CONCEPTUAL INFORMATION RETRIEVAL IN LITIGATION SUPPORT SYSTEMS

Vijay Mital, Agathoclis Stylianou and Les Johnson
Knowledge Based Systems Group
Computer Science Department
Brunel University
Uxbridge, Middlesex UB8 3PU, U.K.
+44 0895 74000; se89vvm@cs.brunel.ac.uk

Abstract

This paper is concerned with information retrieval in the context of supporting complex litigation by managing large numbers of documents. It is shown that the application is sufficiently different from searching for case/statute text or reasoning with the law, so as to render the techniques developed for the latter inappropriate. A new approach to information representation and system design is identified and developed. The paper presents an architecture that takes into account the peculiar characteristics of the application and enables the utilisation of existing skills of professionals, thereby facilitating rapid and consistent encoding. An extended object-oriented paradigm underlies the architecture. Using this paradigm, it has been possible to combine techniques developed for large databases with the purposive or functional similarity approach to search and retrieval taken in case-based design systems.

1 Introduction

The problems associated with full or free text retrieval are well known. Even where thesauri (Bing, 1989) and lexicons (Weaver et al, 1989) are employed, users find it difficult to formulate queries to pinpoint those out of a large collection of documents that might contain the desired information. It is possible to improve the user interface, e.g., by means of a front-end containing rules associating concepts of interest with particular word/phrase patterns (Lewis et al, 1989; Tong et al, 1989). However, the fact remains that, in a rich domain, words and phrases are a poor approximation to meaning without due consideration of the conceptual relationships between them (Rau, 1987).

Researchers concerned with artificial intelligence applications in law have confronted the above issues directly because information retrieval is a task integral to most such applications. They have long recognised the need to organise legal information in a manner that enables retrieval based on the meaning and legal significance of text (Hafner, 1981; Bertaina et al, 1982). The purpose of retrieval may vary. A legal research system may simply display the located information (Dick, 1987); a case-based reasoning system may itself make the inferences (Rissland & Ashley, 1989); or the position may be somewhere in between (Gelbart & Smith, 1990). The common element is the aim to represent the relationships and dependencies between a legal concept and its subconcepts, and between a concept and its categorisation in the universe of discourse (Bareiss, 1989), in as explicit a manner as the technology permits[1]. As either complex generalised formalisms such as conceptual dependencies (Schank, 1975) or sophisticated special-purpose representations (Cross & deBessonet, 1985) are employed to make these relationships and dependencies explicit, theoretically, any (and every) element of legal significance can be indexed upon and differentiated from any other element.

However, the process of indexing on or reasoning with complex knowledge structures is computationally expensive, to the point of being intractable when, say, more than a few hundred realistic documents or cases are to be dealt with (Martin, 1989). It is not being suggested that the problems will not be overcome in due course if research continues apace along the present lines and significant advances continue to be made. But, there is another factor which makes some aspects of the above approach inappropriate for the purposes with which we are concerned in this paper. There are some profound dissimilarities between the computerised support of complex litigation by management and retrieval of documents on one hand and legal research or legal reasoning with a view to giving advice on the other.

It is true that the concepts that a lawyer is interested in when using a litigation support system (LSS) are essentially the same as would be recorded in the judgement of the court. The decoupling of the text actually used and the concepts it conveys has the same effect. For instance, a search of a database of reported cases may reveal that the word 'intention' does not appear in an exposition of the concept of intention (Dick, 1987). Whilst, a user of a LSS may find that letters and memoranda discussing an 'accident' contain only oblique references such as 'what happened last week' and '[w]e all know why we're here' (Blair & Maron, 1985). Or, that the reference in a letter to a 'defect' is in graphic terms - 'our product explodes' - but the word 'defect' is absent (Wallwork, 1989). It is at the practical level - and practical considerations exert great influence on system design (Mital & Johnson, 1991a) - that the litigation support application most significantly differs. Some of the special application characteristics which have influenced the system architecture presented in this paper are as follows:

(a) A litigation support database often has to endure large, spasmodic, additions[2].
(b) Usually, every document to be inserted in the database will have been read and screened for relevance by one or more members of the litigation team - either junior lawyers, or paralegals. These persons have the ability to abstract information from documents for the purpose of indexing/cataloguing documents in a relatively consistent manner, either manually (Halverson, 1979) or using one of the widely available LSS that are loosely based on the manual indexing/cataloguing methodology but do not purport to effect conceptual retrieval (Wilkins, 1989; Christian, 1990).
(c) Any particular LSS is likely to be used only by a small number of persons whose profile is predictable in advance, as is the role of the system.

