New demands on information retrieval provide opportunities for natural language processing to work together with proven statistical retrieval approaches.

David D. Lewis

Karen Sparck Jones

Natural Language Processing for Information Retrieval

The need for automatic *text retrieval* (TR), also known as *document retrieval* (DR) has caught the attention of researchers in natural language processing (NLP). This article explores DR's key properties, summarizes past experience in the field, and reviews various specific NLP research strategies targeting this form of information processing.

For the purposes of this article, we generally view as synonymous the older term DR and the newer term TR. Both involve retrieving texts—from paragraph to book length—for humans to read. In the past, DR pointed the reader to an offline document, usually a journal article or report. New technology makes it practical to store, search, and retrieve online all or part of a document's full text. However, because the technical requirements of the two are the same, we view DR as the general term, except when the distinction just made is relevant and when it is necessary to refer to TR as supplying the user directly with "end" text. We use information retrieval (IR) as a global term covering everything from DR itself to knowledge retrieval. This article concentrates on IR and on DR as an NLP task.

Document Retrieval

Within IR, DR is an important and proper task with its own distinctive properties, not to be confused with data or knowledge retrieval. For example, DR is for the user who wants to learn something by reading about it, as opposed to merely wanting a specific data item or question answered. Just because a user wants to read about, for example

```
cheap production methods for simple prefabri-
cated housing
```

does not mean the user has specific questions in mind, like

```
What are cheap production methods . . .
or
How do cheap and expensive methods . . .
differ?
```

Moreover, even if the user does have questions in mind, the aim is to retrieve overall information so these questions, as well as other questions prompted by reading the documents, are answered. This means that DR must find relationships between the information needs of users and the information in the documents—both considered in a general sense and neither directly available to the system.

Equally important, the relationship between the user's need and the text that meets it is not necessarily obvious. For example, users looking for information on production methods for prefabricated housing may find answers in

```
J. Kirk: Reed Mat Huts of Madagascar: Design
and Construction.
```

Retrieval depends on *indexing*, that is on some means of indicating what documents are about. Indexing requires an *indexing language* with a *term* vocabulary and a method for constructing requests and document *descriptions*. Indexing is the basis for retrieving documents relevant to the user's need. It has to be supported by a search apparatus specifying conditions for a match between request and document descriptions, as well as modulation methods, to alter these descriptions if no match is made immediately.

The aim of indexing is to increase *precision*, the proportion of retrieved documents that are relevant, and to increase *recall*, the proportion of relevant documents that are retrieved. In doing so, however, it has to deal with two kinds of problems:

- Those posed by the external context in which the search is done, typically involving few relevant documents and many nonrelevant documents; and
- Those imposed by the internal constraints of the task itself, produce uncertainty the retrieval system must overcome.

The main internal constraint is the *variability* in the ways a concept can be expressed [8]. This variability is partly a matter of language, such as

`prefabricated` *vs.* `unit construction`

where the notion of prefabrication is the same, and partly one of perspective, such as

`prefabricated` *vs.* `factory-made`

where views differ on how prefabrication is done. Another constraint is *underspecification* due to vague requests, such as

`cheap` *as* `economical production` *vs.* `cheap` *as* `low quality,`

or incomplete requests, such as

```
housing vs. temporary housing.
```

The difference between these last two examples is that in the first case, users may not recognize their own ambiguity and in the second may fail to provide sufficient detail. This characteristic DR problem stems from the user's ignorance prior to reading.

Another constraint is the *reduction* of documents to their descriptions, which loses information through being indirect, such as

`building` *for* `reed mat hut,`

or partial, such as

`construction` *for* `design and construction.`

Because full texts of documents are increasingly available online, the degree of reduction depends increasingly on the indexing method. But reduction is never completely avoided; a document's author always leaves much unsaid on a subject, and reduction is not always pernicious. Compact descriptions of a document's significant content may increase the efficiency of matching and the effectiveness of classifying textual material as relevant or nonrelevant, just as feature selection is critical in other classification tasks.

