A Mixed-Initiative Call Center Application for Appliance Diagnostics

William Cheetham

General Electric Global Research, 1 Research Circle, Niskayuna, NY 12309
(cheetham@research.ge.com)

Abstract

A mixed-initiative system was created to improve customer support at General Electric Consumer Products. The tool helped call-takers solve customers' problems by suggesting questions that can be used to diagnose the customers' problems. The mixed-initiative system provides human-like initiative that improves the correctness of the diagnostic process, speed of the process, and user satisfaction with the process. This has been in use since 1999 and has provided over $80 million in financial benefits to General Electric by increasing the percentage of questions that could be answered without sending a field service technician to the customers' home.

Introduction

General Electric (GE) Consumer Products provides a variety of customer support services over the phone and on the web for all household appliances sold by GE. One of the phone services GE provides is a group of 300 field service call-takers who schedule field service personnel to visit customers' homes. There are around 1.4 million home visits that are scheduled per year. In the past, about 20% of the time a field service representative arrived at the home all that was needed was to educate the customer. This education could usually have been done over the phone saving time for the customer and field service representative. However, the field service call takers were not trained to diagnose and explain issues over the phone. This training was difficult because of the large number of appliances that can be serviced, complexity of modern appliances, and the high turnover in the order takers. The solution for this was to create a software tool that acts as a mixed-initiative (Allen, 1999) assistant for the call-takers.

System Requirements

The primary goals of the call-taker are the correct diagnosis of the appliance, speed of diagnosis, and customer satisfaction with the diagnostic process. The purpose of the software tool is to improve performance on these goals. In order to do this there were multiple requirements for the software tool. These requirements are similar to the principles of mixed-initiative user interfaces described by Horvitz (Horvitz, 1999). The requirements are

- Show human-like initiative
- Suggest correct questions and initiative
- Increase speed of diagnosis process
- Increase user satisfaction with process
- Explain to the user why the initiative was taken.
- Have a process for the tool to learn from its experiences and adapt to a changing environment.

The next section will describe the system we created. The following section will describe how the system requirements were met.

System Architecture and User Interface

Case-Based Reasoning (CBR) (Aamodt and Plaza 1994) has been used to automate customer support help-desks by many companies such as Compaq (Acorn and Walden 1992) and Broderbund (Watson 1997). Using CBR for customer support (Simoudis, 1992) has become of interest to many other companies with help desks. We also used a CBR tool to assist GE Consumer Products customer support personnel. This system allows for mixed-initiative between the call taker and the automated assistant.

The tool we selected was k-commerce from Inference. Inference was later purchased by eGain. One of k-commerce's features is that questions can be attached to cases. The questions allow the appropriate case to be selected by answering the questions that differentiate the possible cases. For our application we wanted to be able to suggest one and only one case as the solution. A way to guarantee that these questions can differentiate every case in the case base is to form a tree of questions over the case base where each case is a leaf in the tree and each internal node in the tree is a question. We consider this approach as both a decision tree and CBR methodology. The decision tree is simply the selection mechanism of the CBR system. Figure 1 shows a decision tree where rectangles are questions, parallelograms are answers, ovals are solutions, and circles are indicators that the decision tree is continued on another page.