Consequently, the representation formalism should allow rapid encoding by a number of persons working without constant reference to each other. The aspect of information that is to be abstracted and represented should be such as the personnel already in place are capable of providing. Lastly, it makes sense to sacrifice generality of representation - even assuming that it is achievable - in favour of giving the users the means to adapt the representation schema to their own peculiar needs. The consideration of computational tractability, so that large numbers of documents may be handled, has already been mentioned.

In subsequent sections, we present the architecture of a system currently under development by the authors that is based on an object-oriented schema which we believe to be particularly suitable for the information characteristics at hand. This system is not intended to *replace* full text retrieval systems, but to *augment* them. We combine techniques developed for object-oriented databases that can handle vast quantities of simply represented information, with the rich notion of retrieval on the basis of purposive or functional similarity usually employed in case-based design (CBD) systems. We start by briefly mentioning the characteristics of the object-oriented paradigm as extended for the application at hand.

2 Extended Object-Oriented Paradigm

The object-oriented (OO) paradigm is based on the idea of abstracting the characteristics of a world truth in a manner that has a direct and natural correspondence between the world and its model, and encapsulating that abstraction. Objects contain a data structure and, in addition, may contain the procedures (methods) associated with the data. There is as yet no established form for the OO paradigm, with application specific adaptations prevailing. Broadly speaking, it is possible to distinguish those applications where the task involves systems analysis or program construction from those where the emphasis is on the richness of representation of data (Kim, 1990) or knowledge (Patel-Schneider, 1990). In the latter manifestations, the OO paradigm formalises and extends the representational ideas underlying semantic networks and frames.

Objects can be classified in a taxonomy formed by generalisation-specialisation or parent-child links. If multiple parents are permitted, the taxonomy may be termed a tangled hierarchy. The primary functions of the parent-child links are to enable properties or information to be inherited and to allow limited inferences to be made. These links are not meant to be determinative of the semantics of the relationship between two concepts that happen to be placed as a parent and a child or siblings in a taxonomy for some limited purpose. Other kinds of links are, therefore, added to the core formulation to represent the required information explicitly and elegantly.

One such link is the association link which can be used to relate two object classes. If the link is treated as an object in its own right, then an instance of the link connects an instance of each of the related classes; the semantics of different kinds of association links can be carried as data and methods within the link, rather than in each of the linked classes.

Another kind of link, which has the same surface structure as an association link, is also sometimes necessary. An example is given in Figure 1, where it is sought to represent the information that an instance of class X can never co-exist with any instance of class Y. This, obviously, is not easy to represent using parent-child links. It is more naturally represented using a link labelled 'cannot exist together' between the two classes. The 'cannot exist together' link in the example in Figure 1 should not be confused with an association link because the latter is constrained into showing the functional dependency and connectivity between the associated objects. Therefore, the link shown in Figure 1 is termed an *extended association link.*

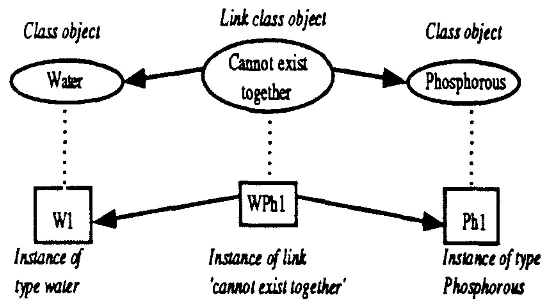

Figure 1 : Conceptual Structure of Extended Association Link

3 Representing the Purpose, Not Entire Text Content

The constructs which we provide do not readily allow *all* the information contained in documents to be represented. In fact, we have no desire to have all the available information encoded, for that would lead to the same problems of retrieval and reasoning being overwhelmed with a surfeit of information that affect full text retrieval. If the entire contents are not to be represented, and only salient features are to be captured, what is salient must be clarified.

We start with the assumption that the user is primarily interested in the retrieval of documents for a particular purpose, and only those features that are relevant to that purpose need be explicit. The primary purpose of the user is, with the aid of documents, to prove or disprove certain legal or factual issues that are in contention - the reference is not just to the issues that a court might frame, but also to those that might be used for indexing/cataloguing documents either manually or using one of the commercially available LSS. A simple example can be given. In the case of an alleged negligent misrepresentation by a financial adviser leading to a loss being suffered by the client, the following may be some of the broad issues of contention, with further decomposition as shown in Figure 2:

(a) Whether the client possessed information from independent sources, such as to enable him to know that the defendant's advice was incorrect.

