MNK Publication

International Journal of Latest Research in Science and Technology **ISSN (Online):2278-5299**
Volume 3, Issue 2: Page No.1-6 ,March-April, 2014
https://www.mnkpublication.com/journal/ijlrst/index.php

COMPUTER FAULT DIAGNOSIS SYSTEM USING CASE-BASED REASONING APPROACH

Hanisah Ahmad
University of Technology MARA (UiTM), Perlis, Malaysia

Abstract*- Computer is an electronic machine that can be used as a helping task for an individual or organizational purpose. Many problems can arise if there is a corruption or failure that happened to the computer, such as incomplete of work and also communication problems. Therefore, maintenance and repairing should be done. With the development of the Computer Fault Diagnosis (CFD) system, it can detect any corruption or problems that happened to the computer and indirectly the user also can maintain their computer by themselves. The approach of Case-Based Reasoning (CBR) has been used to implement the system by using the previous experience to solve the current problems. This paper also explained about the main component in CBR technique such as inference engine, case representation and case base. There are used in this system included architecture and designing.

Keywords **-** Case-Based Reasoning; Case Base; Inference Engine, Case Representation

I. INTRODUCTION

Computer technology is rapidly used. No wonder many are racing to have their own computer. Especially to realize globalization indeed makes the electronic media as a top priority. Computer technology is also getting old will undergo changes as a result of new technologies will emerge and the next computer designers to produce better in the interest of competition. But at the same time, computer has certain limits where it can run into problems such as computer failure. Often, when this happens, the computer user will be sent to computer service.

However, there is also a wish to repair computer by own self. But the constraint is they did not have the opportunity to learn from people who are more skilled or expert in the field. The time allocated to learn from the experts is limited and often to particular interests. With this initiative to build a system that can represent experts deemed necessary due to the requirements. The areas involving a specialist known as Artificial Intelligence (AI) in which the data is derived from the fact of expert based on the experience they have. CFD Systems (Computer Fault Diagnosis) was developed to implement this requirement. It represents the experience of experts in the field to detect computer malfunction. The system provides the freedom for users to submit questions and problems. Solving the problem will be addressed to the user so that the user can identify what the real problems faced. If the problem persists, the user would be able to identify the cause and solution.

By this time, money and energy can be saved or reduced. This system is based on CBR technique (Case-Based Reasoning) in which the problem solution based on case, and the knowledge base contains certain cases.

II. METHODOLOGY

The methodology is a combination of paradigms, rules, policies, procedures, regulations, standards, techniques, tools, programming languages and other methodologies used to analyze and detail the requirements and system design. In developing the software, things that need to be considered is whether the methods, techniques and tools to be used for the smooth running of developing software. To develop this system, Case-based Reasoning (CBR) approach is applied.

A. *Case-Based Reasoning*

CBR made human as a model because it is more like or imitate a human reasoning process [8]. People need information to understand a thing and hence can conclude from this understanding. Understanding the process is said to be a script where it refers to the conditions and situations that occur an event. With the aid of an existing script, people already know what is supposed to happen and does not need a lot of thought because it retained certain scripts describing a situation. These scripts is said to be an experience when someone already knows it and stored in its own memory. This experience will be used in case the same thing at other times. With this, the CBR has used the concept of this script to reflect the situation and stored in a memory which is called the base case.

Reasoning based on case or Case-Base Reasoning, CBR works by selecting cases from a database containing cases and cases with features or criteria that are identical or very similar to the characteristics of new problems to be solved. It is like the way an expert working. For example, a doctor faced with a patient should be given a certain medication.

The quantity of medication to be given must be determined first. This is done by remembering past experiences in which doctors have treated other patients suffering from the same disease within a certain period. Solve problems by reference to the existing problem solving stored in the knowledge base or commonly known as the case base.

The solution to the existing problems stored in the base case is directly applicable to a new problem or repaired as appropriate. New cases have been resolved will be saved in

Publication History
Manuscript Received : 24 April 2014
Manuscript Accepted : 28 April 2014
Revision Received : 29 April 2014
Manuscript Published : 30 April 2014

the base case to be referred to when faced with a new problem that is very similar in the future. Because CBR uses past experiences from the base case then it is considered as reasoning based on experience.

