an improvement of classification algorithm through extraction of fuzzy rules from cases. We have obtained higher classification accuracy for various membership functions and for sequentially reducing amount of training sample by application of special strategies for expanding the scope of fuzzy rules in the control sample classification.

**Keywords:** Knowledge-based systems · Case-based reasoning · Rule-based reasoning · Fuzzy rules · Classification

## 1 Introduction

An important stage in building knowledge-based systems (KBS) is the choice of knowledge representation model for creating knowledge base. The first KBS appeared in 1970s were based on rules. When implementing rule-based KBS, developers faced many challenges. The major one is the problem of knowledge elicitation and formulating it in the form of a set of rules. Most often, the experts intuitively make decisions based on their vast experience, without hesitation, what rules they apply in this or that case. Partitioning a specific behavior of an expert into separate building blocks such as rules is a very complicated problem requiring high skilled specialists. Thus, acquisition of knowledge is the key problem of rule-based reasoning (RBR) systems.

However, since 1980s an alternative reasoning paradigm has increasingly attracted more and more attention. Case-based reasoning (CBR) solves new problems by adapting previously successful solutions to similar problems, just as a human does it. The paper [1] of Shank is widely cited to be the origins of CBR. In this work it has been proposed to generalize knowledge about previous situations and store them in the form of scripts that can be used to make conclusions under similar conditions. Later Shank continued to explore the role that dynamic memory about previous situations (cases), represented as a knowledge container, plays both in the process of decision making and in the process of learning [2]. The model of dynamic memory became a basis for the creation of a

number of other systems: MEDIATOR [3], CHEF [4], PERSUADER [5], CASEY [6] and JULIA [7].

CBR allowed to overcome a number of restrictions inherent to the rule-based systems [8]. The acquisition of knowledge in CBR is reduced to identification of essential features and accumulation of decision making stories (cases) described by the features, that is much easier task than building an explicit knowledge model of the application domain. There are different ways of presenting and storing cases – from simple (linear) to complex hierarchical ones.

At the same time there are two essential shortcomings of traditional CBR. The first one is that description of the cases does not usually take into account the deeper knowledge of the application domain. The second one reveals itself when the number of cases accumulated in the knowledge base becomes great. The large case base results in reduced system performance. Very often the search dictionaries and algorithms for determining similarity are needed to debug manually. These shortcomings could neutralize the benefits of case-based approach.

In order to overcome these disadvantages CBR has been widely integrated with other methods in various application domains [9, 10]. Some systems (ADIOP, CADRE, CADSYN, CHARADE, COMPOSER, IDIOM, JULIA) integrated CBR with constraint satisfaction problem (CSP). Some systems (ANAPRON, AUGUSTE, CAMPER, CABARET, GREBE, GYMEL and SAXEX) combined CBR with RBR. It is worth to note that the first prototype of the system, integrating CBR with RBR was CABARET [11]. In [12] it is proposed possible connection of CBR with RBR and its application to the financial domain implemented in the MARS prototype system.

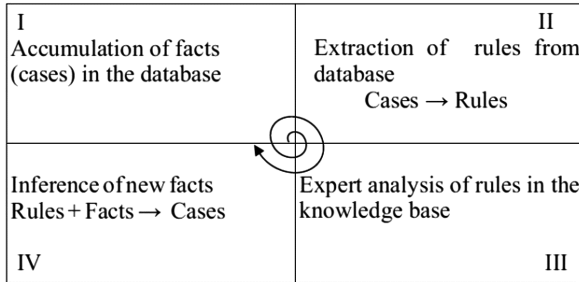
In present paper, an original approach is proposed in which it is possible the dynamic interaction of both CBR and RBR models of knowledge representation and its application to the task of cases classification. Section 2 provides a hybrid model of dynamic interaction of CBR and RBR. In Sect. 3 we give the method of transformation of a set of cases into the system of fuzzy classification rules and discuss its improvement. Section 4 contains some experimental results and their explanation.

## 2 Dynamic Integration of CBR and RBR

The interaction between the employees of the organization on solving the important tasks generates creation of new knowledge. In the process of knowledge transformation both its forms are used: non-formalized and formalized knowledge. While creating new knowledge the formalized (explicit) knowledge and non-formalized (tacit) knowledge interact in four ways: socialization, externalization, combination, internalization according to SECI model (Socialization, Externalization, Combination, Internalization [13]. Sequential alternation of four processes - socialization, externalization, combination, internalization - creates a spiral of knowledge. The process is developing by spiral consistently through these four stages.