(b) Whether the client acted as per the defendant's advice.

(c) Whether the loss was caused by reasons other that the actions taken pursuant to the defendant's advice.

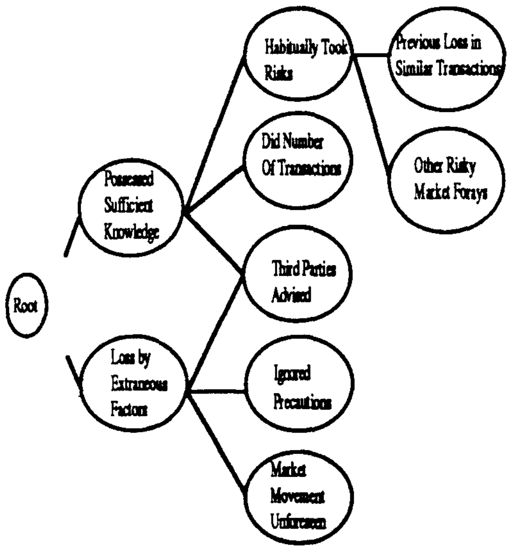

Figure 2 : Issues Taxonomy

The relevance of a retrieved document to one or more issues (or sub-issues) may be the ultimate test of how well it serves the purpose of the user. However, labelling each document merely with those issues to which it is thought to be relevant is too imprecise and coarse. This is because the user may wish to retrieve only those documents that relate to an issue *in a particular way.* For instance, documents that are relevant to the issue 'third parties advised', but are relevant by virtue of being admissions recorded by the client himself, rather than as statements in letters sent to the client by third party advisers, or records by other persons of conversations between the client and the advisers. This is a rich notion of the purpose of retrieval and, as shall be discussed below, is akin to the notion of functionality or similarity in some CBD systems, including where design involves legal reasoning (Mital & Johnson, 1991b). In those systems, functionality of a case (or its similarity to the problem at hand) may be measured by the case's usefulness in solving a certain problem element in the context of the particular situation. This notion of functionality, when applied to litigation support, means that the relevance of a document to an issue is no longer merely a pre-defined and static parameter, but is judged dynamically depending upon the context specified by means of the request for retrieval. We seek to represent only that aspect of information which is required to serve such broad purposes of the user.

4 Conceptual Representation of Documents

The conceptual representation of a document is as an aggregate object containing instances of other objects (i.e., primitive domain concepts, issues, explanatory links, reference links and relevance functions), see Figure 3.

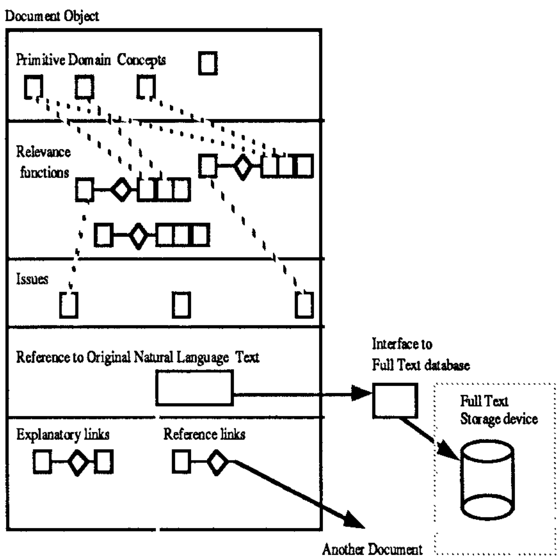

Figure 3 : Conceptual Representation of Document

The original natural language text is not held within the system being described here; it is anticipated that a full text database that relies on fast retrieval devices such as optical disks, perhaps with an industry standard query interface (Cornwell, 1990), will be employed. The facility for interfacing to SQL fronted relational databases has been provided (Stylianou, 1990).

The issues taxonomy has already been mentioned. Primitive domain concepts include (a) the basic facts which are used to judge the relevance of a document to an issue, and (b) the kinds of documents that occur in the domain of interest. They are organised in two separate tangled hierarchies, as illustrated in Figures 4(a) and 4(b).

