Case Based Query Processing in Manufacturing Technical Support

Maurice D Mulvenna[1], Jonathan R C Allen, John G Hughes.

NIKEL[2], School of Information Systems Engineering, Faculty of Informatics
University of Ulster, Newtownabbey BT37 0QB Northern Ireland. UK

Abstract

Case-Based Reasoning (CBR) is suited to problem solving in domains where there are recurring problems. This paper describes the development of a CBR prototype help desk system for use in such a domain, the Technical and Product Support department of an aerospace company. For the purposes of the prototype three types of indexing have been used, knowledge-guided induction, inductive indexing and nearest neighbour matching. Through the CBR prototype system, technical support staff now have the knowledge to tell if a new query is a recurring problem and has been solved before or is a completely new unsolved technical query, in addition to the system retrieving approximate solutions. Initial evaluation by the users of the prototype system has been favourable.

Introduction

Quality, dependability, professionalism and cost effectiveness are all key features in order for a company to sell their product. In the case where the product is a service, the same ethos applies. For this particular practical application the problem domain lies in the Technical and Product Support department of an aerospace company in Northern Ireland. The technical support is provided for a variety of short range commuter and military aircraft. These aircraft are in service in over 40 countries with at least 100 different operators. Although the company no longer manufacture many of the range of aircraft they are obliged to provide product and technical support to the operators of the aircraft. It is envisaged that the aircraft for which the product support department provide technical support will be operation for the next 10-20 years and will therefore require technical support for this period of time.

Product Support

The main function of the company's product support department is to provide responses to technical queries from aircraft operators. This often involves liaison between the operators and specialist departments within the company. The vast majority of technical queries from operators arrive at technical support in the form of a fax. The queries are usually of an urgent nature as aircraft which are grounded are costing operators money.

The scope of a technical query covers problems such as:

- Accident damage;
- Corrosion;
- Technical defect;
- Request for data.

Within technical support there is a wealth of information stored in maintenance manuals, flight manuals, crew manuals and previously solved technical queries that are continuously being updated as new technical queries are solved. The solved technical queries are filed under the ATA 100[3] specification.

---

[1] Contact email: MD.Mulvenna@ulst.ac.uk
[2] Northern Ireland Knowledge Engineering Laboratory
[3] Air Transport Association of America 100 is a set of standards which are used to identify particular systems within an aircraft.

The headquarters engineers use their expertise and knowledge of the domain to solve the new technical queries. Initially this may involve recalling a similar query that has been solved in the past. In this situation, they will go to the current filing system and try to locate such a similar case. The filing system works under the ATA 100 principle, where the aircraft is broken down into categories. An example of this is :- ATA 52 Doors, ATA 52-10 Passenger / Crew, ATA 52-20 Emergency Exit, etc. The previously solved cases found in the current filing system can be used as a starting point to aid in the solving of new technical queries. If this is a new type of problem, the headquarters engineer will then use his own knowledge and experience to arrive at a possible solution. This may involve getting more information from the source of the query, using the reference manuals in the office or in the library, or contacting a specific department in the company. This may include contacting departments such as Engineering- Systems, Design / Structural, Air worthiness, Aerodynamics / Weights, Stress, etc.

However if the headquarters engineer is unable to recall solving a particular type of query before or if it was another headquarters engineer who had solved the query before, this may lead to the situation where a time consuming search of the filing cabinet produces no possible answer to the query.

It is feared that the knowledge which the experts have developed in the solving of technical queries could be lost. Also, in conjunction with the need for fast response to all new queries, it is apparent that this is a domain where computer technology could be applied to assist the headquarters engineer. The fact that there is an abundance of previously solved technical queries already in the office filing system means that potential cases for a Case-Based Reasoning (CBR) system are extant. This paper describes the development of a CBR prototype for the product and technical support department using CBR software (ReMind), which enhances query processing in a RDBMS-based help desk system.

Case Based Reasoning

Case Based Reasoning compares a current situation (or case) with situations that have been encountered in the past to see if one or more of the earlier situations can provide a model for how to act in the current situation[1][2].

There are three basic steps in creating any case-based system:

1. Data must be examined, and the significant features identified;
2. Data must be acquired and represented;
3. Data must be indexed for efficient querying and retrieval [3].