Copyright © 2005, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

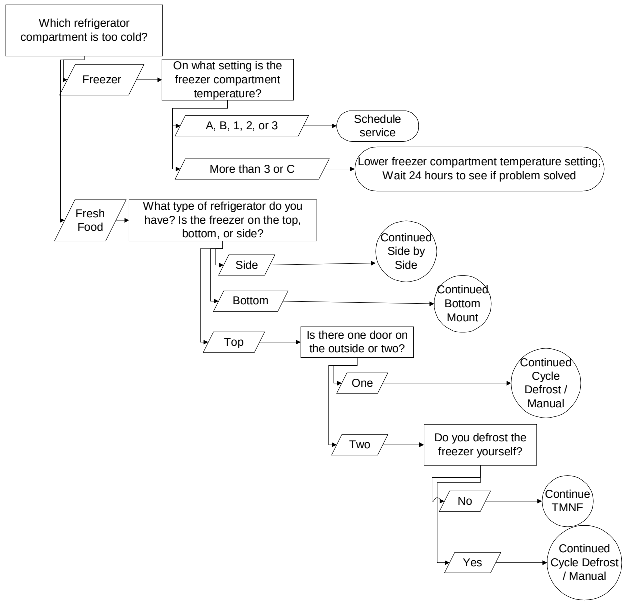

Figure 1: Decision Tree

User Interface

The user interface of the system is shown in Figure 2. The questions that the assistant suggests are listed at the top of the screen. There is a pull down list with the set of possible answers to each question. The call taker can answer any of the questions in any order. The bottom half of the screen lists the previously answered questions, which can be changed at any time. When a question from the top half is answered it moves to the bottom and the top is repopulated with new questions.

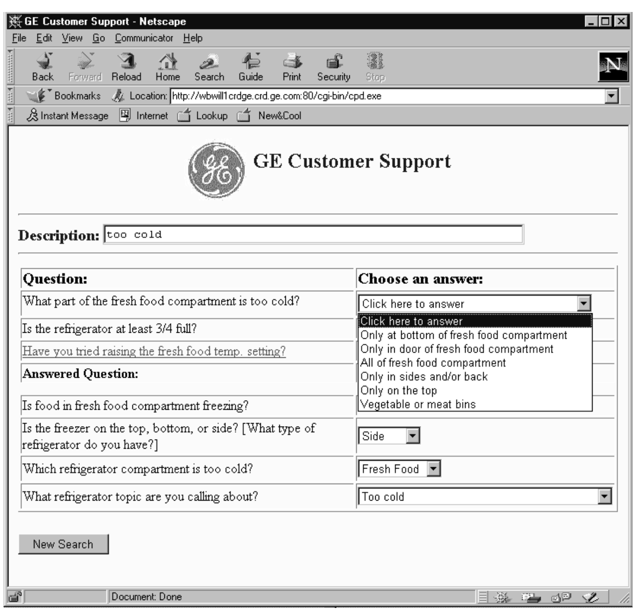

Figure 2: Internal Pilot User Interface

Mixed-Initiative Requirements

The system described was designed to meet the requirements for a mixed-initiative system.

Human-like Initiative

Humans often show initiative. However, they need to be careful to take initiative when it will be useful and not perform the initiative if it will not be useful. One way humans use to determine if initiative is warranted and then take the initiative is the following.

1. Have understanding of goals and priorities
2. Have understanding of current situation
3. Identify a task from current situation that can help goals
4. Identify potential benefits and problems from doing task
5. Determine confidence that task should be performed
    a. high – do task
    b. low – don't do task
    c. medium – ask if task should be done
6. Have ability to perform task
7. Inform others task has been done

Items one, two, and three are needed to correctly identify opportunities for taking initiative. Item four gathers information about the advantages and disadvantages of tacking the initiative. Care must be taken to identify the disadvantages. Possible disadvantages include

- Bothering the user
- Doing an unwanted task
- Doing a low priority task instead of a high priority one
- Being destructive or wasteful

The fifth item is weighing the advantages and disadvantages. If the advantages greatly outweigh the disadvantages then the initiative should most likely be taken. If the disadvantages outweigh the advantages then the initiative should not be taken. If neither of the advantages nor disadvantages outweighs the other then a human can ask another person (e.g., supervisor) if the initiative should be taken. The human needs the ability to perform the task and should inform others effected that the task has been completed.

Software assistants can use the same set of steps when they attempt to show initiative. They should understand the goals, priorities, current situation, and set of possible initiative actions. They should be able to identify advantages and disadvantages. Disadvantages of an automated assistant taking imitative include

- Bothering the user
- Locking the user out (stealing cycles)
- Acting in an unknown way
- Undoing users desired actions

- Making the user undo the computers actions
- Keeping the user from doing tasks (2001's HAL)

The system should be able to determine the confidence that the initiative should be taken (Cheetham and Price, 2004). Then, the system needs to take the action and inform the user of the action taken.