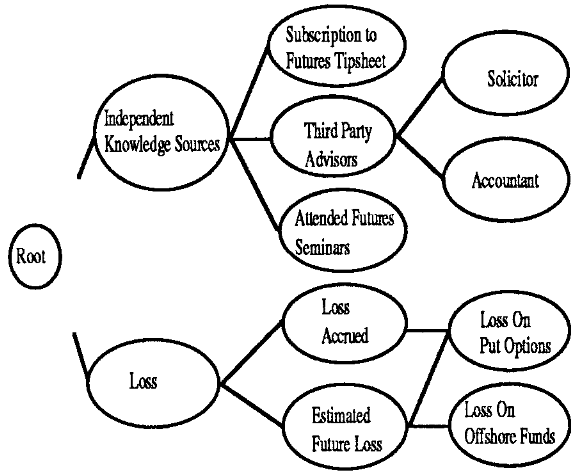

Figure 4(a) : Primitive Domain Concepts (Situation Facts)

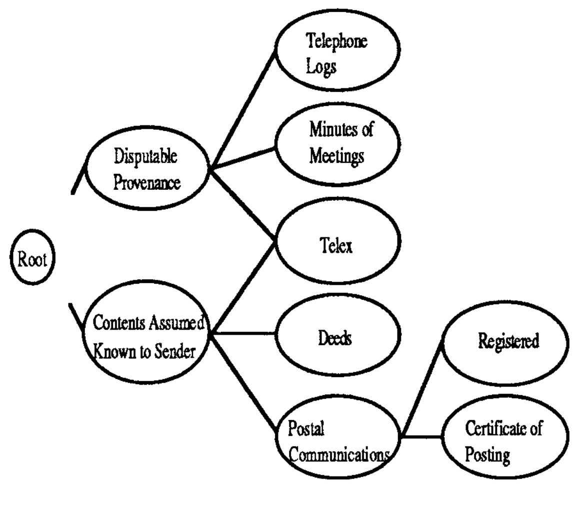

Figure 4(b) : Primitive Domain Concepts (Document-Kinds)

4.1 Explanatory Links

Explanatory links relate issues and primitive domain concepts in any one of the combinations shown in Figure 5, where PDC stands for primitive domain concept[3]. The links represent the extent of the validity of the particular interpretation regarding relevance that the person carrying out the encoding has placed on the document. For example, he may state that a document contains the primitive domain concept "attended seminars on futures". However, there may be an alternative interpretation that the client did not attend the seminar as a passive listener and, instead, went to see the lecturer for personal advice on a particular problem. If so, the primitive domain concept 'advised by third parties' may be related to 'attended seminars on futures' using the 'alternative interpretation' link. It must be noted that explanations are confined to the document in which they are specified, and are not to be thought of as global relationships.

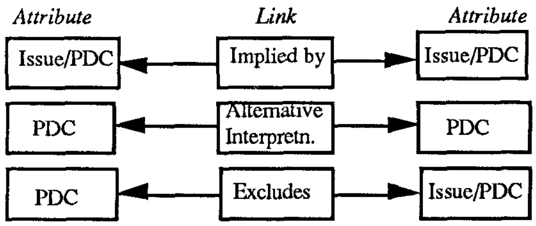

Figure 5 : Types of Explanatory Links

4.2 Reference Links

A reference link relates the document in which the link is specified to one or more other documents, see Figure 6. Each type of reference link has a special semantic significance. For example, a letter may be sent by the plaintiff in reply to an accusatory letter from the defendant, denying liability. If the encoder specifies the "rebuts" link, it signifies that the plaintiff's letter contains information likely to go to disprove some or all those issues in the proof of which the defendant's letter is relevant.

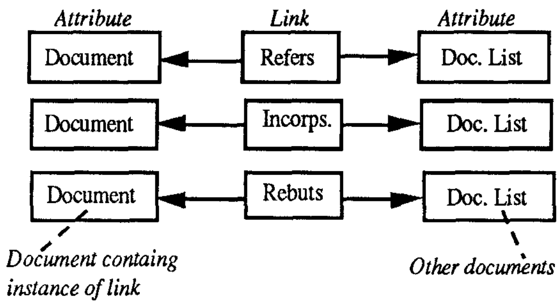

Figure 6 : Types of Reference Links

4.3 Relevance Functions

Relevance functions relate issues with primitive domain concepts. The attributes of a relevance function object, an instance of which is shown in Figure 7, are (a) one, and only one, issue; and (b) one or more primitive domain concepts.

