# Combining Information Retrieval and Case-Based Reasoning for „Middle Ground“ Text Retrieval Problems

Mike Brown, Christiane Förtsch, Dieter Wißmann

Siemens AG, Zentralabteilung Technik, Software und Engineering,
Paul-Gossen-Straße 100, 91052 Erlangen
Michael.Brown/Christiane.Foertsch/Dieter.Wissmann@erls.siemens.de

## Abstract

This paper considers the type of problem for which the potential for amalgamating Information Retrieval (IR) and Case-Based Reasoning (CBR) technologies is the highest. IR is characterised as a bottom-up approach to retrieval of text within unconstrained domains. CBR is characterised as a top-down approach to retrieval of formalised information within domain-specific applications. It is argued that applications that require relatively detailed responses to specific queries of a large-scale, but domain specific text-based archive represent the middle ground between these two disjoint technologies - this is illustrated with a worked example.

## Introduction

Case-Based Reasoning (CBR) and Information Retrieval (IR) are two historically disjoint technologies. This paper contends that the time has now come for CBR and IR to be conjoined in order to collaboratively solve many of the text retrieval problems faced by modern companies. The two technologies fundamentally differ in terms of the type of information retrieval queries they allow to be answered. An attempt will be made to define what types of query remain problematic for both techniques, yet may benefit from a combined approach.

The paper start by characterising CBR and IR. The types of query that can be answered using these technologies are then classified. This is followed by a worked example of the type of queries that should ideally be supported within a corporate text retrieval system in order to support tehnical experts and a discussion of what information needs to be extracted from raw text in order to support such query answering.

### Case-Based Reasoning

The traditional CBR approach to information retrieval is one of top-down design. The core of a CBR systems is the indexing vocabulary used (Birnbaum 1989). Normally, a set of fairly abstract and purpose-specific features will be identified during design and included within a standard case description. Hence, each index feature allows the set of cases in memory to be discriminated at run-time with respect to one of a number of predetermined problem-solving perspectives.

The weakness of this traditional view of CBR is that it is implicitly assumed that the case-base will be specifically generated for the CBR application - this allows the freedom to tailor the case description to precisely suit the envisaged purposes of the application. In other words, the range of queries that the system must answer are determined *a priori* and the system can be optimised to answering these queries. This assumption is, however, often invalid for real-life applications, particularly where CBR might be used to exploit legacy data repositories (Brown, Watson and Filer, 1995). When one is constrained to use already existing data to construct a working case base, the luxury of top-down design is no longer afforded. Instead, the following questions must be addressed:

- What information is available in the routinely stored data, which can be used to (partially) characterise problem solving cases?
- What transformations can be automatically carried out on the raw data to generate indexes with a higher predictive power?
- How can the CBR system automatically adjust its own set of extracted indexes to improve the accuracy of case retrieval over time?

The above arguments are particularly pertinant for CBR applications that are aimed at tackling text-retrieval. For such applications, the existing, raw data may simple be unstructured text. While CBR applications that deal with texts from highly constrained domains may be able to supplement the raw text with manually provided indexes, or even automated indexing based on domain-specific rules, e.g. (Weber-Lee et al 1997), for more open domains, the CBR system is restricted to comparing cases in terms of information that can be directly extracted from the text.

### Information Retrieval

In contrast to CBR, the traditional approach for IR is bottom-up. There is a deliberate rejection of more theoretic approaches to Natural Language Processing (NLP) in favour of algorithmic appoaches that build on the text itself. As a consequence of this philosophy, IR has been dominated by statistical methods and the primary basis for text retrieval has been through combinations of weighted keywords.

The preoccupation with keyword-based retrieval ultimately limits the applicability of traditional IR. Even with the various possible extensions (e.g. boolean queries, morphological word stemming, thesauri, relevance feedback, text clustering, etc.), a non-optimal upper bound will always exist on the accuracy of retrieval that can be achieved by IR (Belkin, In other words, IR aims to support completely general querying while tolerating relatively low quality of retrieval results.

In addition, IR is poorly suited to fine-grained querying of text documents. While keyword lists may be an effective basis for filtering whole documents (e.g. deciding if a particular document suits a particular users profile), or finding relatively large text chunks within a document, they are usually unreliable for finding specific pieces of information within a document. Statistics do not capture specific patterns! To move towards high degrees of accuracy and granularity of retrieval, some semantic processing must be introduced. The open issue is to determine what sorts of semantics can be extracted from text by largely automatic techniques.