1. *CBR Cycle*

In Figure 1 shows, CBR involves the following four cycle [5][6][8]:

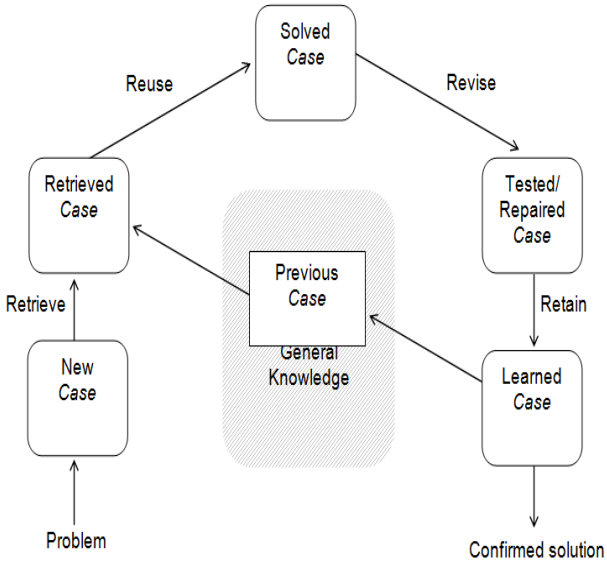

Figure 1. CBR Cycle

i)	Retrieve	The process of retrieving similar cases or very similar from the knowledge base
ii)	Reuse	The process of using the information and knowledge in that case derived from the knowledge base to solve new problems.
iii)	Revise	Review all proposed solutions
iv)	Retain	Save solutions to new cases for future reference case

New problems will be solved by the steps above in order. But there are also systems that do not use or process steps above, especially in the process adapt to revise a custom solution using previously, before it is stored in knowledge base.

B. *Phase Approach*

In addition to the phases already mentioned, the need for more specific phases of the technique or approach used is to be specified. This could reflect the scope of the approach itself can be applied in the system to be developed. With this, the number of phases of the CBR approach has been studied to develop the CFD system. The Figure 2 shows the phase of software development using CBR.

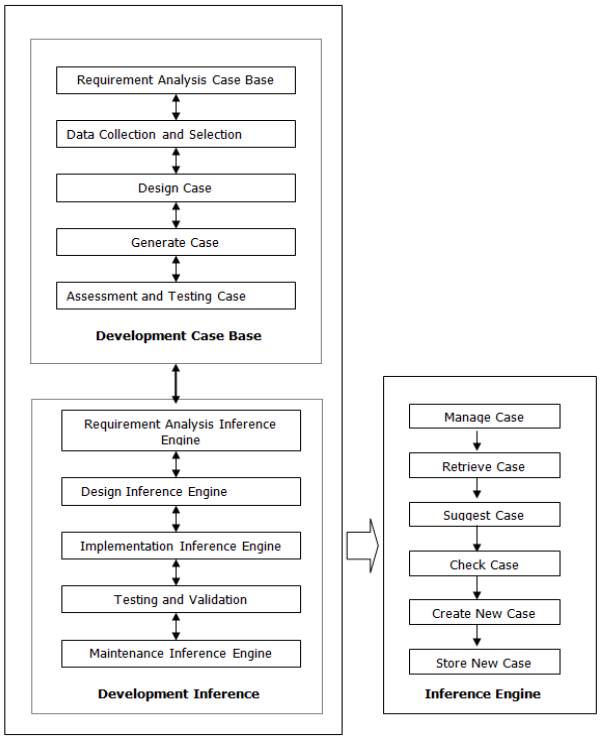

Figure 2. Phase Diagram Software Development

In CBR there are two main parts that need to be emphasized that the development of the case base and inference engine [1].

1. *Phase Development Case Base*

This involves a number of stages to be followed to develop a base case. The first step is to do a needs analysis base case in which initially is to identify what information or requirements that must be included in the system. In this system, the study of the domain you want to use are identified which relate to damage to computer hardware. Information collected by identifying the causes or problems and gets the road solution. Classification of data is done so that the data or information obtained is in accordance with the requirements of this system.

The next step is to try to design a case that you wish to store in the base case that reasoning engine access is easier. In this case the determination of the type of representation that can be used to do. For this system, the representation of cases that can be used is in the form of a list where it is sorted according to its own characteristics or criteria that allow easy access. The final step in this phase is to generate a case in a particular form and to test the effectiveness of the generated cases. For this system, cases are arranged in series or sequences to facilitate reasoning engine search. Testing is done with trying to achieve the effectiveness of the case through reasoning engines developed. If the solution is not accurate, then the modification should be done to reach the accuracy of the case is more effective.

2. Phase Development Inference Engine

In developing an artificial intelligence system, the key to development is inference engine. It reflects a process or activity in a system. As in the base case development phase, this phase has a certain number of levels. Initial stage is to identify or analyze what needs to be there in the reasoning engine. This depends on the type of system to be developed.