Case features

In the area of product support, the solved cases are stored and filed under their ATA reference. This encompasses a vast quantity of potential cases and for the development of the prototype, it was therefore necessary to restrict the prototype to one particular ATA section for case analysis.

For a particular ATA reference section to be selected as a good example for the CBR prototype there were several conditions which needed to be met:

1. Degree of complexity. It was important that the ATA section selected showed a certain degree of complexity in the answer of technical queries i.e., ensuring that the solution to the query was not trivial.

2. Quantity. It was essential that there was a large case history of previously solved technical queries in the current filing system under the particular ATA selected.

3. Suitability. The ATA subset selected had to be representative of the problem domain.

After consultation with the senior headquarters engineer and the manager of the support department, ‘ATA 57 Wings’ was selected as the section on which specific exemplar cases could be derived. ATA 57-Wings possesses the desired attributes that were stipulated for the development of a prototype CBR system. This particular ATA subset requires an ‘Average’ degree of complexity for solving technical queries, it is neither too difficult nor too easy, as there is the need for particular domain knowledge. Technical queries in ATA 57 cover all the possible categories that can arise i.e., damage, corrosion, request, etc.

ATA 57 Wings has well defined sub-sections:

- 57-10 Wing Skin Corrosion.
- 57-20 Wing Strut.
- 57-30 Wing Tip .
- 57-40 Wing Cleat.
- 57-50 Stub Wing.

It was therefore clear that ATA 57 was a suitable reference section on which the prototype could be developed.

Case representation

The technical queries are only filed when they have been answered, so all cases in the paper filing system are completed cases. The potential cases contain information such as *operator name*, *subject of problem*, *detailed description of problem*, *staff member* who dealt with the query, *date received*, *reference Number (In)*, *detailed description of solution*, *and reference number (Out)*. It should be noted that one particular case could easily be made up of several correspondences (usually faxes) between the department's headquarters engineers and the operator with the initial query.

Although all of the above fields may provide good management information in the office environment for good house keeping purposes, it was decided from the offset that only those fields which would aid the retrieval process should be included in the case-base. Those fields that were not included in the case-base would be located in a relational database which interacts with the case-base. The fields which best assist the retrieval process i.e. fields on which indexing is performed, are as follows:

- ATA Reference;
- Aircraft Type;
- Category of problem;
- Key Words;
- Brief Description of Problem;

and two fields which do not affect the retrieval process but are necessary for the solving of new cases:

- Detailed Problem Description;
- Detailed Solution Description.

Once the fields for inclusion in the CBR system were established the next step was to represent them into cases. ReMind provides a feature called *symbols* by which data being represented can be ranked and classified into a hierarchical structure. The symbol hierarchy is a branching graph structure of parents and children and is used to provide the system with knowledge about the domain data [3].

The symbol hierarchy was used to represent the *Aircraft Type*, *ATA Reference* and *Category*. The main purpose behind using the symbol hierarchy was to ensure that the users input would be well structured and be of a consistent nature. The user would be given a multiple-choice to select the necessary information. *Aircraft Type* at this stage has only two aircraft options but for future developments this can be changed to allow for all aircraft types. The *ATA Reference* symbol hierarchy consists of the ATA 57 Group and the subsequent sub-sections of ATA 57. The *Category* symbol hierarchy includes the most common type of technical queries that arise i.e., Request, Repair, Damage, Corrosion, Modification (See figure 1).