DR thus imposes conflicting demands on text descriptions, asking that they be normalizing, discriminating, and summarizing, as well as accurate. As a result, variations in indexing that increase precision usually decrease recall, and vice versa. Beating this tradeoff by increasing both recall and precision is the fundamental goal of index languages.

There are many types of indexing languages. Terms may be the same terms appearing in the text to be indexed (*natural* language) or may be limited to those from an artificial or *controlled* language whose design involves many of the concerns associated with treating meaning representation for NLP.[1] Indexing languages vary according to several factors:

- The form of and emphasis on terms and term relations;

[1] We use the term 'natural language' to mean taking indexing terms from the document itself and 'NLP' to mean natural language processing.

- Whether relations are implicit or explicit; and
- The mixture of *syntagmatic* (document or request specific) and *paradigmatic* (universally asserted) relations.

Natural languages may be used more widely, but hybrids are common, including natural terms combined with artificial relations, such as:

```
(hut MATERIAL (mat MATERIAL reed)) LOCATION
madagascar,
```

or:

```
(reed mat hut) OF (madagascar).
```

Wholly controlled forms might appear like this:

```
(UNIT CONSTRUCTION HOUSING)(MADAGASCAR).
```

Past Research

Tests of a wide range of indexing languages during the past three decades have produced fairly consistent (if not wholly expected) results [22, 24]. These tests have shown that indexing documents by individual terms corresponding to words or word stems produces results at least as good as those produced when indexing by controlled vocabularies, whether simple or complex, and whether produced by manual effort or automatic language processing. Furthermore, automatically combining single indexing terms into multiword indexing phrases or more complex structures has yielded only small and inconsistent improvements over the simple use of multiple terms in a query.

In contrast, statistical DR methods, which ease and enhance use of representations based on single terms, have provided significant improvements over such alternative approaches as Boolean querying [17]. Statistical DR methods rank documents based on their similarity to the query or on an estimate of the probability of their relevance to the query, where both query and document are treated as collections of numerically weighted terms. The query can be an arbitrary textual statement of the user's information need or might even be a sample document.

Statistical DR methods assign higher numeric weights to terms showing evidence of being good content indicators, causing them to have greater influence on the ranking of documents. The number of occurrences of a term in a document, in the query, and in the set of documents as a whole, may all be taken into account when computing the influence the term should have on a document's score. In addition, if the user indicates that certain retrieved documents are relevant, this information can be used to reweight and alter the query terms through a process called *relevance feedback* [18, 21].

The focus of this baseline statistical DR strategy is on tuning the representation to the current user request, rather than on anticipating future user requests in the document descriptions. This strategy has three major benefits:

- It allows for *late binding*. Complex concepts need not be anticipated during document indexing, but are under the control of each user at query time.
- *Redundancy* is supported by drawing indexing terms from the document text, rather than using a limited vocabulary that may not support a particular user's needs.
- The representation is *derived* from the documents themselves so that differences and similarities between document texts are given the best chance to survive into the document representations.

For example, a query presented to a statistical DR system might say

```
A cheap [20] method [5] for prefabricated
[30] housing [20].
```

The term weights 25, 5, 30, and 20 would be assigned automatically to the stems based on their statistics of occurrence in the documents. A document matching the query on the stems `cheap` and `prefabricat` would receive a high score. If the user indicated to the system that this document is relevant, relevance feedback would increase the weights on the words `cheap` and `prefabricat`. In addition, highly weighted terms, such as `unit` and `construct`, from the relevant document might be added to the request with their own weights. They could then promote a hitherto uninspected document through a joint match on `prefabric` and `unit`.

Research showing the effectiveness of statistical DR methods appears solid in tests done in various environments, including different subject domains, ranges of system parameters, say for weighting, and alternative evaluation procedures with distinct performance measures. The methods also generally apply to document routing against standing, rather than one-off needs and perhaps for coarser document categories. However, these studies have involved few documents (30,000 at most, usually fewer) compared to hundreds of thousands in operating DR systems and have largely neglected non-European languages. Moreover, these experiments have generally been *surrogate*-based, that is they use titles and abstracts distilled from full-document content with a high loading for what is especially important in the source. The approach also depends on users entering requests in the form of sensible topic specifications while including several terms for alternative matches.