This system involves the acquisition of the case which is includes access to the search form, the review of the case and create and save a new case for the criteria specified. The next stage is to design a reasoning engine. For this CPS system, inference engine developed by isolating each process in the form of modules such as search modules, review and storage and display. Each module has its own activity on the system to be developed. Once the detailed design phase of the implementation will be done then implement all the design form into coding.

Once all programs have produced the testing or validation inference engine done to ensure that every process running right specifications. This is done by testing to each module, including the relationship between the modules. In maintenance phase, repairing need to be done if any unreliable event or error detected in any part of the system. This stage is very important in this system to ensure that no errors occur and the access that is done to meet the needs and specifications of the actual system. Each phase has a specific goal or objective to be achieved.

The summary of the objectives and outcomes of each phase obtained from this system is presented in Table 1.

Table1. Summary of Objectives and Outcomes for Each Phase

PHASE	OBJECTIVE	ACTIVITY	TECNIQUE	RESULT
1	Understand the problem and choose the right approach.	Literature Review Problem Analysis	Case-Based Reasoning (CBR)	Literature Review Report Computer problem specification
2	Determine the activity of the development phase	Classify the activities of each phase	Prototype Methodology	The detailed planning of system development
3	Getting requirement and system specification Getting system design	Collect the information of computer failure Model the problem Designing systems	State Transition Diagram (STD) UML-use case diagram, sequence diagram and activity diagram	Specifications and system requirements Detailed design system
4	Develop base case Develop reference engine	Development base case Development reference engine	Based on List Matching	Base case Search engine and case reasoning Prototype system
5	Verification System	Testing the system	Testing of the system	Complete system
		Maintenance system	output	Reporting system

C. *Compare And Contrast*

The differentiate concepts between knowledge-based system and Case-based Reasoning are described in Table 2.

Table 2. Knowledge-Based System Vs Case-Based Reasoning

KBS	CBR
Acquisition of general knowledge	Require less general knowledge
Many dependencies between rules	Cases are independent from each other
Implementing KBS is a difficult process requiring special skills or complex and often taking many years	Implementation is reduced to identifying significant features that describe a case, an easier task than creating an explicit model
Implemented kbs are often slow and are unable to access or manage large volumes of information	By applying database technique largely volumes can be managed
Once developed they are difficult to maintain	By acquiring new knowledge as cases thus making maintenance easier.

III.CASE DEVELOPED

Associated with the development of case-based reasoning system, some of components concern in developing this system of case representation, case base and search techniques are used.

A. Case Representation

To develop the base case, the selection of the delegation of the case (Case Representation) is to be considered. In general three main components to create a case that Goal will provide a solution to a problem, Constraints (Constraints) of Goal and also the characteristics of the problem or situation and the relationship between them. For the CPS system, the delegation of the case was based on a list. Each list represents the specific cases of the problem and the solution of its own. In each of the lists have some sub lists and also a few atoms or parts arranged in a particular order. The arrangement of syntax is depending on the programmer designing. The example of a case for computer failure problem is shown in the Figure 3.

```
(70 0.3
 (BIOS CAN READ DISC IN  FLOPPY DRIVE  A  BUT SYSTEM CANNOT
  BOOT FROM IT.)
 (USE THE RIGHT BOOT-DISC  OR BOOT WITH DIFFERENT DRIVES.)
 (INVALID BOOT DISKETTE))
```

Figure 3. Sample Case

Based on the Figure 3 in above, each case in the list is divided into 5 parts which may have a distinctive role. Atom or the first element of the list, 70 is the index number to represent a case. The second element represents the accuracy of the case. Example 0.3 in which case the problem in the present case to match the case with the accuracy of 0.3 or

above, then the case will be accepted and proposed to solve the problem. The third element is a sub list of the cause to a problem in the case. The fourth list is the solution to the problem in the case. The last element was a problem that will be used to compare the current case. All cases are isolated and grouped with its own list or included in a list which facilitate inference engine to search or access.

B. Inference Engine

Inference engine development involves a number of concepts or cycle used in case-based reasoning approach in the recovery (retrieve), use all (reuse), review (revise) and also keep (retain)[2]. However, there are some parts of the cycle are not be applied in this system (CFD), namely the modification of (adapt) and maintian (repair) as well as new cases are automatically saved. Though the system is not able to generate new adaption cases, but the system is still be able to generate new cases to reuse the solution of the existing cases and stored in the base case. That means the system is simply not save or generate a new case with a new solution.