In the previous section we described two basic approaches to knowledge representation - case-based and rule-based approaches. Both models have their advantages and disadvantages. Therefore, in practice it would be reasonable to join advantages of both

approaches. The combination of these approaches could be represented as cyclic knowledge transformation from case-based form to the rule-based and vice versa, as shown in Fig. 1, by the analogy with Nonaka-and-Takeuchi spiral [13].



**Fig. 1.** Transformation of knowledge between case base and rule base

At the initial stages of the KBS life cycle, when there is no insight into the application domain, it is advisable to use a case-based model in which the knowledge is represented by relevant precedents (cases) of decision making (stage I). To create a precedent it is necessary to determine a set of features that uniquely determine the situation and the specific solution made in this situation. A simple linear form of the case can be presented as  $(n + 1)$  - dimensional tuple

$$CASE = (x_1, x_2, x_3, \dots, x_n, s), \tag{1}$$

where  $x_1, x_2, x_3, \dots, x_n$  are the features identifying the situation,  $s$  is a solution to the problem defined in the case. Subsequently, with the deepening into the problem domain, possible complication of the case structure is possible, through, for example, the introduction of hierarchy and other relationships between the features.

Already at the earliest stages of the KBS development it is possible to extract, adapt and apply cases in solving the current problems. After applying the case from the knowledge base to the current situation, a new precedent is recorded to the case base for future use. Note that in terms of Nonaka-and-Takeuchi cycle, we do not carry out the formalization of knowledge here. Moreover, we do not try to analyze why the solution is made (i.e. we do not try to transform tacit knowledge into the formal representation), but we simply fix the fact of tacit knowledge in a particular case.

As soon as we accumulate sufficient volume of decision-making cases and deepen our knowledge on the basis of analyzing the case base, it is possible to carry out the formalization of knowledge, contributing to the transformation of tacit knowledge into the explicit one (stage II). Machine learning offers a variety of approaches to extract knowledge from data (simple cases). In this article we consider the problem of obtaining a set of formal rules for classification of cases. Classification allows, on the one hand, to arrange a space of cases in order to improve the retrieving performance. On the other hand, classifying rules are formal knowledge of higher level than simple cases. They can be meaningfully analyzed by an expert who can get possibly new information for decision-making in the future.

If we have simple cases of the form (1), we could apply, for example, decision tree method [14] to this sample of cases, in which the intermediate nodes of the tree correspond to the features  $x_1, x_2, x_3, \dots, x_n$ , and terminal nodes are the classes to which solutions belong. Another method of forming a set of fuzzy rules for cases classification is given in the next section and based on the original papers [15–17].

The transition to a rule-based model means that we obtain the explicit (formalized) form of knowledge, able to explain the cause-and-effect relationships in the application domain. Such reasoning could be presented to experts for analysis and interpretation (stage III of knowledge transformation cycle). The important point at this stage is the resolution of conflicts arising in a system of rules (conflicting rules lead to different conclusions for the same premises).

At the stage IV the accumulation of new cases takes place using the received version of a decision support system. New cases can be obtained from application of logical inference methods to the knowledge base consisting of rules and facts (cases), and also as a result of CBR-cycle, adapting old cases from the knowledge base to new problems. Getting new cases means the use of formal (explicit) knowledge and turning them into the implicit (tacit) form.

The considered cycle of knowledge transformation is repeated further on the next level. As soon as our knowledge about application domain deepen it is possible to make the structure of cases more complex by transition from the linear parametric form to the hierarchical or more complex logical form. At each stage of the proposed cycle, the conversion is made of the implicit knowledge into the explicit form, or vice versa, and each new turn of the spiral brings additional new aspects and dimensions to the knowledge of the application domain.

### 3 Method of Cases Classification Through Fuzzy Rules

Here we consider the method of obtaining fuzzy rules from cases in application to the problem of cases classification. The idea of the method is as follows. Consider arbitrary case of the form (1), where the features  $x_1 \in X_1, x_2 \in X_2, x_3 \in X_3, \dots, x_n \in X_n$ , and the solution variable  $s$  belongs to a certain set of classes  $D = \{d_1, d_2, \dots, d_m\}$ . Without loss of generality we assume that the features are numerical and sets  $X_1, X_2, X_3, \dots, X_n$  are continuous numerical intervals.