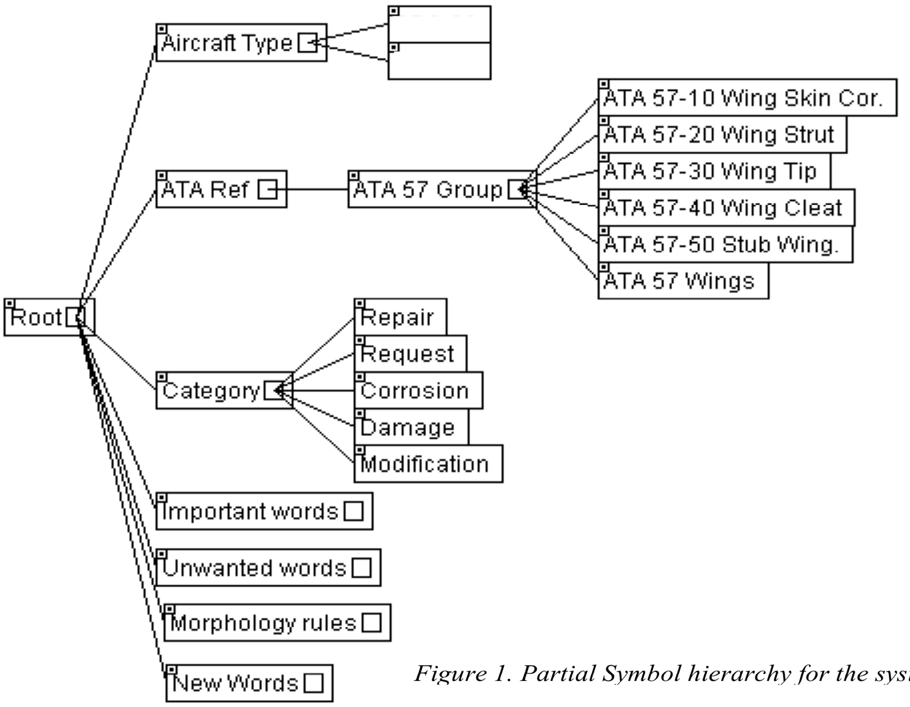

Figure 1. Partial Symbol hierarchy for the system

The free text that is inserted into the Brief Problem Description is parsed by a formula field to generate a list of symbol field called *Key Words*. The formula is derived from several symbols in the symbol hierarchy namely *Important Words, Unwanted Words* and *Morphology Rules*. The formula removes unwanted words such as *the*, *at* , *and*, etc.; to leave possible key words that describe the problem. The remaining words have a set of morphology rules applied to them to strip the plural endings of the word to leave only the trunk of a word e.g. ed -> e, ing->e. At this stage the remaining words are compared with a dictionary of important words, and if there is a match the word is added to the *Key Words* field. Any words which the system does not recognise are placed into a *New Words* symbol hierarchy.

The faxes that arrive at the company generally have a short sentence highlighting what the problem is, the vocabulary used is of a very distinct nature and generally refers to a particular section or part of the plane. Therefore whenever the headquarters engineers enter a natural language description of the problem there are only so many key words which will appear e.g. Corrosion to port and starboard wing to lift strut, lugs part number: X14/345/A-12 on aircraft RT045. The only ambiguities are words like starboard, for example, which may be entered as stbd. It is therefore necessary to ensure that all words entered are not short hand versions but complete words. This should not be to difficult to ensure as the number of headquarters engineers, who are the users, is limited to a small number, and application of established guideline may be ensured.

The general structure that has been adapted using the symbol hierarchy for case representation is ideal for any future additions of knowledge to the *ATA Reference* groups, *Category* and *Aircraft Type*.

Indexing the cases.

The ReMind software allows for four types of index strategies:

- pure induction,
- knowledge-guided induction,
- nearest neighbour matching and
- template matching [4].

For the purposes of the prototype three types of indexing have been used:

1. Knowledge-guided induction, which takes into account specific domain knowledge about an application before inductive indexing is done.

2. Inductive indexing, which is done via a process called clustering in which the system builds a decision tree based on features from past cases that discriminate between various outcome values [5].

3. Nearest neighbour matching, which allows retrieval of cases by comparison of a collection of weighted features in the input case to cases in the library [6].

For the prototype system the knowledge-guided approach was applied first. As the ATA Referencing is already well structured it lent itself to knowledge-guided prototypes. A prototype was created for the ATA 57 group and then for each of the ATA 57 sub-sections (Refer to figure 2).This allowed the cases to be partitioned according to a set of pre-defined criteria. It can be clearly seen from figure 2 that by using this method of indexing the cases are broken down into more manageable quantities. Whenever the remaining ATA references are added this will become an even more profitable exercise.