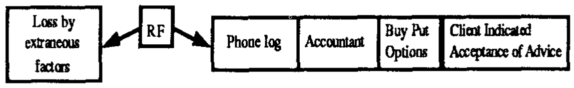

Figure 7 : An Instance of the Relevance Function Link

A particular instance of a relevance function is, strictly speaking, valid only within the context of the document in which it is specified[4]. Still, it is inevitable that *instances* of relevance functions (RF's) containing identical attribute values will occur in a number of documents. Also, that different documents will contain RF's that have attribute values such as to make the documents 'similar' in some sense or for some purpose of the user. As documents are sought to be organised and indexed by the RF's they contain, the RF's themselves must be organised so that searching is minimised. We have chosen to use discrimination by hierarchical subsumption (Galloway, 1987) as the basis of organising RF's. The primary criterion for discrimination is the value of the issue (IU) attribute, the secondary criteria are the values of the primitive domain concept (PDC) attributes in the order of importance pre-specified by the human encoder. Figure 8 illustrates subsumption by discrimination[5].

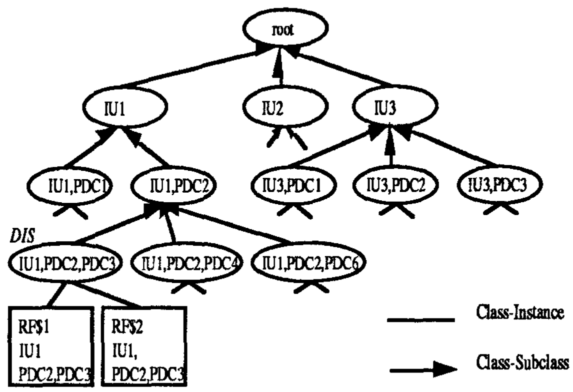

Figure 8 : Subsumption Hierarchy of RFs

Essentially, documents may be thought to be notionally organised *through* the RF subsumption hierarchy. However, as each document may contain more than one RF, the *documents themselves are not in a subsumption hierarchy*. This is in marked contrast to the simpler object or frame-based systems that enforce subsumption between documents (Weaver et al, 1989), making them unsuitable for the complex interrelationships extant in litigation support domains.

5 Querying and Retrieval for Browsing

Generally, it is not sufficient to retrieve a single document that best satisfies some query or is most likely to be relevant to a particular issue in a particular manner. It is more useful to construct a set of documents that are more or less likely to serve the purpose of the user, and then allow the user to browse through this set in a structured manner that has semantic significance. As such, a user query is treated as the means by which the user specifies the *context* in which the relevance of (and similarity between) the documents to be retrieved for browsing is judged.

The user may specify a complex query consisting of a Boolean combination of query elements. Each query element must have the same general structure as relevance functions: ie., consist of one issue and one or more primitive domain concepts. For a document to be retrieved as part of the browsing set, each query element must be matched (or a match excluded where Boolean NOT qualifies the query element) with at least one RF in the document. We will now describe what we mean by a *match* between a query element (referred to below as 'query', for short) and a RF. For a trivial match, i.e. one not relying on explanatory links, the following conditions must be true:

(a) Either the issue specified in the query must be identical to the issue contained in the RF (i.e., both must be instances of the same object in the issues taxonomy); or the two issues must be instances of class objects that share a common parent or grandparent in the issues taxonomy.

(b) Each of the primitive concepts in the query must either be identical to a primitive concept in the RF, or share a parent with that primitive concept.

A non-trivial match can be established by looking at the explanatory links attached to the document to which the RF belongs. For example, consider the situation partly illustrated in Figure 9. There, the two issues to be matched are neither identical nor share a grandparent in the issues hierarchy (Figure 2) because they are in no globally applicable relationship according to the conceptual analysis of the domain. However, a match may be found if the document contains an explanatory link stating that 'loss by extraneous factors' is an *alternative interpretation of* 'advised by third party'. It is necessary to match 'document with disputable provenance' with 'phone log', and 'third party adviser' with 'accountant'. These trivially match, see the taxonomies in Figures 4(a) and (b). However, if a common parent or grandparent could not be found for two primitive domain concepts, but if there was an appropriate explanatory link, say, *implied by*, between the two primitive domain concepts or between one of them and a parent of the other, a match might still have been found.

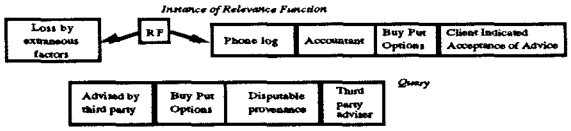

Figure 9 : Matching Query Element with RF

A RF may be said to partially match a query if their issues are matched but one or more primitive domain concepts in the query do not find a match in the RF. Once a set of documents is retrieved, the user can browse through the documents in the order of the degree of matching. Essentially, the system is aiming to provide dynamic clustering of document in accordance with the similarity between them, similarity being judged dynamically in the context of the query, rather than being a fixed parameter. The user can also traverse along the reference links specified in the retrieved documents in order to find other documents which are explicitly referenced or incorporated in, or rebutted by, the retrieved documents.