For the recovery process, the search method used is a serial search [7] because it uses the list by doing a search on each element in the list. In terms of matching a case is dependent on the accuracy specified in each case. This means that if the current case by case matching the previous count value equal to or greater than the value of accuracy, so then the case can be re-used to solve the current case is. The choice is based on the high accuracy of the accuracy of these. Reuse of solving a case is to duplicate solutions found in the previous case and displayed to the user.

The review process is done on a case chosen by the user to solve the problem. If the difference between the current case and previous cases is little or accuracy values of 90% and above, then generating a new case in which the evidence will be problems when combined with the previous problem solving and re-copy the previous case rulings. With this, the new store will be the case for storing new cases generated in the base case. The index of value for new cases will be generated by the system by increasing the number of existing cases. Index used for centralized representation, memory and the recovery of cases [4]. The case will be restructured in order in the new case will be placed at the end of the other cases.

IV. ANALYSIS AND DESIGN

This phase determines what must be done by the system to identify in detail the relevant functions. For the system to be developed, the analysis of the problem has been made to identify how the problem is to be solved. Figure 4.1 below shows the complete cycle of CBR implement the system in this CFD.

A. System Architecture

CFD system architecture diagram gives an overview of the system developed on computer failure diagnosis problem by using Case-based Reasoning approach. Based on this figure, there are five major components or processes that make up the architecture of the system, namely: -

1. *Managing Case Module*

It manages all input and output data, and an interface between the CPS system and the user. Input data is data or information obtained from user input of the questions posed by users. The output data are the results obtained from the system and that will be displayed to the user.

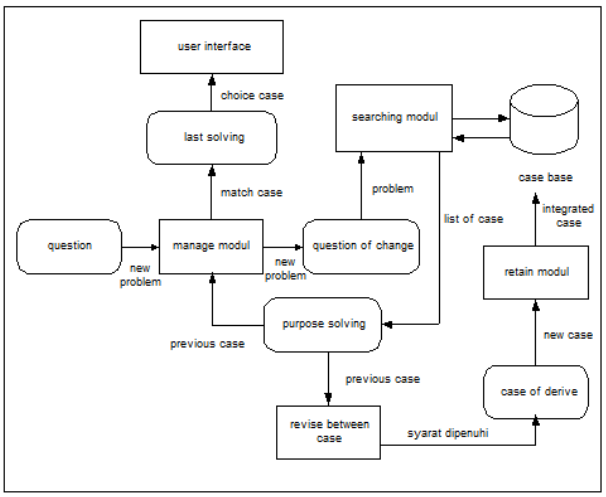

Figure 4. CFD System Architecture Diagram

2. Searching Module

This module is a key process in the system where it will always interact with the base case for an appropriate case to the present case and also contains a search engine. CBR is like query of the database to search for a particular problem or solution [3]. Search techniques used are serial search.

3. Revise Module

A module or process that will review the current case and the case obtained from the base case to determine whether the case can be saved as a new case in the base case or not.

4. Retain Module

A module will keep (retain) new cases in the base case. In addition, it will increase the value of the index in the base case. This module will also be restructured or update the base case in order to be true in order to facilitate access for the next do.

5. Display Module

A module is involving the user display. Its function is to display all the previous cases that match the question. The user is allowed to choose the output cases and the system will display the detail result including the reasoning and suggestion of solution.

B. Calculation Of Accuracy

To ensure that a result is correct or not, is dependent on the probability of matching the current case with previous cases found in the case base. This system will test each element *(si)* that are listed in the question and compared to each element

in the list of cases *(ki)* in the case base. If there is a match *(p)* or match the system will automatically increase the probability that the mean value set is 0.05 for each matching elements. These values will be added to all elements of the revised questionnaire. Elements of the question also tested on the functioning of the hardware test for determining the presence of a hardware key *(u)* corresponding. If the values are match then added with 0.2.

The values obtained will be compared with the probability that the case has been fixed on the case base to allow the case received or matching. If the total value is equal to or greater than the value in the base case then the probability above will be increased by 0.5. If, however, no additional impairment. The last value is the total addition and will be kept in each case there is a list of the base case to facilitate access for you later. With this formula used is:

$\mathbf{kb = \Sigma\, p(s_i k_i) + u}$ where kb= total value of probability

@ s_i = element question of i

$\mathbf{kb = \Sigma\, p(s_i k_i) + u + 0.5}$ k_i = element case of i

p = matching between s and k elements

i = 0,1,2…..n

u = main key of probability (0.2)

V. IMPLEMENTATION AND RESULT

This phase discusses the implementation of the system developed and the results obtained from this project. This involves architecture, coding system and test data together with some examples of the implementation of the specific results for each exercise performed.