In addition to these caveats to the success of statistical DR, the question also remains of why intuitively plausible improvements in document representation have had so little impact on effectiveness. Why is there so little gain from linguistic sophistication (e.g., from the use of syntactic role relations between terms)? Is it that NLP intended to produce sophisticated indexing has been inadequately done? Is it that our transformations of natural language, even when done well by humans, have been misdirected? Or is it that so much leverage was gotten by searching surrogates in previous experiments that little room for improvement was left? Still, with typical effectiveness results in the range of 30% to 60% recall or precision [17], there is considerable room for improvement, even if DR is an intrinsically coarse process. Further, the research results just described must be considered in the context of operational practice and of the new TR situation where full source texts, and not just their surrogates, are available for direct searching.

The State of DR

Users can now access thousands of bibliographic databases, mainly in surrogate form, through various services. The long-running debate on controlled vs. natural language indexing has become less important as many commercial databases now use both. Most searches in these databases are done for end users by professional intermediaries who know about database coverage, as well as about the controlled language and indexing practices with it. These intermediaries generally believe a controlled language is superior to natural language, although the controlled languages involve many different design options with no clear winners.

All the evidence suggests that for end-user searching, **the indexing language should be natural language,** *rather than controlled language oriented.*

However, searching well-cared-for bibliographic databases is no longer the only function DR deals with. DR sessions can now involve PC users scanning their hard disks for missing files or students searching thousands of Internet servers for an archived Usenet posting. End-user, natural-language searching is inevitable because neither opportunities nor resources are available for using intermediaries and indexers; when full text is available, it seems natural to search it directly.

IR research has been brought to bear against this flood of traditional and nontraditional data, with some success. Statistical TR systems suggested by DR research now range from PCs to 100-gigabyte service databases. However, the situation is far from satisfactory, with at least three classes of problems:

- Uneven penetration of the best methods into operational practice. Many systems still require Boolean logic or other user-befuddling query syntax. When natural-language querying is available, weighting may be unavailable or poorly applied, and relevance feedback is rarely supported. Word-stemming operations may also be unsatisfactory or poorly understood.
- Ignorance about applying statistical DR methods to large, heterogeneous databases, particularly to full-text documents. Test collections of this sort only recently became available. Experiments with them have verified that standard techniques work, but many surprises and problems have been encountered [10, 11].
- Most important, many end users have little skill or limited experience in formulating initial search requests or modifying their requests after observing failure. Even when relevance feedback is available, it needs to be leveraged from a sensible starting point.

Although research shows that natural-language indexing and searching are effective to a degree, users may wonder whether the simple strategies described earlier in this article can be improved. More discriminating methods may be necessary to pluck relevant documents from very large databases and to support the fine-grain definitions of relevance possible with detailed full-text documents. There are therefore two issues. One is whether natural-language indexing, perhaps of a more refined kind than statistically controlled use of single-word terms, is wanted, or whether controlled-language indexing is really what is needed. Both controlled-language indexing and more sophisticated natural-language indexing imply nontrivial NLP, so the other issue is whether the required NLP capabilities are available or in prospect, since large-scale human full-text processing is not a practical proposition.

These issues will be addressed in the context of NLP research, which is itself in an exciting and rapidly changing state. An increase of interest in robust processing, in processing large amounts of real-world text, and in statistical methods in NLP make this an opportune time to consider interactions between DR, and more specifically TR, and NLP.