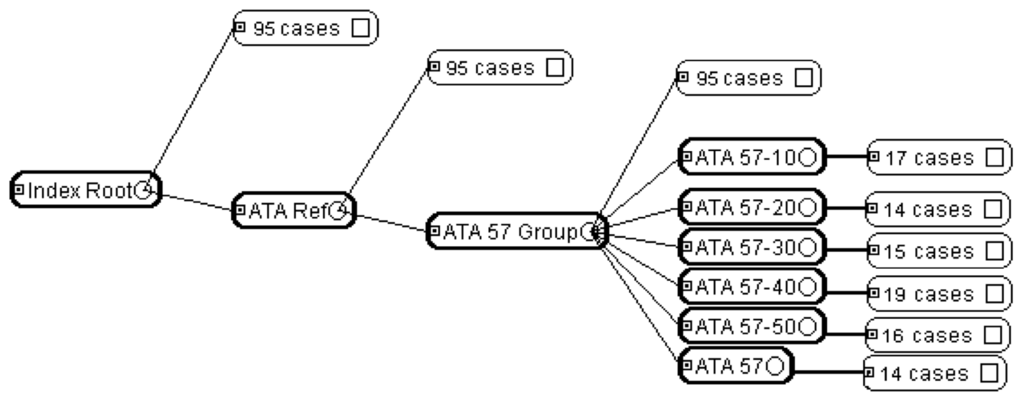

Figure 2. Knowledge-guided prototypes.

The clustering process was then performed within the context of each of the knowledge-guided prototypes i.e., each ATA sub-section. The inductive indexing was performed using the *Key Words* field. The cases are further split into a decision tree structure, which creates a smaller set of cases producing a more efficient and precise retrieval.

The final indexing method used was nearest neighbour matching, and two fields were used to perform this retrieval method: the aircraft type field where an exact match was requested, and the category field where an importance weight of 100% was applied. ReMind automatically places an importance weight of 100% as *Category* is the only field affected. If there were more fields requiring an importance weight, then the 100% value would be split between the fields.

The nearest neighbour matching process now singles out the type of aircraft that the user specifies by selecting either aircraft type. The *Category* field is also used to narrow the set of cases retrieved. Initially, the system will try to retrieve those cases whose Category type the user stipulates e.g. corrosion. However if there are no similar matches in the case-base with *Category* type corrosion, then the system will still retrieve cases which have met all the other input selections but with a different *Category* type. The system will therefore retrieve a meaningful set of cases.

Proposed System Architecture

Currently the system is designed for only two types of aircraft, and a subset of the ATA 100 system. It is intended that the remaining types of aircraft for which the company provide technical support and the complete ATA 100 reference standard will be added to the system symbol hierarchy.

The proposed completed system would incorporate a relational database management system (RDBMS) which would store the information about the aircraft operators and log all their technical queries. This multi-user RDBMS would act as the main data repository for the system. All information such as status of enquiries, temporal data, etc., would reside in the RDBMS, and provide core data for higher level management information generation, relating to work rate and work success rate. It is intended that the system would provide a closed-loop function on technical support enquiries, making adherence to BS (UK) and ISO (EU) Quality Standards easier.

There would be interaction between the RDBMS and the case-base for the retrieval of queries. This retrieval may be considered fuzzy, in the sense that imprecise or mis-spelt queries are made and successful retrievals result, through the CBR mechanisms mentioned earlier. The headquarters engineers would be able to retrieve cases from the case-base. A case-base administrator would have all the privileges for editing new and already stored cases within the case-base. The system architecture is shown in figure 3.