6 Discussion

6.1 The Relevance Function as a Relatively General Index

Using a function consisting of relationships between certain salient features to index documents is not new. However, most current research is reported to be based on matching only one relation per frame, and "*there is potential ... for considerably improving these methods by allowing matches on more than one relation at a time*" (Lewis et al, 1989). We believe that our work goes some way in this direction by allowing a number of functions to be specified within a document object and then retrieving on the basis of the combined effect of the functions.

We also recognise that it is necessary to ensure that any function is designed so as not to be overly sensitive to minor inconsistencies or variations between the ways in which different encoders view or represent the same concepts. There are several factors in the architecture which contribute to the achieving of this aim.

Firstly, the function is not purely artificial or mathematical; it carries a semantic significance in the domain. The encoders are asked to do little more, at the conceptual level, than exercise their existing faculty of making assessment of the relevance of documents to issues in contention. This they are quite able to do. It is explaining the assessment "through a logical chain of inference" (Ashley, 1989), or decomposing concepts into subconcepts and specifying their relationships and dependencies (as would be required if, say, conceptual dependencies were used to represent the text content of documents), which is difficult. Using existing skills also means that there will be fewer problems with the consistency or integrity of encoding.

Secondly, an exact match between the primitive domain concepts associated in an RF is not insisted upon and the matching can be partial. Moreover, primitive domain concepts that are closely related in the taxonomy are said to match, allowing the encoder some leeway. Where there is doubt about the interpretation, explanatory links can be employed to reflect, to a certain extent, the nature and scope of the doubt.

6.2 Relationship to Functional Similarity in Case-Based Design

It is not necessary here to go into the details of CBD (or case-based planning, which is equally relevant for present purposes); they are elaborated elsewhere (Hammond, 1988; Mital 1990). Essentially, for every problem-solving step, a CBD system searches for a case that deals with a situation that is 'similar' to the problem situation at hand. Determining that two situations are similar is a crucial step in the drawing of an analogy. The process of making analogies between two states of affairs allows us to infer from the fact that there are some similarities between the states that there must be other similarities (Leishman, 1990) - i.e., that the step employed in the retrieved case is applicable to the current problem situation. In this sense, a similarity between two situations is a commonality at some level of abstraction. Of course, establishing (or even defining) similarity may be very complex in the case of "without-domain" or inter-domain analogy. For instance, a student learning about heat transfer can map the knowledge that water falls from a high elevation to a lower one into the heat transfer domain, and from that derive an understanding as to the direction in which heat flows between two bodies at different temperature levels. There, complex issues such as systematicity are involved (Gentner & Toupin, 1986). Fortunately, we are dealing strictly with 'within-domain' analogy, where all concepts to be considered belong to the same domain and it can be taken that identical predicate structures have the same sematic significance throughout. In such a situation:

> "*Object similarity can potentially be reduced to predicate similarity: two objects are similar to the extent they serve as arguments of similar predicates.*" (Holyoak & Thagard, 1989).

However, searching for objects which are similar to the problem situation can still be computationally expensive. It is necessary to ensure that the indices bear a close relationship to the particular notion of similarity employed in a system (Mital, 1990). Also, that the features indexed upon are not such as to exist in the domain in very large variety. In CBD, it has been pointed out that while a huge variety of actual 'output behaviours" of design cases exist, the '*desired* output behaviours' are limited in number (Goel, 1989). In choosing as indices features indicating the purpose (seen from the user's point of view) of documents, rather than the actual combination of concepts occurring in the text, we are acting accordingly.

6.3 Part of a wider effort

The system which has been discussed above is part of a concerted, broader approach to information management for practice support being taken at Brunel University. Additional areas of research include using hypertext for legal document assembly (Southam et al, 1991) and neural networks for automatic text analysis and information retrieval (Gedeon & Mital, 1991).

7 Conclusions

One of the central strands of current artificial intelligence research involves adapting, refining and augmenting existing techniques to suit particular, well-defined domains and applications. This is as a consequence of the recognition that the search for general purpose representation schemata and inferencing mechanisms has left behind significant gaps that need to be filled *de novo* every time a practical development is carried out. We have shown that the litigation support application - one of enormous commercial importance - has peculiar characteristics that necessitate the use of special techniques. These characteristics become apparent only when the specific nature of the application is thoroughly investigated, rather than through an analysis of the nature of legal concepts in general.