A TR Research Agenda

All the evidence suggests that for end-user searching, the indexing language should be natural language, rather than controlled language oriented. Indexing, or selective text content characterization, is needed, but should be derived from the text, with redundancy and late binding to compensate for uncertainty. For interactive searching, the indexing language should be directly accessible by the user for request formulation; users should not be required to express

How should we define *the linguistic units of indexing descriptions, including the size and depth of text forms and representation forms?*

their needs in a heavily controlled and highly artificial language. This does not mean that the system cannot enhance users' indication of what they want; for example, with statistical data or concept definitions they may not be able to interpret in detail.

Evidence also suggests that combining single terms into *compound terms*, representing relatively fixed complex concepts, may be useful. Many controlled languages allow this, and it has been found effective to a degree when done "statistically" on a simple collocation basis in a text window. While compounds uncovered by grammatical analysis have typically been less effective than those found by statistical means [6], this may change in a TR context. In any case, grammatical and statistical methods are increasingly combined.

The proposal described in the following sections develops these themes and investigates the role NLP may now play in full-text searching. The proposal addresses three things:

- The "words," "phrases," and "sentences" that form individual document descriptions and express the combinatory, syntagmatic relations between single terms captured by the system's NLP-based text-processing apparatus;
- The "classificatory" structure over the document file as a whole that indicates the paradigmatic relations between terms and allows controlled-term substitution in NLP-based indexing and searching; and
- The system's NLP-based mechanisms for searching and matching.

Indexing descriptions. How should we define the linguistic units of indexing descriptions, including the size and depth of text forms and representation forms? For example, should we go for any words or for only nominal group heads, or for concatenated or case-labeled phrases? We propose well-founded simplicity for both the natural-language units taken from the text as inputs to the indexing process and for the natural-language or near-natural-language units in the indexing language descriptions output by the indexing process. Indexing units would be linguistically solid compounds, such as

```
prefabricated housing
```

or basic propositions, such as

```
produce(factory, house).
```

The success of this proposal depends on its details, which differ from what might be assumed from traditional NLP practice:

- Given the proven value of statistical weighting, any units that NLP produces should be filtered and weighted by the statistics of their occurrences in the database searched and perhaps in other textbases as well [10, 11, 15]. Weighting for phrases may differ from weighting for single-word terms to allow for their lower frequency and different distribution characteristics; it is also less well understood than for words [1, 5, 6].
- We also stress the importance of late binding and sensitivity to the uncertainty of evidence. Compound terms will not be identified as definitely occurring or not occurring in a document. Rather, each document will provide some evidence for the presence of each known concept. An occurrence of the syntactically checked noun phrase `prefabricated units` in a document would be good evidence for the presence of the corresponding concept. An occurrence of the verb phrase `(they) prefabricated units` in a document would provide only slightly less evidence for the noun-phrase concept. The occurrence of the two words in separate paragraphs would provide much less evidence, but more than the amount given by the presence of just one of the words or of a related word [5].
- Basic compound units, such as those just described, would not typically be further combined into frames, templates, or other structured units—unless knowledge retrieval (discussed later) is to be supported. The description of a document would be an unordered set of phrases and individual words. This approach applies whether compound terms are formed at document file time or introduced through requests at search time. The rationale is that more complex structures are labor-intensive to design and difficult to fill accurately, and that matches on even basic propositions are so unusual that finer-grained distinctions are unlikely to provide additional information.

Applying the appropriate natural-language procedure to extract all instances of compound terms should produce a reasonable representation for moderately sized documents. For very large full-text documents, further reduction may be needed to produce a summary representation of content not swamped by the idiosyncrasies of numerous subparts. One could restrict terms to those drawn from particular portions of the text or (better) account for both global and local structures of the document when matching [19]. Either way, statistical control in unit choice and weighting is required. Only experiment can show what forms of reduction are useful and not too costly.

Thus for processing individual texts, we propose representations in which words and compound terms can refer to concepts with a range of complexity, while the loose coupling among these items permits efficient and flexible matching. Experiments are needed to determine the precise form of these compound terms and how they should be selected and weighted, say, relative to their constituents. NLP can at least help justify compound-term selection by referring to the grammatical structure of text and perhaps characterizing internal term structure.