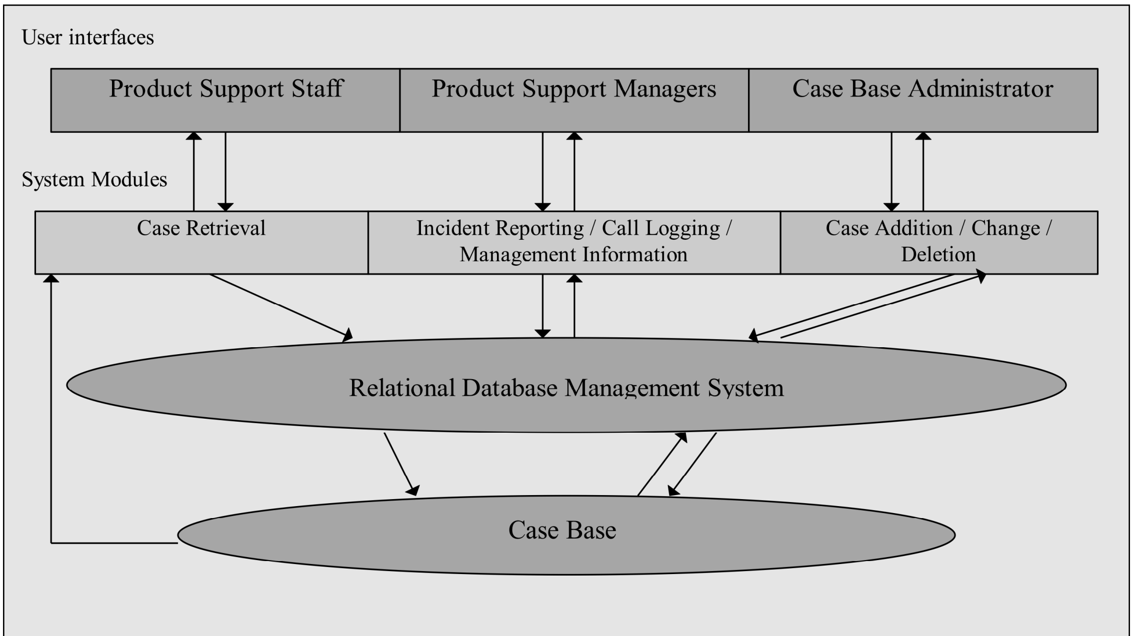

Figure 3. Technical Support System Architecture

It is intended to include on-line technical manuals in the system to help the headquarters engineers when solving technical queries. This would assist in reducing the solving cycle time for technical queries. A prototype system which contains a portion of the technical manuals for one type of aircraft has been implemented. It contains a reference to the ATA 100 filing system, which results in a context-sensitive link to any queries that are made to the system, and thus provides concise reference material on the query to the engineer.

Conclusion

Due to the poor user interface of ReMind, a Graphical User Interface (GUI) front-end for the system was developed using Microsoft Visual Basic. The GUI, through the symbol hierarchy of ReMind enabled multiple-choice selection for the *Aircraft Type*, *ATA Reference* and *Category*. This ensured that the retrieval of cases by the headquarters engineer was relatively quick in terms of entering a new case and easy to use. The RDBMS component of the system is developed using MS Access.

The current paper system in use at the company will not be superseded but will initially be run in tandem with the CBR system. The headquarters engineers will now know through the CBR system if a new query is a recurring problem and has been solved before or is a completely new unsolved technical query, in addition to the system retrieving nearest solutions.

Further research is required on the key-words symbol hierarchy to aid the retrieval of stored cases. This is especially true when the system is scaled up to include the full aircraft range and ATA 100 standard.

Initial evaluation by the users of the prototype system has been favourable. Although restricted to a subset of the ATA 100 standard, the system has already proved useful to the company. In addition to developing the full RDBMS help desk systems, the company are interested in applying the technology to other support functions in the organisation, including IT support.

CBR has been used in help desk applications before [3]. However, the system outlined in this paper is one of the first to integrate CBR with RDBMS to provide a flexible 'help desk' environment. The CBR component provides flexible query processing capabilities, with the ability to handle noisy data, incomplete data, and incorrect data, to a degree. The RDBMS component provides all the traditional help desk functions, such as call logging, reporting, query routing, etc. In addition, the context sensitive links to the technical manuals provide ancillary information that is valuable to the engineers.

References

1. Barletta R. An Introduction to Case-Based Reasoning, **AI Expert** 8, 1991.
2. Harmon, P. Case-Based Reasoning 2, **Intelligent Software Strategies,** 7(11), 1991.
3. Kriegsman, M., & Barletta, R. Building a Case-Based Help Application, **IEEE Expert**, 8(6), 1993.
4. Harmon, P. Case-Based Reasoning 3, **Intelligent Software Strategies,** 8(1), 1992.
5. Forsyth, R. **Expert Systems;** *Principles and Case Studies*, ed. Forsyth R, Chapman and Hall Ltd, (1989).
6. Michalski, R., Carbonnel, J., & Mitchell, T. **Machine Learning: An Artificial Intelligence Approach,** Tioga Publishing Corp, Palo alto, CA, 1983.