Legal Information Retrieval: A Hybrid Approach

Daniel E. Rose
rose%cs@ucsd.edu

Richard K. Belew
rik%cs@ucsd.edu

Cognitive Computer Science Research Group
Computer Science & Engineering Department (C-014)
University of California at San Diego
La Jolla, CA 92093

Abstract

SCALIR is a legal information retrieval system which uses a combination of symbolic and connectionist artificial intelligence techniques. Traditional systems used for automating legal research have many difficulties which make them "brittle"; SCALIR is an attempt to rectify many of these problems. SCALIR's hybrid nature is especially appropriate for the legal domain, which requires both logical and associative inferences. The system also benefits from a unique direct-manipulation style interface and the ability to improve its performance based on user feedback.

1 Introduction

[0] The legal system is remarkable in its reliance on both precise and imprecise concepts. Computers traditionally work with exact problems, exact rules, facts, and conclusions. It is not surprising, then, that the most successful Artificial Intelligence (AI) systems are those which try to handle situations which best fit this mold. It is also natural that researchers would develop different techniques — such as connectionist AI — to deal with the inexact concepts of the real world. Given the dual nature of the law, we have designed a conceptual Legal Information Retrieval (LIR) system which combines both types of knowledge by using aspects of both AI paradigms.

SCALIR (Symbolic and Connectionist Approach to Legal Information Retrieval) is an interactive, adaptive system for aiding legal research on copyright law. When completed, it will contain all federal copyright cases, relevant statutes, and associated terms, as well as the citations which interconnect them.

We begin by demonstrating the motivation for our approach. Next we describe how the SCALIR system works. Finally, we examine the types of problems it can solve.

2 Some AI Background

Artificial Intelligence has produced two rather different paradigms for making computers "smarter." The better-known of these, which we call *symbolic* AI, is based on the idea that the core of intelligence lies in the explicit, largely sequential manipulation of symbols [New80]. By contrast, the *connectionist* or "neural network" view rests on the belief that intelligence is an emergent property of the interconnection of a large number of very simple processing elements (like neurons in the brain), all operating in parallel and communicating only "sub-symbolic" information [RMtP86].

Since widespread interest in the connectionist approach has only recently been rekindled, it is not surprising that all previous work on AI and Law falls into the symbolic category. For example, rule-based expert systems (e.g. McCarty's TAXMAN [McC77]), semantic networks (as used in Hafner's LIRS [Haf81]), and frame/script/schema models (such as Rissland's HYPO [RA87]) are all instances of symbolic AI.

3 Motivation

Computer-assisted legal research (CALR) systems have been in use for many years. LEXIS[1] and WESTLAW[2] terminals are found in many law offices and law libraries; law students often receive special training in the use of the systems. Yet there is at least anecdotal evidence that lawyers are not satisfied with them. Why? What is it about legal research that is not adequately addressed in LEXIS and WESTLAW?

3.1 The Problems of CALR

To answer this question, we will briefly examine how these systems work and what they are lacking. Consider a problem from copyright law: Does home taping of television shows with a VCR constitute copyright infringement? Users of a CALR system are taught to form a Boolean query which uses terms describing the issues involved. There are many possible queries, but a reasonable one for this issue might ask for cases which contain all three of the terms `VCR`, `copyright`, and `infringement`. There are refinements which allow alternate suffixes to be searched and so on, but the current example will suffice for our purposes.

Let us suppose that our hypothetical user is concerned primarily with Supreme Court decisions. He or she selects a database containing all federal cases relating to intellectual property. When run on one CALR system, this query produced a listing of five cases, none decided by the Supreme Court, and none directly addressing the issue in question. Indeed, several of these cases dealt with copyrights on books or songs, and only mentioned the term `VCR` incidentally. The situation is even worse when the user only looks at the Supreme Court database; he or she would then be told that there are no cases relevant to the query.

Experienced users are taught to try their queries again with synonyms, narrowing or expanding their scope, and so on. They also read over the retrieved (but irrelevant) cases in hope of finding more information. This process is made difficult in part by limitations of the interface. Users can see a list of cases found or the text of one case at a time, one screenful at a time. Refining a search means reformulating a query, often with more terms and more Boolean connectives, and getting an entirely new list of retrieved documents. In our example above, a tireless user, reading the full text of the fifth case originally retrieved, might discover a reference to the landmark case on the subject: *Sony v. Universal*[3], decided by the Supreme Court in 1984.