A. Sample Input And Output System

The Table 3 in below show the example of input and output for the CFD system.

VI. CONCLUSIONS

In this paper, Computer Fault Diagnosis System based on CBR (CFD) is briefly explained. This system can be beneficial to all, especially in solving problems related to computer failure. It also serves as a support to the development of systems that want to use the same approach but different domain. All the shortcomings that are present in this system is expected to be further improved and enhanced specifications for the production of more accurate and efficient for all. It can also be used commercially either to a specific individual or a business. The problem of lack in Information Technology (IT) can be reduced and user can get more knowledge and experience from this system.

VII. FUTURE WORK

Goals and objectives are generally achieved. But there are still some flaws in the system. The system can not save new cases interactively or automatically if the problem posed by the user is not found in the case base. In terms of repair or adapt can not be done completely. By using the suitable theorem will significantly improve the accuracy of the reach of the case rather than using the usual formula. The system is also limited to a specific for computer hardware only. The deficiencies in the system are expected to give the initiative to other developers to fix these to use CBR approach or develop such a system.

Table 3. Sample Input and Output

Input	Users enter the problems that are related to computer hardware. The search process starts when the user presses the enter key to show the end of question. Some constraint that need to be followed by the user, in which: - ❑ Cannot enter a full stop '." ❑ Cannot enter a comma "," It is to avoid any error syntax happened. Example of question , *Computer cannot boot* User will enter 1 until several cases which successful matched with the question and system will display the chosen case. Example of choice: *Please choose 1 case in the following list between 1 until 2.* *Your choice? : 1*
Output	The system will list the successful case match the questions posed in sorted as shown below. *The total of 2 Case identified...* *1 - CASE-16 with probability : 0.95* *2 - CASE-17 with probability : 0.9* This shows that the system will display an index of successful cases along with the probability of accuracy of the access. Users are given the option to choose the case that you want displayed. For example, Case 16 in which the user needs to specify is 1 to allow the system to post the solution on a case-14. The system will display as below: - *Your question : (COMPUTER CANNOT BE BOOT)* *- CASE-16 -* *The cause of the detected: (COMPUTER CANNOT BE BOOT FROM DISC DRIVE* *HARDDISK BOOT FAILURE NON SYSTEM DISC)* *The solution proposed: (MAKE SURE TO CHECK YOUR CMOS SETUP PARAMETERS ALL DRIVERS ARE IN THE RIGHT PLACE. THEN TRY TO DO AUTODETECT HARD DRIVE THEN SAVED AND REBOOTING)* *Probability percentage : 95.0%*

ACKNOWLEDGMENT

We like to express sincere appreciation and deep gratitude to all participants and give full cooperation in this work.

REFERENCES

[1] Mariana Maceiras Cabrera, and Ernesto Ocampo Edye (2010). "Integration of Rule Based Expert Systems and Case Based Reasoning in an Acute Bacterial Meningitis Clinical Decision Support System." (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, No. 2, 2010, ISSN 1947-5500

[2] D. Aha, L.A. Breslow and H. Munoz-Avila (1999, to appear). " Conversational Case-Based Reasoning." Journal of Artificial Intelligence.

[3] B. U. Haque, R. A. Belecheanu, R. J. Barson, K. S. Pawar (2000). "Towards the Application of Case Based Reasoning to Decision-Making in Concurrent Product Development (Concurrent Engineering)", 2000 Elsevier Science B.V.

[4] Kolodner J.L. (1993). "Case-Based Reasoning." Morgan Kaufmann Publishers, Inc., San Mateo, CA. ISBN 1-55860-237-2.

[5] Althoff, K.-D., Wess, S., F. Schmalhofer, G. Strube and Wetter, T. (1992). "Case-Based Reasoning and Expert System Development ." Contemporary Knowledge Engineering And Cognition, Springer Verlag, pp. 146-158.

[6] Janet L. Kolodner (1992). "An Introduction to Case-Based Reasoning" Artificial Intelligence Review 6, 3--34, 1992.

[7] Navichandra, D. (1991). "Exploration and innovation in design:towards a computational model." Springer Verlag, New York. NY, US.

[8] Schank, R. (1982). "Dynamic memory; a theory of reminding and learning in computers and people. " Cambridge University Press.