Define linguistic variables  $V_i$  corresponding to the features  $x_i, i = \overline{1, n}$ . Let each variable  $V_i$  has  $J$  term-values  $A_i^{(j)}$  that are fuzzy sets determined by membership functions  $\mu_{A_i^{(j)}}$ , on the universal sets  $X_i, j = \overline{1, J}$ . Then the rule *Rule* corresponding to the case (1), is formulated as follows:

$$\text{Rule: IF } V_1 = F(x_1) \& \dots \& V_n = F(x_n) \text{ THEN } R = d \tag{2}$$

where  $R$  is the classifying variable,  $d \in D, F(x_i)$  is a fuzzy term-value which the linguistic variable  $V_i$  accepts as a result of fuzzification of the feature  $x_i$ ;

$$F(x_i) = A_i^{(j^*)}, \text{ where } j^* = \arg \max_{j=1, J} \mu_{A_i^{(j)}}(x_i). \tag{3}$$

Then, knowledge base consisting of a set of rules from the initial case base is formed in accordance with the following algorithm.

**An algorithm of obtaining fuzzy classification rules from case base**

- Step 1. Define the number and the type of membership functions and universal sets  $X_i$  for each linguistic variable  $V_i$  corresponding to the feature  $x_i$ .
- Step 2. Determine fuzzy rule *Rule*, of the form (2) for each case of the form (1).
- Step 3. Assign truth degree  $TD(Rule)$  to each rule. The simplest way to do this is to compute minimum of all  $\mu_{A_i^{(j^*)}}(x_i)$  for every  $x_i$  and fuzzy sets  $A_i^{(j^*)}$  defined by the formula (3) as a result of fuzzification for each rule *Rule* of the form (2). The result is obtained in the following way

$$TD(Rule) = \min_{i=1, n} \mu_{A_i^{(j^*)}}(x_i) = \min_{i=1, n} \max_{j=1, J} \mu_{A_i^{(j)}}(x_i). \tag{4}$$

- Step 4. Resolve conflicts between the rules. After the step 2 we obtain a set of rules uniquely corresponding to the set of cases. But this set can contain subsets of the rules with the same premises. These rules can have the same or different conclusions (solutions  $d \in D$ ). If for all rules in such subset we have the same conclusion we simply remove all duplicate rules from the set except one. For the rules with conflicting conclusions, we propose two different strategies for classification problem. In each case we leave only one rule in the subset with conflicted conclusions in the rule base. Let we have a subset of rules  $Rule_1, \dots, Rule_m$  with the same premise. Divide the set of indexes  $I = \{1, 2, \dots, m\}$  into  $K$  subsets  $I_1, I_2, \dots, I_K$  corresponding to the classes of solutions  $d_1, d_2, \dots, d_K$  in the

$$\text{conclusions of the rules (3), } \dim I_k = m_k, \sum_{k=1}^K m_k = m.$$

The first strategy of resolving conflicts is to obtain the optimal rule

$$l^* = \arg \max_{l \in I} TD(Rule_l), \tag{5}$$

and then to choose a solution  $d$  corresponding to the conclusion of this optimal rule. The second strategy of resolving conflicts is a bit more complex. The optimal solution  $d_{k^*}$  that is assigned to the conflicting set of rules could be obtained as follows

$$k^* = \arg \max_k \left( \sum_{i \in I_k} TD(Rule_i) \right). \tag{6}$$

### 4 Experimental Results

In this section we investigate the proposed approach with the data set Iris (available at <http://archive.ics.uci.edu/ml/>) with 150 cases, characterized by 4 features ( $n = 4$ ), which are classified into 3 classes ( $K = 3$ ) through classifying attribute.

In the first experimentation, we randomly selected training sample of given size (120 cases) from Iris set, while the rest of the data (30 cases) was used as a control sample. Then, for each feature, minimum and maximum values were determined from the training sample. The obtained intervals were used as the universal sets for corresponding linguistic variables. Then we divided each universal set  $X_i$  into equal intervals for three variants of choosing the number of term-values  $A_i^{(j)}$ ,  $j = \overline{1, J}$ :  $J = 3, 5$  and  $7$  ( $J$  does not depend on  $i$ ) for membership functions (3MF, 5MF and 7MF). We also considered two classes of MF – linear (class T) and quadratic (class S).