Correctness

We found that call-takers quickly lost trust in the system if it gave poor suggested questions or initiative. That is why we went though the additional effort of creating a decision tree on top of the case base. K-commerce does not require a decision tree; it can automatically suggest questions from the most likely cases. However, at the start of the diagnosis process it is often unclear which cases are most likely. We created diagnostic trees for each possible symptom of each type of appliance to improve the correctness of the system and retain the users trust. Figure 1 shows the tree for a side-by-side refrigerator where a compartment is too cold.

Multiple methods can be used to reduce the cost of an incorrect suggested question. Some of these are

- Provide multiple suggestions from which the user can pick.
- Do not force the user to act on the suggestion.
- Unobtrusively present the suggested questions to the call taker

Similarly, a false positive, where incorrect initiative is taken, is usually much worse than a false negative, where correct possible advice is not given. Because of this we concluded that the initiative taken by the automated system should only be given when it is highly likely to be correct. If initiative is taken and not correct then it could distract, slow, and/or annoy the user.

Speed

Before this system was fielded much of the information needed to diagnose a problem was in manuals and tacit knowledge of a few call takers. Customers would often be placed on hold while the correct manual was found and searched or the correct expert was available and informed of the issue. Having a system that could sore this information, provide it automatically, and guide the call-taker in the diagnosis removed many of these frustrating delays.

Another way the diagnosis can be done more rapidly is to have the system automatically answer the question that it is confident it can answer correctly. Some of these questions include information that can be obtained from other sources than the user. Information about the user, such as models owned and previous diagnostics sessions, can be stored in a database. When questions require this information it can be answered automatically. One future goal for high-end appliances is to include a phone or Ethernet connection so additional diagnostic data, such as a temperature sensor, can be sent directly to the diagnostic tool, allowing it to take initiative on a wider set of questions. It is important that the user can change the answers when a question is answered automatically.

The flow of control of the mixed-initiative system is shown in Figure 3. When a new call is received the system, represented by the Agent, searches for any questions that are needed for the diagnosis, automatically answers as many of these questions as possible, then passes initiative back to the user. The user can now take action to answer as many questions as needed. After each question is answered the agent can update the list of questions, automatically answer new questions, or do nothing. The call-taker is not forced to answer any questions and can jump to a competed diagnosis at any time during the question answer process. We try and have the agent do as much of the work as possible, so the process is as quick as possible.

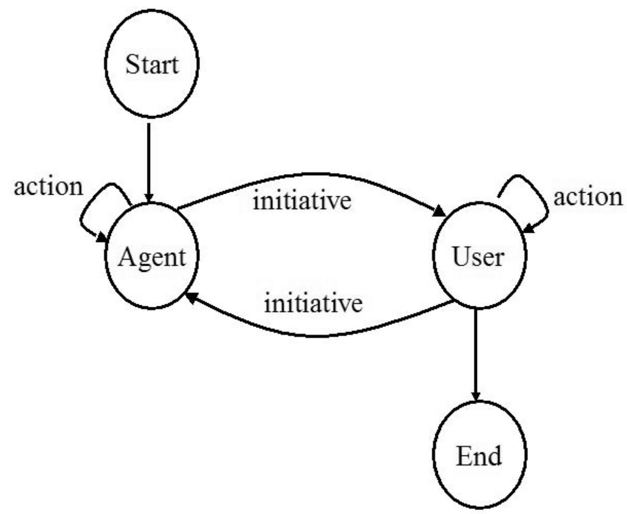

Figure 3: Mixed-Initiative Flow

Finally, the action suggested after diagnosis is completed can be automated. The call-taker could step the user through a pre-created repair process, or email / fax the process to the user. If parts are required, one click off the solution can place an order for the parts to be sent to the address of the customer. If a service technician is required, the time for that visit can be scheduled and description of the problem automatically sent to the technician for their review.