Why wasn't the case retrieved originally, since it deals with exactly the issues mentioned in the query? The reason is simple: the court referred to the devices in question by the older term `VTR` (videotape recorder) rather than the queried term `VCR`. This is an example of what AI researchers call *brittleness*; a total inability to respond correctly when the system's input does not exactly match its stored knowledge. Its opposite is the desirable property of *graceful degradation*. If the system could be given conceptual requests, it might be able to give a better answer — perhaps not the best answer, but not an irrelevant one, either.

The problem is deeper than the brittleness of traditional systems, however. The underlying assumption of CALR is that legal research involves finding exact answers to exact questions. This assumption is true in some cases, but it does not tell the whole story. In fact, many lawyers prefer to do their research the old-fashioned way — using bound volumes. They browse through cases, following many leads in parallel and using analogical reasoning to find relevant material. Often the most successful research session is one in which new answers to new questions are found.

Another aspect of the problem is that the relevance of a document is a relative concept. The most relevant case on VCRs and copyright infringement may differ depending on whether the client is a movie studio or a school district which tapes educational programs. Retrieving *Sony* might be unwanted clutter for the firm which specializes in copyright law and deals with subtle distinctions of statutory language; the firm's attorneys would already be intimately familiar with the case.

One way to approach the relevance problem is to allow the users to give relevance feedback, pruning or expanding their search, and having the system learn better retrievals. At first glance, the idea of learning may seem undesirable in a legal research system. But judiciously applied learning techniques solve some of the problem of traditional systems. First, learning improves the accuracy of the retrieval. Cases which are misclassified or terms which are misused will be "corrected" by the system's users. Second, learning allows the system to adapt to changing terminology. While our system already gets around the VCR/VTR problem (see Section 5), learning eliminates it by essentially creating synonyms.

To summarize, there are several problems with the traditional approach to legal IR:

- brittleness caused by syntactic rather than con-

[1] LEXIS is a registered trademark of Mead Data Central, Dayton, Ohio.

[2] WESTLAW is a registered trademark of West Publishing Co., St. Paul, Minnesota.

[3] 464 U.S. 417, 104 S.Ct. 774.

ceptual searching;

- inefficiency caused by a question-and-answer interaction rather than an ongoing exploration of the space of related documents;
- the artificial division of documents into "relevant" and "not relevant" categories, instead of a display of relative relevance;
- an inability to adapt to changing terminology.

Some of these problems have also been pointed out by Jon Bing [Bin88].

The AIR system addressed these problems by using a connectionist network in an adaptive conceptual IR system for bibliographic data [Bel86]. The prospects of applying these techniques to the law have also been described [Bel87]; we have incorporated many of AIR's techniques into SCALIR.

3.2 Dual Nature of the Legal System

What sort of knowledge is embedded in the law? What sort of knowledge is required to understand it? Is it explicit "symbolic" knowledge which could be incorporated into a symbolic AI system using condition-action rules or frames? Or is it "sub-symbolic" knowledge, better handled by the emergent statistical properties of a connectionist network? We believe it is both.

In some ways, the legal system[4] is inherently symbolic. After all, laws are explicit rules of conduct, often including precise definitions of important terms. The law is frequently analyzed as a system based on an underlying logical structure. In some instances, a set of "legal primitives" is postulated [Hoh19]. Even in common law, there are explicit symbolic relationships reflected by the Shepard's[5] treatment phrases, or the taxonomy formed by West Publishing's key number system.

However, our legal system can also be viewed as operating in connectionist fashion. (French [Fre87] has made a similar observation, contrasting it with the rule-based Civil Law tradition.) Legal decisions cannot be captured by rules (see Berman and Hafner [BH88] for several illustrations of this problem). In accordance with the Common Law doctrine of *stare decisis*, cases are decided on the basis of precedent. Judicial decisions are the result of combining the holdings of previous cases on an issue and the special facts of the current case; they in turn have an effect on all cases decided afterward. Law is made by the repetition of this process in many cases on many topics in many jurisdictions. This process — global concepts emerging from the interaction of a large number of local decisions — is a defining characteristic of connectionist networks. Furthermore, legal language contains ambiguities and overlapping shades of meaning [AS87], just as ordinary natural language does, and some AI research now suggests that connectionist techniques are needed to handle them [HK87], [CS83].

