An Object Oriented Case-Based Expert System for Awarding Punishment for Serious Discipline Cases

Simon C. K. Shiu
Department of Computing, Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
Email: csckshiu@comp.polyu.edu.hk

Abstract

This paper reports the application of an object-oriented case-based expert system for supporting an officer in awarding punishment for serious discipline cases in the Hong Kong Civil Service (HKCS). Over the years, there exist more than 1,500 serious discipline cases in HKCS, such as negligence of duty, conflict of interest, misbehavior, fraud and dishonesty. The determination of punishment is a very time consuming and error prone process. Furthermore, management also has to balance the interests and demoralizing effect that may arise from the decision. By building an object-oriented case-based expert system to tackle this problem, the efficiency and effectiveness of coming to a decision has been greatly improved. Basically, previous discipline cases were organized into an object hierarchy using the common features for determination of class membership. Retrieval of cases is done in two phases, in phase one, primary features are matched. If they are successfully matched, secondary features will be used with weighting factors to find out the best match. The solution adaptation is done by firing the production rules captured during the knowledge acquisition stage, the result would be an adapted solution based on the parameters from the new input case. The system was built using an object-oriented expert system shell, KAPPA-PC©, and the prototype was tested by existing officers. The results illustrated that the system is a very useful aid for supporting their decision.

Keywords: Object-Oriented, Case-Based Expert System, Discipline Cases

1. Introduction

"A postal officer left the counter at public hall without locking up his cash box. When he returned to the counter, it was discovered that $4,000 was found missing. A further check by Internal Audit Section revealed a further shortage of $1,000 in the counter sub-stock."

"A nurse, while working at the Gynecological section of South Kwai Chung Polyclinic, neglected her duty by failing to properly handle a vaginal swab contained in a test tube; hence the same swab was used for a second time on another patient."

How to process the above two cases is commonly confronted by government officers. It is both a difficult decision as well as a time consuming process. In order to determine a punishment which can be seen as both fair and equitable, an officer may be required to retrieve all the previous cases and study them thoroughly before he is able to find the relevant portion for drawing the analogy. In practice, the officer usually will ask his colleagues for help i.e., ask all officers to find out whether anyone has come across similar cases before. However, as officers are subject to transfer of posting, the expertise usually disappears when the experienced officers have left. Therefore, in order to create a corporate memory of cases, and to allow for sharing of case information, a **P**unishment **A**warding case-based expert **S**ystem (PAS) was built [7].

There are a number of similar case-based expert systems developed in the past, and the most representative two are JUDGE developed by Bain [1] and HYPO developed by Rissland and Ashley [2]. These systems work in the area of legal cases, they model a judge who is determining sentences for people convicted of crimes. The case library contains previous crimes and sentences. A new crime is first interpreted by evaluating the events for seriousness, intentionality, justification, and so on, guided by interpretations assigned to prior cases. Then, the interpreted case is used to retrieve cases from the library. Finally, the sentence stored in the retrieved case is adapted to the current crime by making the sentence more or less stringent, depending how the new crime compared to the old ones. Common to the above two systems is the specially designed knowledge driven retrieval mechanism for identifying similar case [3]. The major difference of PAS compared with JUDGE and HYPO is the organization of case knowledge, and the hierarchical matching [9] of features. These are further explained in the following sections.

2. Design of the Expert System

The knowledge acquisition stage [4,6] involved three activities: 1) searching the existing documents, 2) conducting interviews and 3) real life experience. In activity 1, the relevant documents revealed include the Government Manual on Discipline, Labor Laws, Staff Contract information. They provide definitions of terminology, the regulations being enforced and the standard guidelines in handling discipline cases. In activity 2, a total of 20 officers were interviewed, the objectives of these interviews were to understand the decision process and record the procedure for processing. In addition, difficulties and problems in handling the tasks were also investigated (see Appendix 1). In activity 3, the knowledge engineer tries to do several cases himself with the aid of experience officer in order to go through the whole decision process and to experience the problem solving cycle.

Having interviewed the experts and searched through the documents, value of precedent cases is that they help to give guidance for officers to make the judgement and these precedent cases can also be cited to add weight to the argument and recommendations. However, officers find it very frustrating to search for precedent cases. Firstly, there is only a manual system and they have to go through the whole library before identifying the possible cases they want. Secondly, since about 15 officers are concurrently sharing the same library and many a time the search is being delayed due to the queue. In addition, the officer has to go through each and every case before he can safely say that the case found is the one he want. It is obviously a problem, which requires immediate attention. The fact finding exercise also revealed that each case was found to consist mainly of four major categories of information:

1) general data about the staff and the offence,
2) nature of the offence, and detail breakdown of the severity of offence into sub-categories,
3) considerations for punishment, e.g., the severity of offence, the offender's attitude towards the case and
4) the punishment itself.