Then a set of fuzzy classifying rules obtained as a result of machine learning algorithm was applied to control sample to compute an accuracy of classification as a ratio of correctly classified cases to the general volume of the control sample. In Table 1 we presented the results of classification accuracy for two strategies of control sample classification with linear membership functions (3MF, 5MF and 7MF).

**Table 1.** Classification accuracy for two strategies of control sample classification with linear membership functions (3MF, 5MF and 7MF)

Number of trial	Control accuracy for strategy 1			Control accuracy for strategy 2		
	3 MF	5 MF	7 MF	3 MF	5 MF	7 MF
1	0,90	0,83	0,83	0,97	0,88	0,88
2	0,87	0,80	0,63	0,93	0,92	0,78
3	0,97	0,77	0,73	0,95	0,83	0,83
4	0,80	0,63	0,63	0,97	0,83	0,83
5	0,83	0,73	0,57	0,95	0,88	0,77
6	0,87	0,83	0,60	0,95	0,90	0,82
7	0,80	0,70	0,63	0,97	0,88	0,83
8	0,90	0,77	0,70	0,95	0,85	0,78
9	0,93	0,83	0,73	0,93	0,85	0,83
10	0,87	0,67	0,80	0,97	0,87	0,83
<b>Average</b>	<b>0,88</b>	<b>0,76</b>	<b>0,69</b>	<b>0,95</b>	<b>0,87</b>	<b>0,82</b>

From Table 1 we can see that the best results were obtained for 3MF. Reducing in the accuracy for 5MF and 7MF could be explained by the fact that the number of possible combinations of preconditions in the rules increases significantly, so classification on the control sample fails because of the lack of appropriate rule for the classified case. Therefore we compare two types of control cases classification. For the first strategy we apply each rule only to exactly corresponding control cases. If there are no appropriate rules, incorrect classification is registered. For the second strategy, if there is no appropriate rule to classify the control case, we exclude one feature and try to find classifying rule for reduced number of features (Here we assume that the excluded feature is not

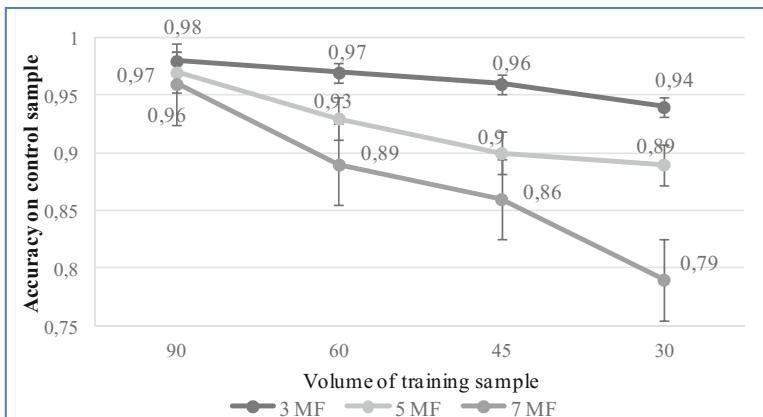
important for classification). The first appropriate rule is applied for classification of the corresponding case. The second strategy permits to improve the results of cases classification. The experiments were performed ten times, each time having arbitrary set of cases for training.

In Table 2 we give classification accuracy for linear and quadratic membership functions (class T and S for 3MF). Here we can see that for smoother membership functions (class S) the classification accuracy is better.