Given that the symbolic and connectionist paradigms in AI have different strengths and weaknesses, it is sensible to take advantage of the best features of both. Indeed, there is a growing interest in such hybrid systems among the AI community [Dye88], [Hen89]. The need for a hybrid approach is even more pressing when both symbolic and sub-symbolic knowledge are inherent in the problem domain. As we have shown, the law is such a domain.

[4] We use the term to refer to the Anglo-American legal system, and U.S. federal law in particular.

[5] Shepard's is a registered trademark of Shepard's Division of McGraw-Hill.

4 Design of the SCALIR System

SCALIR's fundamental retrieval mechanism is the spread of a numerical quantity we call *activation* through a network which forms the legal database. Three types of nodes — corresponding to terms, cases, and statute sections — constitute the substrate of this network.

The spreading activation mechanism works as follows: A user indicates items of interest and, optionally, specific symbolic relationships to be examined between these items. The nodes corresponding to these items each receive a certain amount of activity. Activation spreads from each node to all its neighbors, then to its neighbors' neighbors, and so on. In addition, activity "jumps" across the network whenever it finds a node connected by the specified logical relationships. Eventually, the new set of active nodes is displayed as the response to the query. (Details of the procedure are given in Section 4.2 below.) The combination of associative (sub-symbolic) and logical (symbolic) retrieval gives SCALIR advantages over either technique; this argument is developed further elsewhere [RB89a].

4.1 How the Network is Constructed

It should be clear by now that retrievals depend on which nodes are connected to which other nodes. In fact, it is fair to say that the "intelligence" of such a system is in the connections. The problem is determining which connections should exist.

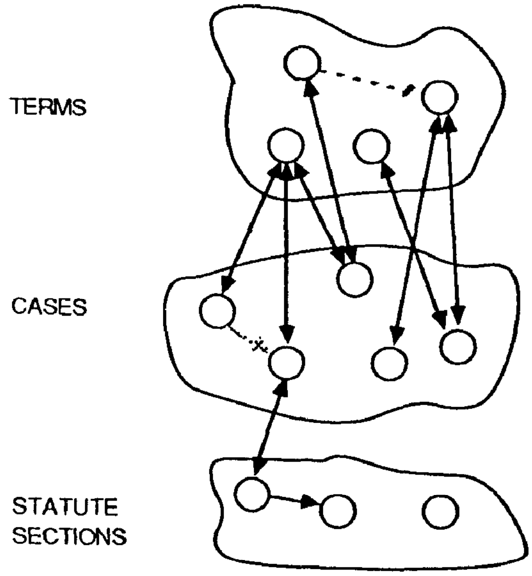

Figure 1: The structure of the SCALIR networks.

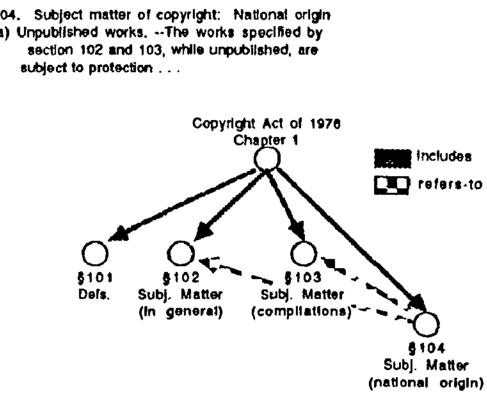

Figure 2: Some S-links for a statute

SCALIR uses two interleaved networks, a connectionist net embodying sub-symbolic information, and a semantic-type net embodying symbolic information. This is shown schematically in Figure 1.

The connectionist network is constructed by inserting *weighted, unlabeled* connections called C-links between the nodes. C-links correspond to "microfeatures" of the domain, sub-symbolic data below the level of conceptual features. One type of microfeature relationship is the fact that a certain word is used in a case; another is that two cases are often retrieved together. No knowledge engineering or editorial enhancement is required; all connections are formed by information directly available from the data, such as word occurrence. In addition, the weights on the links allow the system to improve its representation by learning from user feedback.