Resources used during indexing and searching. Increased recall and precision depend on finding a way for nonidentical terms to match. The traditional approach is through *normalization*, replacing several forms with a single canonical form. For example, stemming is a normalization based on morphology:

```
prefabricated, prefabrication –> pre-fabricat.
```

Semantic normalizations are also possible based on manually defined classes:

```
house, apartment, hut –> DWELLING.
```

or automatically detected but previously unrecognized, statistical associations in a document file:

```
house, lawn, gasoline –> CLUSTER-1738.
```

Any normalization applicable to indexing can also be used more flexibly during matching. Retaining original document descriptions has important advantages—notably fidelity—and relational knowledge can be invoked in a context-sensitive and adaptive way during searching. Relationships can be adjusted to suit individual queries either directly (e.g., through user browsing in a graphical display of associations) or indirectly (e.g., through inference from the user's relevance judgements). This strategy also avoids costly reindexing of the entire document file when alterations or additions are made to the system's paradigmatic knowledge.

NLP might also provide various paradigmatic information. In addition, under a model where term relationships suggest, rather than demand, normalization, any resource specifying relationships among terms can also provide paradigmatic information. For example, symbols in knowledge bases, expert-system rule bases, data dictionaries, and source code are usually given names that are natural-language words or compounds. The relationships between terms implied by these structures may be more useful for retrieving text in a particular domain than a general thesaurus.

Procedures for searching. For searching, what mechanisms should be used to set matching conditions and determine request modification? Should matching be loose or tight? Should modulation be free or constrained? Natural simplicity seems to be best, allowing straightforward element stripping or substitution in compound terms, such as using

```
cheap prefabricated housing
```

for

```
prefabricated building.
```

Permitting obvious relational relaxation or substitution is also appropriate, such as using

```
cause (building)
```

for

```
produce (factory, house).
```

The assumption is that statistics will be applied as a guide or control in iterative searching through selection and weighting. Explicit probabilistic models may be favored over alternative matching schemes for their ability to combine a variety of evidence, but all current models have difficulty dealing with complex descriptions and their elements, so more work is needed.

Comments. Drawing on past DR lessons, we propose that future TR and DR systems include simple flexible natural-language indexing forms, support devices, and use strategies. Such approaches allow and encourage users to concentrate on request development, which matters much more than document characterization, and to do so in a way that supports derivation, redundancy, and late binding. This approach is also both potentially economically viable, even for large volumes of material, and practical from the user's point of view, given modern interface technology exploiting windows. It is also appropriate for two particular full-text cases:

- Retrieving subtexts, including paragraphs; it is still necessary to index on significant concepts even for short, focused pieces of text; and
- Two-level retrieval, first coarse, then fine, allowing motivated zooming.

In principle, many indexing strategies are applicable either at document indexing or request search time, depending on space, speed, and portability factors. For instance, NLP might be restricted to queries, while proximity searching might be used to identify the same compound terms in documents. Besides the efficiency advantages of avoiding natural-language analysis of the document file, an interface applying NLP to queries can enhance access to an existing retrieval system without requiring changes in the system. Another tradeoff between efficiency and precision in matching would be to apply NLP only to documents scoring high on a word-based query. Even applying NLP to the whole document file involves tradeoffs between explicit indexing on compound terms—speeding queries, but increasing the size of access structures—and indexing only their components or generalizations of their components, such as stems. In other cases, both efficiency and effectiveness may dictate the same course, as when reduction is used in indexing. Careful design of the whole system is required to optimize the related factors, given their interdependencies.

For end users, natural-language indexing strongly related to actual texts is attractive, and while they are required to participate in search development, fast processing and multi-window displays make it easier to exploit available information sources. There are, however, challenges in ensuring that any user understands what is happening and both can and does, for instance, exploit a store of paradigmatic knowledge. It may be difficult to convey the significance of statistical data; and while artificial description forms, like predicate-argument structures, can be applied in TR in a way that is hidden, so users are not confused or repelled, it is still necessary to motivate retrieval output for the user and hence to link the indexing descriptions the system actually uses with comprehensible reports to the user.