## "Medium-Scale" Text Retrieval Problems

The introductory discussion is summarised in Figure 1. CBR tends towards the application-specific and supports selective retrieval based on relatively fine-grained information. Conversely, IR is more widely applicable, but provides much cruder retrieval. Hence two modes of collaboration are possible: - using CBR to increase the granularity of IR for application-specific purposes, or using IR to allow CBR systems to be developed in open-domain environments, such as the WWW.

In reality, the difficulty of a text retrieval problem lies on a continual spectrum; increasing as the range of possible texts increases, and/or as the granularity of information to be retrieved becomes finer. However, for the purposes of the following discussion, three levels of retrieval complexity (simple, medium and difficult) are identified in the above diagram and will be described below. It will be argued that the most interesting area for possible collaboration between IR and CBR in the immediate future lies in medium-scale retrieval problems.

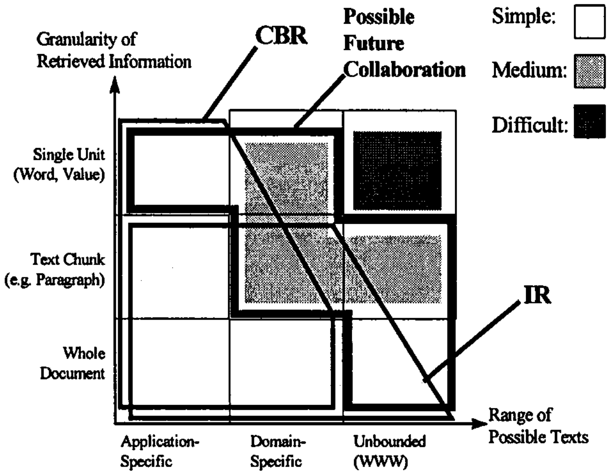

**Figure 1 - Classifying IR and CBR with respect to types of text retrieval**

Narasimhalu and Willett 1997, Jones and Willet 1997).

- **"Simple" text retrieval queries**: Applications at this level are application-specific (allowing for the derivation of specific retrieval indexes for the text) or/and involve the retrieval of large text bodies, where only a keyword list approximation of the document content is a sufficent basis for retrieval. The term "simple" is not meant to belittle the many first-rate applications of this type that are currently in existence (e.g. email filters, FAQ retrieval applications (Lenz and Burkhard 1997), legal CBR systems (Weber-Lee et al 1997) ), rather, that either the range of required queries is fairly constrained and/or the granularity of queries is quite coarse (can be implemented by keyword lists). This level involves answering queries of the form **„Give me all documents that are about lightbulbs“**
- **"Medium-Scale" text retrieval queries**: Applications at this level have a more open range of required queries than at the „Simple“ level, although retrieval may still be confined to documents from a single domain. In addition, querying on the content of documents (rather than more coarse-grained classification), is required; i.e. identifying the location within a single document where a specific piece of information exists. This level involves answering queries of the form **„Show me all paragraphs discussing lightbulbs with power <=30 W“**
- **"Really Difficult" text retrieval queries**: Applications at this level only differ from those for „medium-scale“ retrieval in the extent of the range of queries that can be posed - here, it is assumed that no limit on the content of documents exists, therefore, a domain-specific tailored vocabulary to aid retrieval cannot be developed. This level represents the "holy-grail" of text retrieval, e.g. of how search-engines for the WWW *should* work in an ideal world. This ideal is still some way off and may require something approaching natural language understanding to be fulfilled. Currently, only traditional IR techniques are mature enough to effectively deal with the enormity of the WWW. This level involves answering queries of the form **„Find me a joke about nuns changing a lightbulb. How many nuns were involved?“**[1]

[1] Note, given a query such as „joke & nuns & lightbulbs & changing“, current search engines do retrieve a lot of humours material, about nuns or about lightbulbs and probably even a correct response to the query - the author spent an enjoyable 15 minutes filtering the retrieval results before giving up. Finding an exact match to the query without returning the morass of nearly-relevant material is, however, the real problem.

