Exploiting domain-experts knowledge within an evidential process for case base maintenance

Safa Ben Ayed[1,2], Zied Elouedi[1], and Eric Lefevre[2]

[1] LARODEC, Institut Supérieur de Gestion de Tunis, Université de Tunis, Tunisia
[2] Univ. Artois, EA 3926, LGI2A, 62400 Béthune, France

Abstract. Case Base Maintenance (CBM) presents one of the key factors success for Case Based Reasoning (CBR) systems. Thence, several CBM policies are proposed to improve their problem-solving performance and competence. However, to the best of our knowledge, all of them are not able to make use of prior knowledge which can be offered by domain experts, especially that CBR is widely applied in real-life domains. For instance, given symptoms of two different cases in medicine area, the doctor can affirm that these two cases should never follow the same treatment, or conversely. This kind of prior knowledge is presented in form of *Cannot-Link* and *Must-link* constraints. In addition, most of them cannot manage uncertainty in cases during CBM. To overcome this shortcoming, we propose, in this paper, a CBM policy that handles constraints to exploit experts' knowledge during case base learning along with managing uncertainty using the belief function theory. This new CBM approach consists mainly in noisy and redundant cases deletion.

1 Introduction

Case Based Reasoning is a methodology for reasoning through adapting previous experiences to solve new problems. Each success solving operation will be retained for future learning, where an incremental aspect characterizes the case bases evolution [1]. As CBR systems are widely applied within real-life domains, and as they are designed to work over long time frames, the Case Base Maintenance (CBM) becomes a fundamental task to guarantee their success. In fact, CBM has been defined as the field that cares on implementing policies that aim to reach a particular set of performance objectives through revising the content and the organization of case bases [2]. Indeed, we note a great interest within current research that addresses issues for growing case bases. For instance, CBM policies may be divided into two strategies, even to the optimization strategy where the deletion is done after optimizing a given evaluation criterion, or to the partition strategy which allows to treat a set of small case bases independently. In the latter strategy, uncertainty about the membership of cases to the different classes (clusters) have also been handled [3][4]. However, these CBM policies are not offering the possibility to exploit background knowledge which can be provided by an expert of domain in which the CBR system is deployed. Therefore, we aim, in this paper, to propose a new CBM approach based on an evidential clustering to manage uncertainty about the membership of cases.

Moreover, this approach handles extra-information for cases clustering presented in the form of two types of constraints [5]: *Must-link* constraints which specify that two cases have the same solution and *Cannot-link* constraints which specify that two solutions cannot belong to the same cluster. To do, we used then the Constrained Evidential C-Means algorithm (CECM) [6]. The remainder of this paper is organized as follows. Section 2 reviews briefly some CBM approaches based on clustering techniques. Section 3 describes the used constrained evidential clustering technique called CECM. Our new CBM approach will be detailed in Section 4. Throughout Section 5, we discuss experimental settings, the pairwise constraints generation, testing strategy, and results.

2 Clustering-based CBM policies

Intuitively, when addressing the problem of maintaining a large case base, its decomposition into a number of related closely cases groups appears to be a good solution for their maintenance. Indeed, clustering techniques have been well applied within CBR since the notions of neighborhood and distances between cases are well presented. Actually, there are several works in this way. However, during the rest of this Section, two of them which handle uncertainty regarding the membership of cases to different clusters will be reviewed. The first one is called SCBM noting "Soft case base maintenance method based on competence model" which groups cases within the frame of fuzzy sets theory [7]. Then, it tries to detect the right case types to be removed without decreasing the competence of the CBR system. The second policy is named ECTD for "Evidential Clustering and case Types Detection for case base maintenance" which is more able to manage uncertainty using the belief function theory [8][9]. First, ECTD applies ECM [10] algorithm to group cases and obtain the credal partition of cases along with the different clusters centers. Then, it reasons on the way of detecting four types of cases in order to be able at the end to eliminate noisiness and redundancy. However, techniques used inside these methods do not allow to make use of the background knowledge that helps to guide to the best solution. For this paper, we consider prior knowledge in form of Must-link and Cannot-link constraints. To do, we apply on the case base a constrained evidential clustering technique as presented in the following Section.