References

Ashley, K.D. (1989) "Indexing and analytic models." *Proceedings, DARPA Case-Based Reasoning Workshop*, 197-202. Bareiss, R. (1989) *Exemplar-Based Knowledge Acquisition*, San Diego, Academic Press.

Belew, R.K. (1987) "A connectionist approach to conceptual information retrieval." *Proceedings of First International Conference on Artificial Intelligence and Law*, 116-125.

Bertaina, P.; Borello, E.; Di Leva, A. & Giolito, P. (1982) "The architecture of a legal information system." in C. Ciampi (ed.) *Artificial Intelligence and Legal Information Systems*. Amsterdam, North-Holland.

Berul, L.H.; Stern, M.P. & Keane, J. (1980) "The need for computer-managed litigation files." in B.K. Eres (ed.) *Legal and Legislative Information Processing*, Westport, Greenwood Press.

Bing, J. (1989) "The law of the books and the files: possibilities and problems of legal information retrieval." in G.P.V. Vandenberghe (ed.) *Advanced Topics of Law and Information Technology*, Hillsdale, NJ, Lawrence Erlbaum, 151-182.

Blair, D.C. (1990) *Language And Representation In Information Retrieval*. Amsterdam, Elsevier.

Blair, D.C. & Maron, M.E. (1985) "An evaluation of retrieval effectiveness for a full-text document retrieval system." *Communications of the ACM*, **28**, 3, 289-299.

Christian, C. (1990) "Litigation support systems - one small step for credibility." *New Law Journal*, November 9, 1990, 1567-1568.

Cornwell, D. (1990) "Innovative applications of full text retrieval". *Computers and Law*, October, 7-8.

Cross, G.R. & deBessonet, C.G. (1985) "Representation of legal knowledge for conceptual retrieval." *Information Processing & Management*, **21**, 1, 35-44.

Dick, J.P. (1987) "Conceptual retrieval and case law." *Proceedings of First International Conference on Artificial Intelligence and Law*, 106-114.

Galloway, T.Y. (1987) "TAXI: a taxonomic assistant." *Proceedings of AAAI-87*, 416-420.

Gedeon, T.D. & Mital, V. (1991) "An adaptive learning network for information retrieval in a litigation support application." *Proceedings of AMSE International Conference on Neural Networks*, San Diego.

Gelbart, D. & Smith, J.C. (1990) "Knowledge-based information retrieval systems." *Computers & Law*, October, 1990, 23-26.

Gentner, D. & Toupin, C. (1986) "Systematicity and surface similarity in the development of analogy." *Cognitive Science*, **10**, 277-300.

Goel, A. (1989) "Integration of model-based reasoning and case-based reasoning for design problem solving." *Ph.D. Thesis*, Ohio State University.

Hafner, C.D. (1981) *An Information Retrieval System*. Ann Arbor, Research Press.

Halverson, J.T. (1979) "Use of computer for manipulating information: a trial lawyer's guide to document storage and retrieval." in J.H. Young, M.E. Kris and H.C. Trainor (eds.) *The Use of Computers in Litigation*, American Bar Association.

Hammond, K. (1988) *Case-Based Planning*, San Mateo, CA, Morgan Kaufman.

Holyoak, K.J. & Thagard, P. (1989) "Analogical mapping by constraint satisfaction." *Cognitive Science*, **13**, 295-355. Keane, J.I. (1989) "The automated litigator." *Trial*, January, 1989, 30-38.

Kim, W. (1990) "Object-oriented databases: definition and research directions." *IEEE Transactions on Knowledge and Data engineering*, **2**, 3, 327-341. Leishman, D. (1990) "An annotated bibliography of works on analogy." *Int. Journal of Intelligent Systems*, **5**, 43-81.

Lewis, D.D., Croft, W.B. & Bhandaru, N. (1989) "Language-oriented information retrieval." *Int. Journal of Intelligent Systems*, **4**, 285-318.

Martin, C. (1989) "Indexing using complex features." *Proceedings of DARPA Case-Based Reasoning Workshop*, 26-30.

Mital, V. (1990) "Case-based design and conceptual information retrieval in domains with a weak theory: financial/legal planning and litigation support applications." *Brunel University Technical Report* no. CSTR-90-7, available from author.

Mital, V. & Johnson, L. (1991a) *Information and Knowledge Systems in Law and Finance*, London, Chapman & Hall.

