Hybrid Technique and Competence-Preserving Case Deletion Methods for Case Maintenance in Case-Based Reasoning

A. LAWANNA* and J. DAENGDEJ**

*Department of Information Technology, Faculty of Science and Technology, Assumption University, Bangkok, Thailand

**Department of Information Technology, Faculty of Science and Technology, Assumption University, Bangkok, Thailand

Abstract

Case-Based Reasoning (CBR) is one of machine learning algorithms for problem solving and learning that caught a lot of attention over the last few years. In general, CBR is composed of four main phases: *retrieve* the most similar case or cases, *reuse* the case to solve the problem, *revise* or adapt the proposed solution, and *retain* the learned cases before returning them to the case base for learning purpose. Unfortunately, in many cases, this retain process causes the uncontrolled case base growth. The problem affects competence and performance of CBR systems after few runs. This paper proposes two case maintenance methods; the first method is Hybrid technique which combines case addition strategy and the footprint deletion and footprint utility deletion strategy and the second is competence-preserving case deletion technique which is consisted of four steps: *determine* a set of target problems, *determine* a candidate of cases , *determine* target problem and its candidate, *delete* less relevant cases.

Keywords— Case-Based Reasoning, Case Base Maintenance, Coverage, Competence, Performance

1. Introduction

Case Based Reasoning (CBR) is an algorithm of solving new problems based on the solutions of similar past problems [1]. The well-known 4R processes of traditional CBR are *retrieve, reuse, revise,* and *retain* [2-3]. In brief, retrieving one or more similar cases, and attempting to reuse the solution of one of the retrieved cases, possibly after adapting it to account for differences in problem descriptions. The solution proposed by the system is then evaluated and its solution can then be retained as a new case, and the system has learned to solve a new problem [4-5]. In this case, the "retain" process is the one that causes the uncontrolled case base growth which can result in dropping of competence and performance of the system for the next run [6-8]. That is, the utility problem which is a serious performance problem of CBR [9].

Therefore, Case Base Maintenance (CBM) methods are developed to solve this problem [10-12]. Case Base Maintenance (CBM) can be deletion, addition, partitioning technique that can reduce case base size. Accordingly, random deletion (UD) is a simple, inexpensive technique and it is completely domain independent. This technique, simply randomly select and delete cases from a case base once a case base size exceeds some predefined limit [13]. Later on, Minton proposes utility deletion (UD) which is based on Minton's utility metric. It selects cases for deletion by estimating its performance benefits and removing cases with negative utility [14]. However, both techniques degrade the competence and performance of system after deletion [15]. Therefore, Smyth and Keane propose the footprint deletion and footprint utility deletion (FD&FUD) [16]. The key concepts in categorizing cases are coverage and reachability. In brief, given a case base $C = \{c_1, c_2, ..., c_n\}$ and T is the set of target cases in a case base. Formally: Coverage(c) = $\{T \in C$: Adaptable $(c, T)\}$, For $c \in C$, Reachable(c) = $\{T \in C$: Adaptable $(T, c)\}$, For $c \in C$. As a result, 4 categories of cases are considered: pivotal cases (relevant cases), spanning cases which their coverage spaces span regions of pivotal cases and do not affect the competency, support cases are a special class of spanning cases and do not affect the competence, and auxiliary cases are the cases that do not affect the competence at all. The deletion technique then selectively deletes cases from a case base guided by case categorization until a limit on the case base size is reached [17]. Another branch of CBM is case addition proposed by Zhu and Yang. They describe a case addition algorithm that has the added advantage of providing a guaranteed lower bound on resulting competence [18]. They demonstrate three theorems. Theorem 1 shows that FD and FUD can lose almost all the competence in the worst case. They prove that by using FD&FUD, auxiliary cases should be deleted first, followed by supporting, and spanning cases. As a result, only pivot problem is left in the case base if auxiliary cases are increasing up to millions, then the coverage is of the original competence. The percentage of coverage will approach to zero. This causes the quality of the case base is arbitrarily bad. Theorem 2 shows the case addition algorithm produces a case base C such that the coverage of C is no less than 63% of the coverage of an optimal case base. Theorem