3 Constrained evidential clustering technique: CECM

When dealing with clustering-based CBM policies, it is gainful to express prior knowledge in form of instance level constraints as indicated in the Introduction. In what follows, we will present CECM through its constraints expression and work standard.

3.1 Constraints expression by CECM

Let two objects $\boldsymbol{o_i}$ and $\boldsymbol{o_j}$ and their associated mass functions m_i and m_j. The mass function $m_{i \times j}$ regarding their joint class membership may be calculated in the Cartesian product $\Omega^2 = \Omega \times \Omega$, as the combination between m_i and m_j [11] such that:

$$m_{i\times j}(A \times B) = m_i(A)\ m_j(B)\ , \qquad A, B \subseteq \Omega, A \neq \emptyset, B \neq \emptyset \tag{1a}$$

$$m_{i\times j}(\emptyset) = m_i(\emptyset) + m_j(\emptyset) - m_j(\emptyset)\ m_j(\emptyset) \tag{1b}$$

Let the subset $\theta = \{(\omega_1, \omega_1), (\omega_2, \omega_2), ..., (\omega_c, \omega_c)\}$ in Ω^2 (where c is the number of classes) presents the event *"The pair of objects $\boldsymbol{o_i}$ and $\boldsymbol{o_j}$ belong to the same class"*. Therefore, after calculating the plausibility $pl_{i\times j}$ from $m_{i\times j}$, the value $pl_{i\times j}(\theta) = 0$ corresponds to a Cannot-link constraint ($\mathcal{C}$) between $\boldsymbol{o_i}$ and $\boldsymbol{o_j}$ and the value $pl_{i\times j}(\overline{\theta}) = 0$ corresponds to a Must-link constraint ($\mathcal{M}$) between $\boldsymbol{o_i}$ and $\boldsymbol{o_j}$.

3.2 Objective function and Optimization of CECM

First of all, let mention that CECM [6] is a variant of ECM [10] algorithm (noisiness is assigned to the empty set partition). The principle of both of them during the evidential clustering is to minimize an objective function in order to maximize distances between objects belonging to different classes and minimizing those belonging to the same one. The objective function for ECM algorithm is defined such that:

$$J_{ECM}(M, V) = \frac{1}{2^c n}\ [\ \sum_{i=1}^{n} \sum_{A_k \neq \emptyset} |A_k|^{\alpha} m_{ik}^{\beta} d_{ik}^2 + \sum_{i=1}^{n} \rho^2 m_{i\emptyset}^{\beta}\ ] \tag{2}$$

subject to:

$$\sum_{j/A_j \subseteq \Omega, A_j \neq \emptyset} m_{ij} + m_{i\emptyset} = 1 \qquad \forall i = 1, .., n \tag{3}$$

where M represents the credal partition of n objects to c clusters, V presents 2^c clusters centers, d_{ij} represents a given distance between $\boldsymbol{o_i}$ and $\boldsymbol{o_j}$, ρ and β are two parameters to treat noisy objects, and the coefficient α controls the penalization of degree's allocation to subsets with high cardinality.

CECM algorithm shares the same standard of ECM with an additional requirement that $pl_{i\times j}(\theta)$ (respectively $pl_{i\times j}(\overline{\theta})$) should be as low as possible if $(\boldsymbol{o_i}, \boldsymbol{o_j}) \in \mathcal{C}$ (respectively $(\boldsymbol{o_i}, \boldsymbol{o_j}) \in \mathcal{M}$). Consequently, its objective function to be minimized is defined such that:

$$J_{CECM}(M, V) = (1 - \xi) J_{ECM}(M, V) + \xi J_{CONST} \tag{4}$$

where the parameter ξ controls the balance between constraints and geometrical model, and J_{CONST}, which indicates $\mathcal{C}$ and $\mathcal{M}$ violating cost, is defined such that:

$$J_{CONST} = \frac{1}{|\mathcal{M}| + |\mathcal{C}|}\ [\ \sum_{(\boldsymbol{o_i}, \boldsymbol{o_j}) \in \mathcal{M}} pl_{i\times j}(\overline{\theta}) + \sum_{(\boldsymbol{o_i}, \boldsymbol{o_j}) \in \mathcal{C}} pl_{i\times j}(\theta)\ ] \tag{5}$$

To minimize Equation 4, an alternate optimization scheme has been proposed in [6] aiming to fix the partition matrix M and the centroid matrix V. Furthermore, CECM with adaptive metric (Mahalanobis distance) is proposed to support arbitrary shapes of clusters. More details of optimization will be found on [6].