User Satisfaction

The customers' satisfaction is primarily based on the correctness, speed, and professionalism of the help provided by the call-taker. We have already described how the system helps the correctness and speed of the process. The system also helps the professionalism of the call-taker by providing the exact wording that should be used to ask a question to the customer. Each call taker is given two weeks of training before they are able to take calls from a customer. This training used to involve education about the appliances and how to professionally interact with a customer. Now, it is only on professionalism and the system acts as “on the job” training about the appliances.

Explanations

Call-takers and customers often wonder why the system is suggesting a specific question. The user trust is enhanced if there is a clear explanation for why the system is taking some action. When the questions are created we often also create an explanation for why the question should be asked. This explanation can be displayed for the call-taker by clicking on the question in the user interface.

Maintenance

Every time the case base was used to answer a call a description of the call was written to a reporting database. The description included start time, end time, caller phone number, call taker ID, type of appliance, short text description of issue, all questions asked by the system, all answer given to these questions, final diagnosis suggested, and if this answer was accepted by the caller. Each week a case author would analyze this information in the reporting database for every call that was taken for the week. The author was looking for any trends and especially any times a caller would not be satisfied by a suggestion. Any trends or outstanding items would be discussed in a weekly feedback meeting with the call-takers. The result of the feedback meetings would be a few changes in the case base. These changes would be made immediately and reviewed in the next weeks meeting. The call takers were a valuable part of the development team and felt ownership in the case base when the pilot test was finished.

Results

Before the system was implemented the call takers were only using their personal experience and paper documents to determine when a field service representative was or was not needed. Even though feedback from service technicians indicated that 20% of the calls could be answered on the phone, they were only successful on 3.9% of the calls. A cost benefit analysis of this problem showed that creating a mixed-initiative system would cost one million dollars for software, development time, domain expert time, and ongoing maintenance costs for the first five years, but the benefit from answering just 8% of calls was over two million per year.

The system was deployed to the 300 order takers in June of 1999. The percentage of calls that could be correctly answered over the phone, without sending a field service representative, increased from 3.9% to 12.3%. The average call times decreased slightly, from 256 seconds to 232 seconds. Since each service call that is handled over the phone saves over $50 the system has resulted in a savings of over $80 million from 1999 to 2005. The cost of development, including time of developers and purchased software, was $1.2 million. Surveys of customer satisfaction also showed the customer preferred having their issues solved without the need for a house visit.

Conclusion

GE Consumer Products has been successful using this mixed-initiative system to provide customer support. The system contains exhaustive knowledge about appliance problems and the questions needed to diagnose them. The human call-taker provides some knowledge about appliances and professional interface between our customers and the diagnostic knowledge stored in the tool. The team of computer system and human call-taker performs better than the call-takers had before the system was created.

References

Aamodt, A., Plaza, E., 1994. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, AICOM, Vol. 7, No. 1.

Acorn, T., Walden, S., 1992. SMART: Support Management Cultivated Reasoning Technology for Compaq Customer Service, In IAAI 4, Cambridge, MA. AAAI Press / MIT Press.

Allen, J. (1999). Mixed-initiative interaction. IEEE Intelligent Systems, 6, 14-16.

Cheetham, W., Price, J., (2004). Measures of Solution Accuracy in Case-Based Reasoning Systems, Seventh European Conference on Case-Based Reasoning, Madrid, August 30 - September 2.

Horvitz, E. (1999). Principles of mixed-initiative user interfaces. Proceedings of the Conference on Human Factors in Computing Systems (pp. 159-166).

Thomas, H., Foil, R., Dacus, J., 1997. New Technology Bliss and Pain in a Large Customer Service Center. Lecture Notes in Computer Science, Vol. 1266. Springer-Verlag, Berlin Heidelberg New York. pp. 166-177.

Simoudis, E., 1992. Using Case-Based Reasoning for Customer Technical Support. IEEE Expert 7(5), pp. 7-13.

Watson, I., 1997. Applying Case-Based Reasoning: Techniques for Enterprise Systems. San Francisco, Cal. Morgan Kaufmann.