Typical information are source of case, date of offence, offender personnel data and employment history, nature of offence (e.g., negligence of duty, abuse of power, unauthorized absence, fraudulent use of information, improperly receive allowance, disobey instruction, conflict of interest, contravention of acceptance of advantages, .. etc), punishment (reprimand, reduction in rank, compulsory retirement, dismissal ..etc), and severity of offence (job-related, circumstantial factors, financial gain, abuse of official position, sexual harassment .. etc.)

In building the knowledge representation scheme of the case-based expert system, cases are classified into a structural hierarchy with the first node determining whether the case belongs to the "Criminal" or "Non-Criminal" classes. Within the "Non-Criminal" category, it is further sub-divided into two streams – the "CR56" cases and "CR57 cases" (see Figure 1). Under each stream the cases is classified by means of "Nature of Offence". Cases within the same category share some common features, e.g. in criminal cases, more detail information is stored in each case, and whist in non-criminal cases, less information is required.

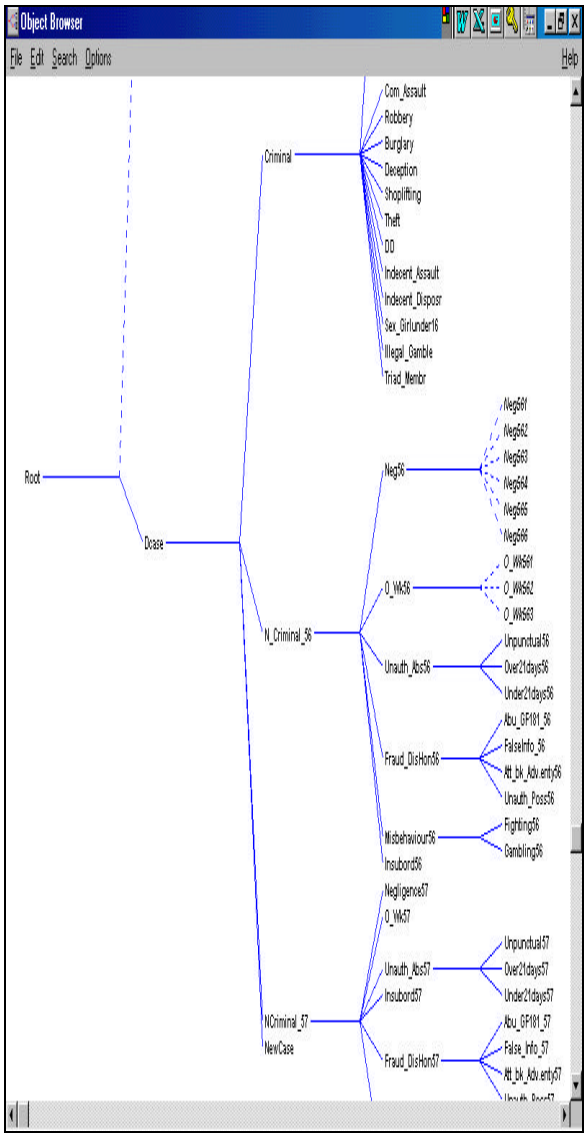

Figure 1: Object hierarchy with property inheritance

Using such a hierarchical organization with inheritance, values can be assigned to objects down the hierarchy more quickly and efficiently. Furthermore, this hierarchical structure also facilitates the applications of method, function and rule modules developed. In total, 5 classes and 12 sub-classes of objects have been created, each with 20 to 30 slots. The user interface consists of 30 individual screens, with windows type icons as shown in Figure 2. Some explanation facilities were also built into the system to help the user to understand how the solution was derived.

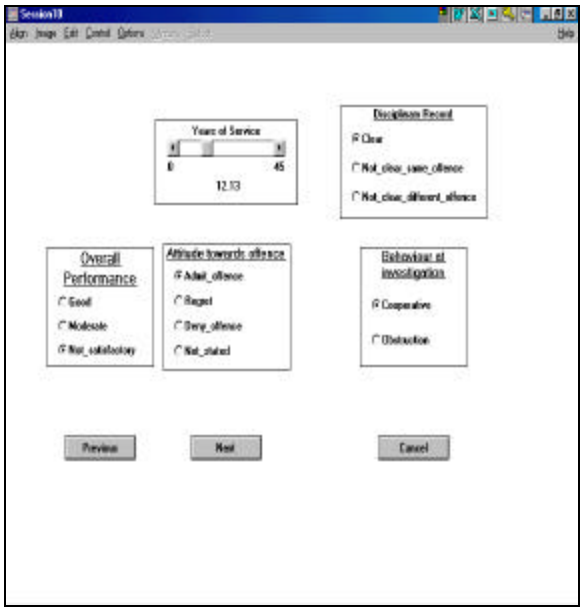

Figure 2: User Interface

3. Implementation

Matching and evaluation algorithms were written using the KAL language interpreter provided by KAPPA-PC [5]. First, each new case will be compared with all the cases in the case library using the primary features, i.e., job-relatedness, office hours, abuse of position, isolated or recurrence offence, consequence to Government. Then, secondary features will be compared with the potential cases to identify the best match, (see Appendix 2). The method used is the nearest neighbor algorithm [3,6].