The initial terms and their weights[6] are chosen using techniques drawn from traditional information retrieval research. First, any words in a document appearing on a "noise word" list (consisting of about 250 prepositions, conjunctions, etc.) are eliminated. Next, the number of documents in which each word j occurs (the "document frequency" DF_j) is computed and normalized by the number of documents (N). Finally, the number of occurrences of each word j in document i ("frequency" F_{ij}) is also found. The basic idea behind the term weight is that terms which occur frequently in one document, but do not also occur in many other documents, will help to discriminate that document from the others. In order to prevent term weights from being dependent on document size, we add a normalization term; the one chosen was suggested by Salton [SB88]. This gives the following formula for term weight (TW_{ij}) for term j in document i:

$$TW_{ij} = \frac{F_{ij} \times \log(N/DF_j)}{\sqrt{\sum_{k=1}^{N}(F_{ik} \times \log(N/DF_k))^2}}$$

After the term weights have been calculated, the system creates C-links between the document and all terms whose weights for that document are above a certain threshold. The initial link weights are set proportional to the relative term weights.

The semantic network is formed by the explicit symbolic or editorially determined relationships between cases. These connections, called S-links, are *labeled* (with the type of relationship) and *unweighted*. A typical S-link between cases would be one labeled "overturned-by," indicating that the first case was overturned by the second. Since S-links represent factual relationships and previously acquired knowledge, they are not modifiable.

For statute nodes, S-links reflect structural relationships, such as the fact that a section falls under a certain chapter. In addition, S-links indicate show dependencies between statute sections, such as "except where noted in section x" or "as defined in section y." An example of this is shown in Figure 2.

The topics specified by the West key numbers for Copyrights will be included as term nodes. These will be related by structural S-links as well. Since the top-

[6]Note that this differs from the method used in AIR, where all terms were initially weighted equally.

ics represent a taxonomy of the relevant legal issues, these S-links can be considered a form of the IS-A link common in semantic networks [Bra83].

It is also interesting to note that our representation can be viewed as a natural generalization of more traditional methods in information retrieval [Bel89]. For example, Salton's vector space methods involve a measure of similarity between a query and a document that is based on the inner product of two vectors formed by the sets of weighted keywords associated with each [SM83]. Our associative retrieval methods compute a very similar "first order" match score between keywords and documents. However, our connectionist matching process also considers many other, "higher-order" associations as well. In fact, there are infinitely many possible paths between elements of a query and retrieved items, each providing a separate chain of evidence that the item is relevant (or not). Our retrieval methods have been tuned to consider a great many of these paths without becoming computationally intractible.

As a result of its incorporation of symbolic and subsymbolic knowledge, SCALIR is able to handle both "associative" and "logical" queries. For example, the first type of query might correspond to the question "what cases involving VCRs are related to *Sony v. Universal*?" while the second might be "were any cases overturned by the decision in *Sony v. Universal*?"

As of this writing, a prototype version of SCALIR has been partially implemented for preliminary testing. It contains the full text of all 120 Supreme Court cases dealing with copyrights, and the terms associated with those cases, and the C-links relating them. The addition of the Copyright Act, the West key number taxonomy for Copyrights, and their associated S-links is in progress.

4.2 Details of Hybrid Activation

In a typical connectionist system, each unit has a real-valued activity which propagates to its neighbors. The activity of the units is determined by taking the sum of the activity contributions of its neighbors multiplied by weights on those links, and passing it through some type of threshold or "squashing" function. This is shown schematically in Figure 3. The "knowledge" in such a system is in the link weights, which generally have been trained to produce the correct behavior.

The search in a semantic network, while also classified as spreading activation, has a quite different character. Rather than real-valued activities, there are generally discrete markers passed between neighboring nodes. Links are also labeled with their type. The markers may also be of different types, and the decision about

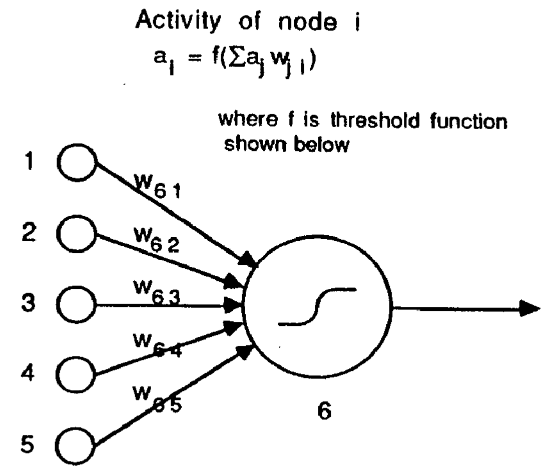

Figure 3: Activity of a node in a connectionist network

whether to pass a certain marker along a certain link often involves a complex logical function of these labels.