## A Worked Example of Medium-Scale Retrieval Queries

While there are many practical applications within a company for systems that can classify documents, or perform retrieval based on matching keyword lists, there are also many problems that involve experts trying to locate *specific* information from within large technical documents. For these problems, classical IR is too weak, yet the application-specific tailoring of most CBR systems is also infeasible. These ideas will be illustrated with the example text of Figure 2, taken from (KWU 1993). As might be expected, this document is just one of thousands of documents containing technical information about power plants. The types of query that might be useful for an expert to pose are summarised in the following table:

**Paragraph 3**: In September 1990, just three years after the award of the contract, the first steam turbine-generator was put into combined-cycle operation with the first of the three pairs of gas turbine-generators which had already operated in the simple-cycle mode over two years.

**Paragraph 4**: After a record construction period of only 10 months following the award of the contract, the first gas turbine-generator went on line

**Paragraph 7**: The three GUD blocks achieve a net efficiency well in excess of the contractually guaranteed 51.37% level. In fact, official acceptance test measures on the first GUD block demonstrated 52.5% at rated load and 53.2% at peak load which means that the plant utilizes natural gas for power generation to an unrivaled high degree of efficiency

**Paragraph 17**: All of the six gas turbines are accommodated in a building 140 m long and 18 m wide, which is equiped with a 50/10/7.5 t crane. The distance between the machine axes is 23 m and, therefore, provides spacious lay-down areas for major maintenance work.

**Paragraph 18**: The Model V94 gas turbines installed in the new Ambarli power plant are heavy-duty machines with the proven Siemens design features:...

**Paragraph 34**: <TABLE 2> Operating data of a heat-recovery steam generator with the associated natural-gas-fired gas turbine at rated base load.

**Paragraph 50**: Prior to the handing over of the station, the utility's staff received detailed instruction in the form of special courses and perparatory training in the plant. By the end of 1989 the six gas turbine-generators had already generated more than 4 billion kWh, which required a fuel input of 1381 million cubic meters of natural gas with a calorific value of 33,620 Kj/m3

**Paragraph 54**: The new power plant at Ambarli supplies two separate electrical systems; two GUD blocks feed into a 154 kV grid and the third into the 380 kV national grid.

**Figure 2 - Examples from a Technical Text**

| QUERY | IDEAL RETRIEVAL | SOURCE(s) |
|---|---|---|
| Ql) Where is the power plant? | Ambarli | Paragraph 18, 54 |
| Q2) When did the power plant go into operation? | In September 1990, just three years after the award of the contract, the first steam turbine-generator was put into combined-cycle operation<br>By the end of 1989 the six gas turbine-generators had already generated more than 4 billion kWh | Paragraph 3, 50 |
| Q3) How long was the construction period for the plant? | 10 months | Paragraph 4 |
| Q4) How many turbines does the plant contain? | six | Paragraph 17, 50 |
| Q5) What is the temperature at the generator inlet? | <TABLE 2> *** a graphics image plotting various system parameters *** | Paragraph 34 |
| Q7) What is the efficency of the plant? | the first GUD block demonstrated 52.5% at rated load and 53.2% at peak load<br>4 billion kWh, which required a fuel input of 1381 million cubic meters of natural gas with a calorific value of 33,620 Kj/m3 | Paragraph 7, Paragraph 50 |
| Q8) What are the physical dimensions of the turbine building? | 140 m long and 18 m wide | Paragraph 17 |
| Q9) What is the voltage level of the supplied electricity? | 154 kV grid and the third into the 380 kV national grid | Paragraph 54 |
| Q10) What design aspects of the plant are important for environmental impact? | ??? | ??? |

**Q10)** is included as an example of the problem of top-down approach to text retrieval. The question posed is valid and interesting but, unlike the other questions, it does not correspond in any simple way to the content of the document. What is envisaged here is that an expert with the overall goal of answering such a query can do so by formulating a series of more basic queries more closely grounded to the content of the text, so as to retrieve the various types of information that collectively answer the more abstract, goal-directed query.

The location of relevant text to many of the above queries may be achived by simply extracting keywords from the queries. The following table summarises the possible keywords associated with each query. The 2nd and 3rd columns give an estimate of the *Precision* (High, Medium, Low) and *Recall* (Yes, No) of the resultant keyword query, assuming word variations (plurals, etc.) are also taken into account:

| Query | Keywords | Recall | Precision | Pargraphs Retreived |
|---|---|---|---|---|
| Q1) | „Power Plant“ | Yes | Medium | Full - 18, 54 Part - 7, 50 |
| Q2) | „Power Plant“ + „Operation“ | Yes | Low | Full - Part - 3, 7, 18, 50, 54 |
| Q3) | „Construction Period“ + „Plant“ | Yes | Low | Full - Part - 4, 7, 18, 50, 54 |
| Q4) | „Turbine“ + „Plant“ | Yes | Low | Full - 18, 34, 50 Part - 3, 4, 7, 17, 54 |
| Q5) | „Temperature“ + „Generator Inlet“ | No | --- | Full - Part - |
| Q6) | „Model“ + „Turbine“ | Yes | Medium | Full - 18 Part - 3, 4, 17, 34, 50 |
| Q7) | „Efficiency“ + „Plant“ | Yes | Medium | Full - 7, Part - 18, 50, 54 |
| Q8) | „Physical Dimensions“ + „Turbine“ + „Building“ | Yes | Low | Full - Part - 3, 4, 17, 18, 34, 50 |
| Q9) | „Voltage“ + „Electricity“ | Yes | High | Full - Part - 54 |

As is shown, the main problem expected with keyword queries is the lack of *precision*, rather than the lack of *recall.* The only keyword-based query that is expected to fail from the above is **Q5)**, where the required information is hidden within a table of technical data. The interpretation of information in technical documents that is in non-textual form (diagrams, tables, etc.) is a major concern, but beyond the scope of this paper.

The major observation from the above queries is that they are usually about a specific type of information: **Q1)** about a location, **Q2)** about a time point, **Q3)** about a time duration, etc. These types of information are signified by function words in the query (where, when, how long, etc.) and the keywords to match against in the text are not explicitly given. Indeed, for a given type of information (such as time), the set of associated keywords may be extremely large and therefore practically impossible for a user to specify. Therefore, the ability to introduce such concepts as primitives for constructing queries is required.

## Features for Text Retrieval

From the above worked example, it should be clear that the level of sophistication of text-retrieval that can be achieved by a system and the types of information used for retrieval are inherrently related. In this section, a short list of possible information types are presented. These are approximately classified with respect to two dimensions: discriminating-power and cost-of-extraction (from raw text).

At the easy-to-extract and poorly-discriminating end of the spectrum belong those information types primarily used as the basis for IR. These include:

- Weighted Keyword
- Structural Constraints (i.e. information concerning the various types of component within a document
- Syntactic Information: such as the part-of-speech tagging that is produced by parsers

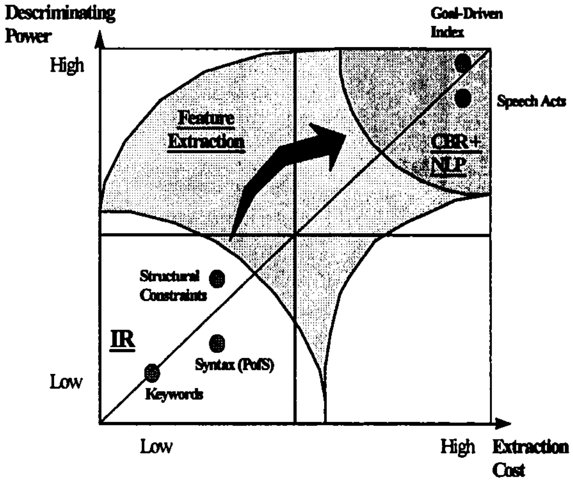

**Figure 3: Bridging between low-cost-low-performance and high-cost-high-performance retrieval**

By contrast, the types of information typical of traditional CBR systems lie at the difficult-to-extract but highly-discriminating end of the spectrum. In this sense, CBR is similar to theoretic NLP concerning the structuring of dialogues and texts. For example, the types of abstract concepts proposed in theories such as Speech Acts (Austin 1962) and Rhetorical Structure Theory (RST) (Mann and Thompson 1987) can be thought of as classical indexes, in the CBR sense, because they pertain to the intentions and goals of the communicating individuals, rather than the details of the text and speech used to portray these purposes. Not suprisingly, the main criticism of such theories has been the difficulty in grounding them in sufficiently concrete rules and terms to allow them to be implemented in automated text-analysis systems.

In the few systems that have made some progress in this respect, the mapping from text to the required high-level concepts is not direct; typically intermediate types of information must be first recognised as occuring within the text and then the higher level concepts recognised from the intermediate concepts, e.g. (Aretoulaki 1996). These intermediate types of information will be refered to here as "annotations". This terminology stems from a number of systems involved in the series of Message Understanding Conferences (MUCs), e.g. (MUC-5 1993), where, as part of the preliminary processing of text documents, all occurences of specific types of information (i.e. annotations) within a document are identified and in some way labelled. The annotated document provides the basis for subsequent, more detailed semantic analysis of the text. Examples of annotations relevant for the previous worked example would include „location“ (Q1), „time-point“ (Q2) „time range“ (Q3), etc.