Mital, V. & Johnson, L. (1991b) "Professional negligence and financial-legal expert systems: architectures to enable the reasonableness defence." *International Journal of Law and Artificial Intelligence*, **1**, 1.

Mital, V.; Stylianou, A.K. & Johnson, L. (1991) "Litigation support systems: an object-oriented approach to conceptual information retrieval." *International Journal of Systems Research and Information Science*, to appear.

Patel-Schneider, P. (1990) "Practical, object-based knowledge representation for knowledge-based systems." *Information Systems*, **15**, 1, 9-19.

Rau, L.F. (1987) "Knowledge organization and access in a conceptual information system." *Info.Processing & Management*, **23**, 4, 269-283.

Rissland, E.L. & Ashley, K.D. (1989) "HYPO: A precedent-based legal reasoner." in G.P.V. Vandenberghe (ed.) *Advanced Topics of Law and*

Information Technology, Hillsdale, NJ, Lawrence Erlbaum, 213-234.
Schank, R.C., Ed. (1975), *Conceptual Information Processing*, Amsterdam, North Holland. Southam, M.; Mital, V. & Thomas, P. (1991) "HyperNotary: Legal document assembly using hypertext." *Proc. 13th BCS Information Retrieval Colloquium*, Lancaster.
Stylianou, A.K. (1990) "An Object-like Knowledge Representation Language (OKRL) for knowledge and information representation." *Brunel University Tech. Report* no. CSTR-90-9.
Tong, R.M., Appelbaum, L.A. & Askman, V.N. (1989) "A knowledge representation for conceptual information retrieval." *Int. Journal of Intelligent Systems*, **4**, 259-283.
Wallwork, N.J. (1989) "Managing documents in complex litigation." *Legal Economics*, November/December, 1989, 52-57.
Weaver, M.T., France, R.K., Chen, Q-F & Fox, E.A. (1989) "Using a frame-based language for information retrieval." *Int. Journal of Intelligent Systems*, **4**, 223-257.
Wilkins, R.P. (1989) "Computerizing your litigation practice." *Trial*, April, 1989, 82-83.

1.Another line of research is predicated on the twenty five year old assumption that the formal aspects of text, such as the relative frequencies with which a particular word occurs in a particular document and in the document database in general, can predict the meaning or subject content of the text (Blair, 1990). This line of research is of more relevance to the connectionist approach to artificial intelligence (Belew, 1987; Gedeon & Mital, 1991) than to the symbolic processing stance implicitly taken in this paper.

2.The litigation team may have first tried to deal with the information by manual cataloguing and indexing, until the problems became overwhelming (Berul et al, 1980). The possibility of an early settlement may have been in the air (though, some enlightened litigators use LSS as an aid to settlement itself (Keane, 1989)). Discovery from multiple parties may have taken place asynchronously or at a late stage.

3.More correctly, an instance of the explanatory link class object relates instances of objects in the issue and primitive domain concepts class lattices. A similar comment will apply to the description of reference links.

4.There is no global theory of relevance of concepts to issues in the domains likely to be litigated about and universally applicable relations are difficult to state(cf. Ashley (1989)).relevance of concepts to issues in the domains likely to be litigated about and universally applicable relations are difficult to state (cf. Ashley (1989)). Yet, given the facts of a situation, it is possible for a lawyer or a paralegal thoroughly familiar with the case, to usually say that a document is likely to be relevant to a particular issue, and that it is so relevant because of the presence in the document of references to certain concepts.

5.Actually, rather than linking and relinking RF's (which are complex objects, much memory/storage management would be needed for reorganisation), the discrimination is done by means of special objects called RF-skeletons. Every time an RF is specified a corresponding RF-skeleton object containing ***attributes*** that are equivalent to the ***values of the attributes*** of the RF is created: the rectangles in Figure 8 represent RF-skeletons which have been so created. The discrimination algorithm involves checking whether there exists in the hierarchy a RF-skeleton class object which has attributes such as to subsume the new RF-skeleton. If it does, the new RF-skeleton is made an instance of the existing class object: in Figure 8, the class object DIS subsumes two new RF-skeletons. If no such class exists, a new class is created (the user being consulted as to the location of the class where more than one location is possible). Further details are given elsewhere (Mital et al, 1991). RF-skeleton class objects accumulate a list - not shown in Figure 8 - of unique identifiers of all those documents which contain RF's with identical attribute values. By extension, by means of siblings in the hierarchy, we can also have direct access to those documents which contain RF's with only one attribute differing in value, or two, and so on.