Since SCALIR contains both C-links and S-links, it must provide a mechanism for communication between the two. This is accomplished by conceptually dividing up activity into a vector of several components. Each component corresponds to activity filtered by a certain type of S-link, except for one component reserved for the "unspecified" activity carried by C-links.

When activity traverses a C-link, all components of the activity vector are individually multiplied by the weight on the link. When activity traverses an S-link, only the component which matches the link type is allowed to pass; the others are set to zero. An example of the two kinds of propagation is shown in Figure 4. We can think of the different components of activity as different wavelengths (colors) of light. C-links serve as grey filters, modulating the intensity of all wavelengths. S-links are like colored filters, allowing only light of the right wavelength to pass.

This approach has two immediate benefits. First, and perhaps most obviously, it supports both strict associative (connectionist) retrievals and logical (symbolic) retrievals. Second, the logical operations are completely local to individual nodes. For example, suppose one is interested in only cases which supported a certain decision. The user could draw a "supported-by" link to an unspecified case. The first node would thus be instructed to pass activity only along the corresponding vector. After activity traversed the first S-link, however, no other nodes would need this instruction. Due to the filtering of the S-links, only the desired compo-

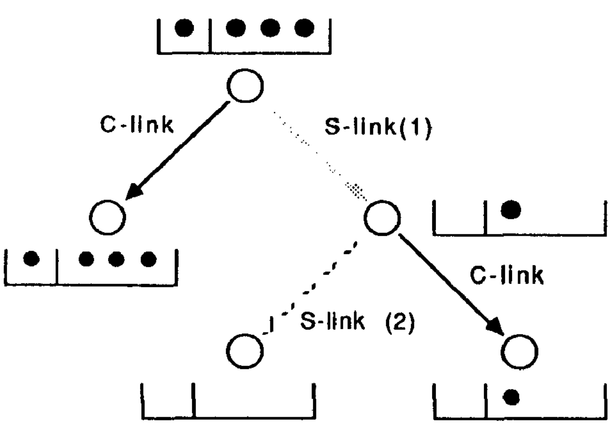

Figure 4: Hybrid activity propagation

nent of activity would be present at successive nodes. It is as if all other S-links were instantaneously disabled. This feature makes the system especially amenable to future implementation on parallel hardware.

The retrieval set presented to the user is chosen by identifying the most active nodes at each time step, where "most active" is defined as being above a certain significance threshold. As activity dissipates through the network over time, the significance threshold decays. This allows the inclusion of nodes related to the query more subtly by many small pieces of evidence. The process stops when enough nodes have become active, or when the total amount of activity has fallen to a quiescent level. We view the relative activation values of nodes as relative relevance of their associated documents or terms. While we have not fully implemented this feature, we intend to give users some control over the amount of data retrieved, either by allowing them to control the significance threshold or by displaying less active nodes off the edges of the screen and letting the user scroll to find them.

4.3 Interacting With SCALIR

In a traditional CALR system, users type textual queries using a boolean query language. Responses are then displayed serially, usually one screenful at a time. By contrast, SCALIR uses an interactive graphical interface [RB89b]. At the beginning of a research session, the user selects "dummy" nodes and labels them with the desired terms, documents, or statute sections. (This approach is reminiscent of the "query-by-example" technique used in some database management systems.) After this, however, the nature of the interaction changes dramatically. The system graphically displays the responses and their interconnections in one window of the screen. They appear in order of relevance, both temporally and spatially. From then on the user can search further simply by selecting displayed nodes (using a mouse) and "drawing" additional relationships to be queried. Relevance feedback can be used to automatically expand or prune the search.

Figure 5 shows one version of the interface. It contains several notable features. At the top is a menu of options which allows the user to modify the display; this version also contains commands which are used for incremental testing as the system is developed. At the left is the network window showing the current retrieval set and its relationships. C-links are displayed as solid lines whose thickness indicates their weight. No S-links appear in this figure, though they are displayed as various types of dotted and dashed lines, each corresponding to a certain type of relationship. Menu options allow varying complexity of the display; for example, the user might choose to view only the links in or out of the currently selected node.