From the CBR perspective, an annotation lies somewhere between the ideal, goal-directed indexes of the top-down perspective and the surface-level text features of a bottom-up perspective. It is the compromise between what we can reasonable expected to automatically extract from the text, and the ideal basis for retrievals. Therefore, as shown in Figure 3, this paper concludes that the topic of „feature extraction from raw text“ is where IR and CBR should come together.

## Conclusion

This paper has argued that the types of problem where CBR and IR can be usefully conjoined involve the retrieval of detailed information from the large, technical, domain-specific documents that are still the main form of information storage and exchange in most industrial settings. Furthermore, it is argued that the main area for immediate work is in providing techniques for extracting semantic features (annotation) from raw texts. The following questions can therefore be identified as relevant issues for discussion and the focus of future work:

- What different types of annotation exist
- What is the (hierarchical) dependency between different types of annotation
- What technologies are best suited for extracted annotations (neural nets, specialist parsers, CBR!)
- What measureable, retrieval performance improvements can be achieved through use of annotations.

The importance of tackling „medium-scale“ text retrieval problems should not be under-estimated. As an illustration, a realistic, medium-term goal for the amalgamation of IR and CBR could be the realisation of a „corporate memory“ (PAKM 1996) - a term used for the idea of routinely capturing and reusing the experiences gained through carrying out projects within a company. The main hinderance to the realisation of a "corporate memory" is that, currently, for most companies, the information that requires to be reused is contained in natural-language documents, albeit in electronic form. For the simple reason that people prefer to communicate via normal (i.e. informal or semi-formal) text, this situation is not likely to change in the near future. Hence, if a corporation really wants to make the use of its recorded information, text retrieval technologies will need to be harnessed. To be useful, these technologies must do more than just classify the various documents archived within the corporation, they must support the access of specific information embedded in those documents.

## References

Aretoulaki, M. 1996. COSY-MATS : A Hybrid Connectionist-Symbolic Approach to the Pragmatic Analysis of Texts for their Automatic Summarisation. Ph.D. diss., Dept. of Language Engineering UMIST.

Austin, J. L. 1962. How to do Things With Words. Oxford University Press.

Belkin, N. J., Narasimhalu, A. D., Willett, P. eds. 1997. SIGIR-97 - Proceedings of the 20th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, Philadelphia, USA.

Birnbaum, L. chair. 1989. Panel Discussion on „Indexing Vocabulary“. Proceedings of the Case-Based Reasoning Workshop, Florida.

Brown, M., Watson, I., Filer, N. 1995. Separating the Cases from the data: Towards More Flexible Case-Based Reasoning. Proceedings of ICCBR-95; First Int. Conf. on CBR, Portugal.

Cardie, C., 1993. A Case-Based Approach to Knowledge Acquisition for Domain Specific Sentence Analysis. Proceedings of the 11th Nat. Conf. on AI. AAAI Press.

Jones, K. S., Willet, P. eds. 1997. Readings in Information Retrieval. Morgan Kaufmann.

Power Generation Group (KWU), Siemens AG. 1993. The Ambarli Combined-Cycle (GUD) Station - A Power Plant That Sets Standards.

Lenz, M., Burkhard, H-D. 1997. CBR for Document Retrieval; The FALLQ Project. ICCBR-97, proceedings of the 2nd Int. Conf. on CBR. Providence, USA.

Mann, W. C., Thompson, S. A. 1987. Rhetorical Structure Theory: A Theory of Text Organisation, Tech. Rep. ISI/RS-87-190, USC Information Sciences Inst.

MUC-5. 1993. Proceedings of the 5th Message Understanding Conference. Morgan Kaufmann.

PAKM. 1996. Proc. of the 1st Int. Conf. on Practical Asepcts of Knowledge Management, Basel Switzerland.

Weber-Lee, R., Barcia, R. M., da Costa, M. C., Filho, I W. R., Hoeschl, H. C., D'Afostini Bueno, T. C., Martins, A., Pacheco, R. C. 1997. A Large Case-Based Reasoner for Legal Cases. ICCBR-97, proceedings of the 2nd Int. Conf. on CBR. Providence, USA.