4 Maintaining case bases through Constrained Evidential Clustering and case Types Detection (CECTD)

In this Section, we present the different steps of our CBM approach. To build our case base maintainer, our method applies the constrained evidential clustering analysis, detects cases that should be eliminated from the case base, and performs the maintenance.

4.1 Case bases clustering with background knowledge

First, we perform on case bases the CECM constrained evidential clustering as presented in Section 3, where each object is considered as a case and its class presents the solution part of that case. The background knowledge is presented as case-level constraints. Actually, CECM algorithm manages uncertainty by offering clusters centers along with the credal partition which provides the belief degree of cases membership to the different partitions. These two outputs are the source of case types detection strategy.

4.2 Case types detection

Several works on the CBM field divide cases into different types according to their role towards to whole case base or their competence for other problems resolution. In this paper, we classify cases into four types [4][3] such that:

- *Noisy cases*: They present a distortion of values and cannot be correctly classified in any one of clusters.
- *Similar cases*: They present a number of cases which are so close that they are considered as redundant.
- *Isolated cases*: They are dissimilar and situated in clusters borders.
- *Internal cases*: They present the center of each group of similar case.

Detect Noisy cases Since CECM algorithm allocates a high belief's degree to the empty set for noisy cases, we propose, as in [4], to detect them such that:

$$\boldsymbol{x_i} \in NC \quad iff \quad m_i(\emptyset) > \sum_{A_j \subseteq \Omega, A_j \neq \emptyset} m_i(A_j) \tag{6}$$

where $\boldsymbol{x_i}$ presents one case and NC represents the set of all the Noisy cases.

Distinguish between Similar and isolated cases Let c clusters are obtained after cases clustering step. Logically, the majority of cases are situated in the core of each cluster (Similar cases). However, we find some cases which are isolated and far somehow to the cluster's center (Isolated cases). To distinguish between these two types, we compare cluster-case distance to a given threshold (Th_k) which has been defined as the mean of all cases distances to a given cluster's

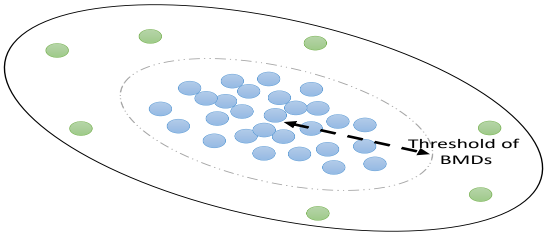

Fig. 1. Distinguish between similar and isolated cases within a cluster using a threshold

center (see Figure 1). To calculate the distance between a case and cluster's center, we chose to use the following Belief Mahalanobis Distance (BMD) [4]:

$$BMD(\boldsymbol{x_i}, \boldsymbol{v_k}) = \sqrt{(\boldsymbol{x_i} - \boldsymbol{v_k})^T \Sigma_k^{-1} (\boldsymbol{x_i} - \boldsymbol{v_k})} \tag{7}$$

where $\boldsymbol{v_k}$ is the k^{th} cluster's center generated by CECM, and Σ_k presents the *Belief Covariance Matrix* which has been presented in [6] as follows:

$$\Sigma_k = \sum_{i=1}^{n} \sum_{A_j \ni w_k, A_j \subseteq \Omega} m_{ij}^2 |A_j|^{\alpha-1} (\boldsymbol{x_i} - \overline{\boldsymbol{v_j}})(\boldsymbol{x_i} - \overline{\boldsymbol{v_j}})^T \tag{8}$$

where k is the cluster's number with $k = 1, .., c$, m_{ij} and $\overline{\boldsymbol{v_j}}$ are respectively the credal partition and their prototypes defined by CECM.