Users "select" a node by positioning the mouse pointer over it and clicking the mouse button. This causes the corresponding text for that node — for example, the full text of a case — to appear in the text window at right, as the figure shows. The text window can be scrolled up and down to view any part of the case; the title of the case remains visible regardless of the scroll position. Recently viewed cases can be quickly redisplayed by selecting their corresponding nodes in the network window.

One added benefit of this type of interaction is that all three types of nodes — terms, cases, and statute sections — can be used in *both* queries and responses. For example, a user could simply ask for cases "like" *Sony v. Universal*, or those "related to" section 107 of the Copyright Act of 1976. At the other end, the system would respond to queries not just with a list of cases, but also with relevant terms and statute sections. As a result, each response to a query is in effect a suggestion for further avenues of research.

An additional feature, also present in the original AIR system, is the integration of term and citation information. Both queries and responses can incorporate both. By contrast, term-querying and citation-following are completely separate operations in conventional systems.

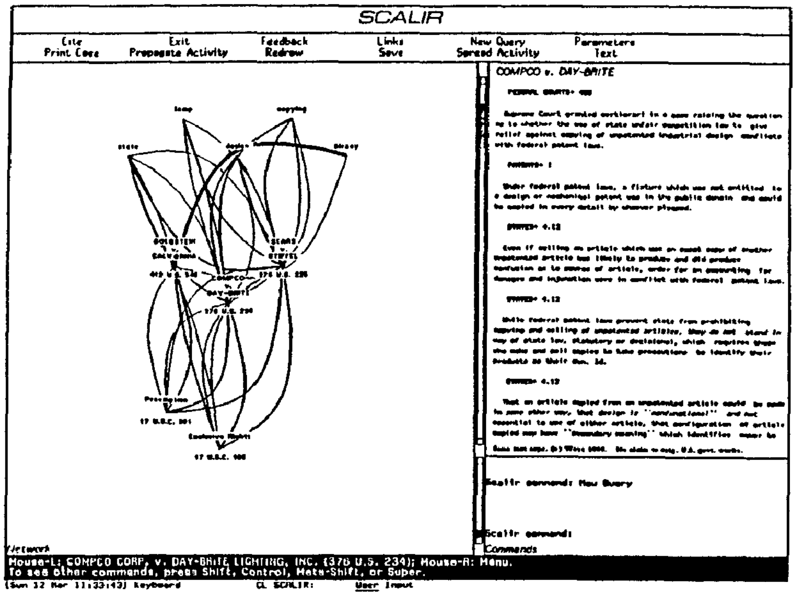

Figure 5: The SCALIR interface

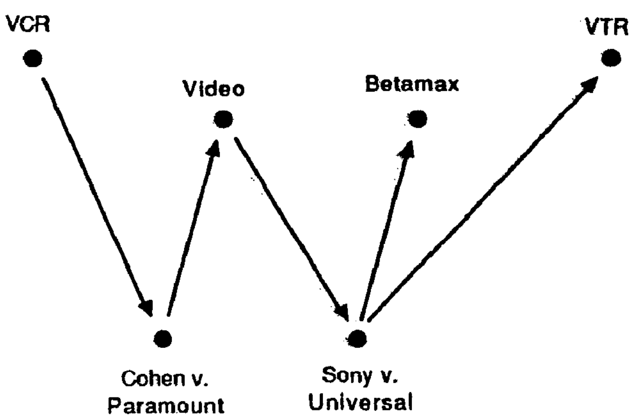

Figure 6: Some associated nodes

5 Examples

We now consider some simplified examples of SCALIR retrievals. Let us return to our VCR example. How would SCALIR find *Sony v. Universal*? Figure 6 shows a possible chain of associations. As we saw earlier, the term **VCR** appears in other (less relevant) cases, such as *Cohen v. Paramount*, which shares the term **video** with the *Sony* case. As a result, both *Sony* and the terms it uses are displayed.

At this point, the user is likely to give feedback to the system, wanting to explore the search further with the new terms. The nodes marked for positive feedback are considered to be "co-relevant," and are joined by new C-links. Those which receive negative feedback have their connection strengths reduced. In the future, a query to the term **VCR** would retrieve *Sony* directly, while the less relevant cases would no longer be displayed. This is shown in Figure 7.