Production rules are used to derive the adapted solution for the new case. A typical rule is:

IF Punishment_awarded = Reprimand
THEN debar_from_promotion
OR appointment_for X years

where the value X depends on the current policy being enforced. Therefore, the punishment recommended reflect the most up-to-date information, which was being held within the database. This information could be stored as global information.

4. Testing and Evaluation

Testing [8] is done by comparing the results of the existing manual system and also by potential users. All together, 15 officers have evaluated the system in detail. The results showed that the system not only can help them to analyze the problem, it also assists them with understanding the scenario. Further analysis of the results reveals an interesting pattern, i.e., respondent's perception and acceptability of expert system technology increased with their degree of computer literacy. Another observation is that there is only a slight variation of responses between the experienced and inexperienced officers.

5. Conclusion and Future Enhancement

An object-oriented case-based expert system was developed to award punishment for serious discipline cases in HKCS. Cases were organized into an object-hierarchy with property inheritance, KAPPA-PC was used to implement the design. The system also has a user-friendly interface, and testing was done by comparing with existing manual process as well as through user assessments.

Future development includes the matching of qualitative information among the cases such that the matching process resembles the experienced officers. An extra module "case-establishment" could also be added to the current system, its function is to determine whether the evidence surrounding a misconduct case is sufficient enough to call for formal disciplinary actions. This could be a very challenging job giving the fact that establishing a "Case" is a highly non-trival task.

6. Acknowledgment

The author would like to thank the three anonymous reviewers for their very useful comments and suggestions for this paper. Furthermore, thanks also to my MSc student Louise Mok for her effort in programming the expert system, and also provides the case information.

Reference

[1] Bain, W. M., *Case-Based Reasoning: A Computer Model of Subjective Assessment*, Yale University, May 1986.

[2] Ashley, K. & Rissland, E., "A Case-Based Approach to Modeling Legal Expertise," *Journal of IEEE Expert*, Vol.3., 1988, pp. 70-77.

[3] Kolodner, J.L., *Case-Based Reasoning*, Georgia Institute of Technology, 1993.

[4] Gebhardt, F., Vob, A., Grather, W. & Schmide-Belz, B., 1997, *Reasoning with Complex Cases*, Kluwer Academic Publishers, 1997.

[5] Helton, T, Software review – Kappa PC, *AI Expert*, 6(3) 91, pp65-67.

[6] Lenz, M., Bartsch-Sporl, B., Burkhard, H. & Wess, S., 1998, *Case-Based Reasoning Technology: From Foudations to Applications*, Springer-Verlag.

[7] Mok, L.W.K., *An Expert System for Awarding Punishment for serious discipline cases in the HKCS*, MSc dissertation, HK PolyU, 1995.

[8] O'Leary, D., "Verification and Validation of case-based Systems," *Journal of Expert Systems with Applications*, 6(1), 1993, pp. 57-66

[9] Watson, I. & Perera S., 1997, The Evaluation of a Hierarchical Case Representation Using Context Guided Retrieval, in *Proceedings of the 2nd International Conference on case-based Reasoning, ICCBR-97*, pp. 255-266.

[10] Watson, I., 1997, *Applying case-based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann Publishers.

Appendix 1

Questions asked during the user interviews:

1. What is the existing procedure for processing discipline cases?
2. Are the existing procedures comprehensive enough to enable you to process cases confidently and independently?
3. What things you look for if cases are passed to you?
4. How to gather the necessary information for the case?
5. How to decide whether there is sufficient prima facie evidence to lay a charge against the accused officer?
6. How important is the part played by precedent cases?
7. Is it easy to identify relevant precedent cases for citation and reference?
8. Are the retrieved precedent cases able to meet your need?
9. Is there a faster method for case retrieval?
10. On what basis is punishment being awarded?
11. Whether there are general rules guiding you to come up with the appropriate form of punishment?
12. What are the common problems you faced?

Appendix 2

CHECK input case for parent class attribute (i.e., criminal or non-criminal)
    IF criminal GOTO criminal sub-class ELSE non-criminal sub-class
    IF further sub-classes exist, perform sub-class matching until final sub-class is reached
COUNT no. of cases under sub-class
FOR case 1 to case N in sub-class
DO
    COMPARE the five critical features of the new case with those of the old cases
    IF match, PUT the case no. in counter
UNTIL no match find
FOR each match case
DO
    GET case details
    COMPARE minor features and COMPUTE score
UNITL EOF
SORT cases according to scores
DISPLAY the best 5 matches for user
COMPUTE adaptation values for the new case using production rules.
    e.g. (IF Puishment_awaded = Peprimand THEN debar_from_Promotion / appointment for 2 years)