Ultimately, we distinguish between Similar and Isolated cases such that:

$$\boldsymbol{x_i} \in \begin{cases} SC_k & if\ \exists k / BMD(\boldsymbol{x_i}, \boldsymbol{v_k}) < Th_k \\ IsC & Otherwise \end{cases} \tag{9}$$

where SC_k is the set of similar cases, IsC is the set of Isolated ones and the threshold Th_k is defined such that:

$$Th_k = \frac{\sum_{\boldsymbol{x_i} \notin NC} BMD(\boldsymbol{x_i}, \boldsymbol{v_k})}{\#TotalCases - \#NoisyCases} \tag{10}$$

Flag Internal cases From each group of Similar cases, we have to flag an internal case as a representative for covering all of them. Hence, we choose to detect this case as the closest one to each cluster's center using BMD. Hence, they can be formally defined such that:

$$\boldsymbol{x_i} \in InC \;\; iff \;\; \exists k; \neg \exists \boldsymbol{x_j} / BMD(\boldsymbol{x_j}, \boldsymbol{v_k}) < BMD(\boldsymbol{x_i}, \boldsymbol{v_k}) \tag{11}$$

where $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ are two cases, and InC represents the set of Internal cases.

4.3 Case base maintenance

While maintenance, we aim to remove cases that are dispensable or distorting the problem-solving process. Through this idea, we remove cases detected as Similar in order to eliminate redundancy and improve performance, as well as Noisy cases so as to improve the competence of CBR systems in problem resolution.

5 Experimental study using artificial constraints

During this Section, we aim to differently generate the pairwise *Must-link* and *Cannot-link* constraints, as well as to validate our new CBM method benefit.

5.1 Experimental setting

Our new CBM approach has been developed using R-3.3.2 and it is tested on a number of numeric case bases from UCI Repository which are described in Table 1 by their references, number of attributes, size, number of classes and their classes distribution. While developing, default values are taken for the CECM parameters, and the number of clusters and classes were equally taken. Besides, we used CECM with adaptive metric to consider arbitrary clusters' shape.

Table 1. UCI data sets used in our experimental study

Case base	Reference	Attributes	Instances	Classes	Class distribution
Sonar	SN	60	208	2	97/111
Ionosphere	IO	34	351	2	226/125
Heberman	HB	3	306	2	225/81
Seeds	SD	7	210	3	70/70/70
Mammographic	MM	6	961	2	516/445
Banknote authentication	BA	5	1372	2	762/610

5.2 Pairwise constraints generation

The aim of this subsection is to implement two different ways for artificially-generating constraints in conjunction with experiments applied on our method. The idea consists in randomly picking two cases. If they are classified with high degree of certainty ($m_i(A) > 0.5$ with A is a singleton partition), we generate a constraint through their solution (If they have the same solution, we create a Must-link constraint, otherwise we generate a Cannot-Link constraint). Therefore, we perform the following two ways:

- Batch constraints generation (CECTD$_{bat}$): Apply ECM algorithm (CECM without constraint), generate a number of constraints equal to 10% of the case base size. Then, apply our CECTD method.
- Alternate constraints generation (CECTD$_{alt}$): Within the first step of our method, we alternate between running CECM and generating randomly one constraint having high degree of certainty, until reaching 10% of constraints.

5.3 Maintenance testing strategy

To measure the effectiveness of our maintaining method, we track the following testing strategy. Each case base is divided into Training set (T_r) and Test set

(T_s), and we apply our maintaining method on T_r to obtain a modified Training set (T'_r). Then, we compute three evaluation criteria as follows:

1. Classify T_s from T'_r using 1-Nearest Neighbor algorithm. Therefore, the classification accuracy to measure the performance is calculated such that:

$$PCC(\%) = \frac{\#\ correct\ classifications\ on\ T_s}{size\ of\ T_s} \times 100$$

2. Measure the Retrieval Time (RT) as the time spent to classify all cases' instances in T_r using 1-NN.
3. Calculate the storage size as the data Retention Rate (RR) of T_r comparing to T'_r as follows:

$$RR\ (\%) = \frac{size\ of\ T'_r}{size\ of\ T_r} \times 100$$

The final estimation of each evaluation criterion is obtained by averaging ten trials values using 10-Folds cross validation technique.