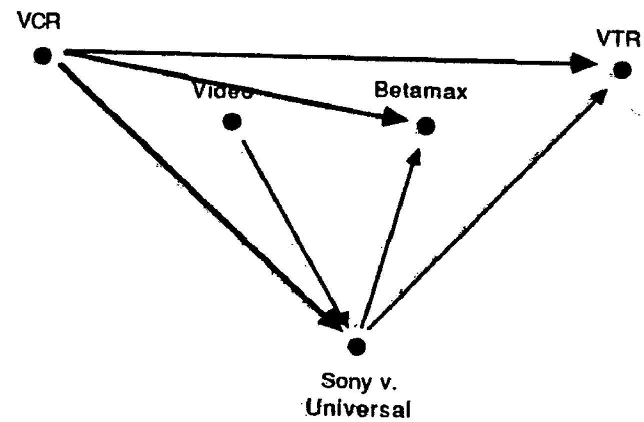

Figure 7: Retrieval after feedback

The previous example was concerned with simple associations, using only C-links. SCALIR can also make inferences based on S-links. For example, suppose that a user is interested in whether the formats used in a computer program can be copyrighted. In one of the relevant decisions (*Whelan v. Jaslow*[7]), the judge cited another case (*Synercom Technology v. University*

[7] 797 F.2d 1222.

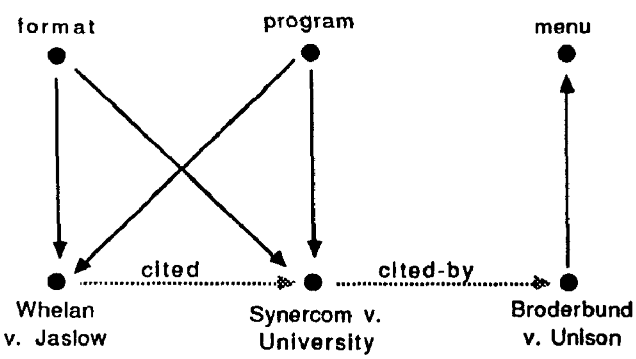

Figure 8: Citation chains

Computing[8]). As shown in Figure 8, SCALIR can pass activity along the citation chains, retrieving a third case (*Broderbund Software v. Unison World*[9]) which cited *Synercom*. Though not superficially relevant to the query, *Broderbund* — which deals with the related topic of whether menus and displays in a program are subject to copyright — would still be retrieved.

6 Future Work

SCALIR's hybrid learning feature has yet to be implemented. We believe this will add enormous value to the system, not only for the reasons mentioned earlier, but also because learning allows SCALIR to develop a shared body of expertise for its users, *without the need for explicit knowledge engineering*. For example, if SCALIR's user community is a law firm, associates in the firm will be able make use of what partners have taught the system about their interpretation of a term. Simply by using the system, they have incorporated some of their knowledge into it.

We will claim that the information structures built and used by SCALIR are more than simply a way to help lawyers do legal research more effectively. Rather, we view these structures as *artifacts* of the legal process. Just as an analysis of ancient tools can help us understand the users of these tools and their prehistoric cultures, the analysis of modern tools and their use by lawyers reflects important aspects of the underlying law. The indexing systems developed by information vendors (e.g., taxonomies, digests, citations, dictionaries) come to reflect the structure of the law. By incorporating as many of these structures as possible into a single representation, as we do in the SCALIR system, it may be possible to see regularities that are not obvious when they are considered alone. Further, because SCALIR's adaptive components are designed to capture regularities in the patterns of its users, we expect to identify additional structures in the law. The result is a representation that comes to mirror the law itself.

Much work remains before a full-scale version of SCALIR is complete. After testing the prototype, we will install the full text of all federal copyright cases (approximately 5000) and statutes. This version will be used for a study of the system's performance with actual users. We expect to complete this by the end of 1989.

[8] 462 F.Supp. 1003.

[9] 648 F.Supp. 1127.

7 Conclusions

There are many problems with traditional CALR. Some of these are due to an inability to deal with inexact information; others with the failure to integrate existing knowledge. Since the law makes use of both symbolic and sub-symbolic information, an AI system for facilitating legal research should do so as well.