**Table 2.** Classification accuracy for two strategies of control sample classification with linear and quadratic membership functions (class T and S for 3MF)

Number of trial	Control accuracy for strategy 1		Control accuracy for strategy 2	
	Class T	Class S	Class T	Class S
1	0,90	0,97	0,97	0,98
2	0,87	0,97	0,95	0,97
3	0,93	0,93	0,98	0,95
4	0,87	0,97	1,00	0,93
5	0,87	0,83	0,98	0,95
6	0,85	0,92	0,95	1,00
7	0,87	0,88	0,95	0,93
8	0,82	0,90	0,95	0,92
9	0,87	0,87	0,98	0,98
10	0,85	0,90	0,97	0,95
<b>Average</b>	<b>0,87</b>	<b>0,91</b>	<b>0,97</b>	<b>0,96</b>

Figure 2 gives visual illustration of comparison between the three classes of linear membership functions (3MF, 5MF and 7MF) for sequentially reducing volume of training sample. Here we can observe that the classification accuracy for sequentially



**Fig. 2.** Classification accuracy for sequentially reducing volume of training sample for linear membership function (T) for 3MF, 5MF and 7 MF

reducing volume of training sample (from 90 cases to 30) remains quite high. The results could be explained by the fact that the number of rules obtained from the initial training sample of 150 cases was not more than 17 in all the experiments. If we randomly choose less amount of cases for training, the information content (or representativeness) of the sample remains still high.

Thus, we have proposed rather simple method of generating classification rules from case base that can be used in the cycle of knowledge transformation in the KBS. Despite the simplicity we have achieved quite high accuracy of classification even for sequentially reducing volume of training samples. We believe that accuracy of classification can be improved further through the extension of the rule structure (2) by introducing disjunctions into a conjunctive premises as well as considering the possibility of reducing the number of the rule premises by leaving the most informative ones, that is the subject of further research.

**Acknowledgments.** The reported study was funded by Russian Ministry of Education and Science, according to the research project No. 2.2327.2017/4.6

## References

1. Schank, R.C., Abelson, R.P.: *Scripts, Plans, Goals and Understanding*. Erlbau, New York (1977)
2. Schank, R.C.: *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*. Cambridge University Press, New York (1982)
3. Simpson, R.L.: *A computer model of case-based reasoning in problem solving: an investigation in the domain of dispute mediation*. Technical report GIT-ICS-85/18, Georgia Institute of Technology, School of Information and Computer Science (1985)
4. Hammond, K.J.: *CHEF: a model of case-based planning*. In: American Association for Artificial Intelligence, AAAI 1986, Philadelphia, PA (1986)
5. Sycara, E.P.: *Resolving adversarial conflicts: an approach to integrating case-based and analytic method*. Technical report GIT-ICS-87/26, Georgia Institute of Technology, School of Information and Computer Science (1987)
6. Koton, P.: *Using experience in learning and problem solving*. Ph.D. thesis, Massachusetts Institute Technology, Laboratory of Computer Science (1989)
7. Hinrichs, T.R.: *Problem Solving in Open Worlds*. Lawrence Erlbaum, Hillsdale (1992)
8. Watson, I., Marir, F.: Case-based reasoning: a review. *Knowl. Eng. Rev.* **9**(4), 327–354 (1994)
9. Marling, C., Sqalli, M., Rissland, E., Hector, M.A., Aha, D.: Case-based reasoning integrations. *AI Mag.* **23**(1), 69–86 (2002)
10. Yang, H.L., Wang, C.S.: Two stages of case-based reasoning - integrating genetic algorithm with data mining mechanism. *Expert Syst. Appl.* **35**, 262–272 (2008)
11. Rissland, E.L., Skala, D.B.: Combining case-based and rule-based reasoning: a heuristic approach. In: Eleventh International Joint Conference on Artificial Intelligence, IJCAI 1989, Detroit, pp. 524–530 (1989)
12. Dutta, S., Bonissone, P.P.: Integrating case- and rule-based reasoning. *Int. J. Approx. Reason.* **8**(3), 163–203 (1993)
13. Nonaka, I., Takeuchi, H.: *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, New York (1995)



14. Quinlan, J.R.: Induction of decision trees. *Mach. Learn.* **1**, 81–106 (1986). Kluwer Academic Publishers
15. Xiong, N.: Learning fuzzy rules for similarity assessment in case-based reasoning. *Expert Syst. Appl.* **38**, 10780–10786 (2011)
16. Xiong, N.: Fuzzy rule-based similarity model enables learning from small case bases. *Appl. Soft Comput.* **13**, 2057–2064 (2013)
17. Avdeenko, T.V., Makarova, E.S.: Integration of case-based and rule-based reasoning through fuzzy inference in decision support systems. *Procedia Comput. Sci.* **103**, 447–453 (2017)