5.4 Experimental results

According to the evaluation criteria mentioned above, we compare our method with its two different ways to generate constraints (CECTD$_{bat}$ and CECTD$_{alt}$) to the Initial case base (ICBR) as well as to ECTD method [4]. Results are therefore shown in Tables 2 and 3. Obviously, we tolerate some degradation in accuracy after maintenance at the aim of accelerating cases retrieving task and improving CBR systems performance. Nevertheless, Table 2 shows some improvements in accuracy especially with the alternate version of our approach. For instance, it moves from 80.78% to 82.10% after applying CECTD$_{alt}$. In parallel, Table 3 presents, in term of cases retention rate and retrieval time, how our approach can notably boost CBR systems. Herein, we note that we were able to reduce more than half of all case bases. For example, "Heberman" dataset were reduced by CECTD$_{alt}$ until almost quarter. Moreover, even with using 1-NN for classification, we clearly note the improvement of retrieval time values particularly comparing to the Initial non-maintained case base, where all of them move from about 0.1 s to about 0.001 s.

Table 2. Accuracy evaluation (%)

Case bases	ICBR	ECTD	CECTD$_{bat}$	CECTD$_{alt}$
SN	80.78	68.31	79.78	82.10
IO	85.47	79.45	85.00	84.90
HB	72.88	67.23	70.85	72.88
SD	90.00	83.16	88.70	90.18
MM	79.81	72.13	80.01	79.92
BA	99.12	86.40	88.97	95.14

Table 3. Data Retention Rate (%) and Retrival Time (s) evaluation

CB	ICBR		ECTD		CECTD$_{bat}$		CECTD$_{alt}$	
	RR	RT	RR	RT	RR	RT	RR	RT
SN	100	0.1003	48.98	0.0021	48.50	0.0026	46.51	0.0020
IO	100	0.0094	37.04	0.0017	35.36	0.0017	33.89	0.0015
HB	100	0.0993	29.72	0.0027	34.52	0.0021	28.14	0.0019
SD	100	0.0911	44.13	0.0023	45.77	0.0018	43.98	0.0016
MM	100	0.0852	26.23	0.0014	39.57	0.0016	40.02	0.0022
BA	100	0.1033	31.82	0.0026	44.54	0.0036	39.15	0.0027

6 Conclusion

Aiming at the performance and learning capability issues that the growing scale of CBR case bases brings, a new CBM approach based on a constrained evidential clustering technique has been developed, in this paper, using two ways for constraints generation with managing uncertainty. Better results are offered, during experiments, when generating constraints one by one alternatively with running CECM.

References

1. A. Aamodt, E. Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. In *AI communications* (IOS press, 1994), pp. 39-59.
2. D. C. Wilson, D. B. Leake. Maintaining case-based reasoners: Dimensions and directions. In *Computational Intelligence* (2001), pp. 196-213.
3. A. Smiti, Z. Elouedi. SCBM: soft case base maintenance method based on competence model. In Journal of Computational Science (Elsevier, 2017), DOI: 10.1016/j.jocs.2017.09.013.
4. S. Ben Ayed, Z. Elouedi, E. Lefevre. ECTD: Evidential Clustering and case Types Detection for case base maintenance. In *International Conference on Computer Systems and Applications* (IEEE, 2017), pp. 1462-1469.
5. K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning* (2001), pp. 577-584.
6. V. Antoine, B. Quost, M. Masson, T. Denoeux. CECM: Constrained evidential C-means algorithm. *Computational Statistics & Data Analysis* (Elsevier, 2012), pp. 894-914.
7. L.A. Zadeh. Fuzzy sets. *Information and control 8*, pages 338-353, 1965.
8. G. Shafer. A mathematical theory of evidence. In Princeton university press (Princeton university press, 1976).
9. A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. In *The annals of mathematical statistics* (1967), pp. 325-339.
10. M. H. Masson and T. Denoeux. ECM: An evidential version of the fuzzy c-means algorithm. *Pattern Recognition 41* (2008), pp. 1384-1397.
11. P. Smets, R. Kennes. The transferable belief model. *Artificial Intelligence* (1994), pp. 191-234.