SCALIR uses a combination of connectionist and symbolic AI methods to integrate these two types of information. It overcomes many of the problems of traditional IR systems and provides a new interactive paradigm for computer-assisted legal research. In addition, SCALIR's ability to learn from experience allows it to form a valuable conceptual representation of what terms and cases mean to its users.

Acknowledgements

We would like to thank West Publishing Co. for making their data available to us for use in SCALIR.

References

[AS87] Layman E. Allen and Charles S. Saxon. Some problems in designing expert systems to aid legal reasoning. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, pages 94-103, May 1987.

[Bel86] Richard K. Belew. *Adaptive Information Retrieval: Machine Learning in Associative Networks*. PhD thesis, University of Michigan, 1986.

[Bel87] Richard K. Belew. A connectionist approach to conceptual information retrieval. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, pages 116–126, May 1987.

[Bel89] Richard K. Belew. Adaptive information retrieval: Using a connectionist representation to retrieve and learn about documents. In *Proceedings of the 12th International Conference on Research and Development in Information Retrieval*, Cambridge, MA, June 1989. (to appear).

[BH88] Donald Berman and Carole Hafner. Obstacles to the developments of logic-based models of legal reasoning. In Charles Walter, editor, *Computer Power and Legal Language*, chapter 16, pages 183–214. Quorum Books, New York, 1988.

[Bin88] Jon Bing. The text retrieval system as a conversation factor. In Charles Walter, editor, *Computer Power and Legal Language*, chapter 11, pages 119–134. Quorum Books, New York, 1988.

[Bra83] Ronald J. Brachman. What ISA is and isn't: an analysis of taxonomic links in semantic nets. *Computer*, 1983.

[CS83] Garrison W. Cottrell and Steven L. Small. A connectionist scheme for modeling word sense disambiguation. *Cognition and Brain Theory*, 6(1):89–120, 1983.

[Dye88] Michael G. Dyer. Symbolic neuroengineering for natural language processing: A multilevel research approach. Technical Report UCLA-AI-88-14, Computer Science Department, University of California, Los Angeles, August 1988.

[Fre87] Robert French. Romano-germanic and common law traditions and their analogy with expert-systems and emergent, evolutionary approaches to artificial intelligence. Unpublished, 1987.

[Haf81] Carole D. Hafner. *An Information Retrieval System Based on a Computer Model of Legal Knowledge*. UMI Research Press, Ann Arbor, Michigan, 1981.

[Hen89] James A. Hendler. Marker-passing over microfeatures. *Cognitive Science*, 1989. To appear.

[HK87] S.J. Hanson and J. Kegl. PARSNIP: A connectionist network that learns natural language grammar from exposure to natural language sentences. In *Proc. Cognitive Science Society (9th)*, Hillsdale, NJ, 1987. Lawrence Erlbaum.

[Hoh19] Wesley N. Hohfeld. *Fundamental Legal Conceptions as Applied in Judicial Reasoning*. Yale University Press, 1919.

[McC77] L. Thorne McCarty. Reflections on TAXMAN: An experiment in artificial intelligence and legal reasoning. *Harvard Law Review*, 90:837–93, March 1977.

[New80] Allen Newell. Physical symbol systems. *Cognitive Science*, 2, 1980.

[RA87] Edwina L. Rissland and Kevin D. Ashley. A case-based system for trade secrets law. In *Proceedings of the First International Conference on Artificial Intelligence and Law*, pages 60–66, May 1987.

[RB89a] Daniel E. Rose and Richard K. Belew. A case for symbolic/sub-symbolic hybrids. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, Ann Arbor, MI, August 1989. (Submitted).

[RB89b] Daniel E. Rose and Richard K. Belew. Toward a direct-manipulation interface for conceptual information retrieval systems. In *Proceedings of ASIS Mid-Year Meeting*, 1989. To appear.

[RMtP86] David E. Rumelhart, James L. McClelland, and the PDP Research Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, Cambridge, Massachusetts, 1986.

[SB88] Gerard Salton and Chris Buckley. On the use of spreading activation methods in automatic information retrieval. Technical Report 88-907, Department of Computer Science, Cornell University, Ithaca, NY, April 1988.

[SM83] Gerard Salton and M.J. McGill. *Introduction to modern information retrieval*. McGraw-Hill, Inc., New York, NY, 1983.