Implications for NLP

From the NLP point of view, a generic challenge is whether the necessary NLP is even doable; specific challenges are whether nonstatistical and statistical data can be appropriately combined and whether data about individual documents and whole files can be combined, since documents should always be treated in their file contexts.

The demands imposed on NLP by this program differ from those in most NLP tasks. TR, even more than DR, tolerates errors in document representations. In addition, ambiguities in NLP system output, such as alternative decompositions of a sentence into phrases, can be assigned probabilities of correctness and used in a probabilistic indexing method [7]. On the other hand, NLP applied to documents must deal with vast amounts of variable-quality text from broad domains. User requests involve smaller amounts of text, but even more variability in form and content. Each of the three main aspects of our strategy—forming text descriptions, providing and exploiting terminological resources, and ensuring matching in searching—poses special NLP challenges.

For example, we left open the issue of which syntagmatic relationships between terms in text would suffice for those terms to form a compound term. Strategies for traditional, if partial, syntactic analysis, allowing processing of hundreds of megabytes of text have been tested for TR [10, 11], but traditional semantic analysis on a large scale has not been demonstrated. New approaches are also possible. Accurate, highly efficient syntactic taggers are available, and some compound terms, like head nouns and premodifiers, are easily extractable from tagged text [3]. Various strategies for finding important collocations in large corpora have also been developed [20], possibly representing an improvement over traditional IR methods for statistical-phrase formation. Compound terms must not only be generated, but selected and weighted. Methods for exploiting the discourse structure of large texts may be useful in identifying which terms are central to the content of a text.

NLP also has a role in automated and semiautomated acquisition of paradigmatic knowledge. Automated formation of clusters of related words is again attracting the attention of researchers, despite the technique's historical lack of success in DR. More linguistically motivated approaches, such as clustering based on syntactic context, may be an improvement over traditional strategies [12]. Leveraging of hand-coded resources, such as inducing semantic information from labeled training data or from machine-readable dictionaries, may be a more effective, if less general, approach.

Finally, the type of NLP also constrains the forms of matching that can be performed. For example, element stripping might be restricted to adverbs or to words not in a domain-dependent vocabulary, but these restrictions can be implemented only if NLP has marked compound-term elements with the necessary information. NLP need not be applied identically to queries and documents; the system might do a very careful extraction of compound terms from the request, while the system uses a quick and dirty approach to find compound terms in the vastly larger amount of document text. The resulting uncertainty in the document representation may be compensated for during the matching process. NLP applied to the user request might also help distinguish between request words that should be matched against documents and those that convey other information about user needs, such as *Please retrieve journal articles published after 1987 about . . .*

A general caution is needed about the prospects that simple NLP strategies will significantly improve TR effectiveness. Recent work in NLP makes heavy use of the context of a word as a clue toward its meaning. For example, methods similar to request/document matching in IR have been used for word-sense disambiguation. It is not surprising that when a document and request match on several words, individual matching words are likely to have the same word sense [14]. The matching process itself provides a kind of disambiguation. Another example is that words tend to be accompanied by paradigmatically related words in documents, and relevance feedback may add these words to the request, much like a paradigmatic knowledge base would.

Thus NLP techniques are challenged by the basic methods of statistical IR, which has apparently picked some of the low-hanging fruit off the tree. The result is that alternate statistical retrieval methods have had greater impact than alternate text representations [1]. This should not discourage research into NLP applications in IR, but suggest that researchers and practitioners examine IR tasks carefully to see where NLP provides added value.

DR vs. Data Retrieval

Within IR, we distinguish DR from other forms of retrieval. The rest of this article concentrates on the relationships of these other forms to DR, how they may be combined, and how NLP experience may be transferred from one form to another.

We define *data retrieval* as the case where file information is precoded for specific properties and where the conceptual categories for queries must be known in advance.

Natural-language access to databases can replace or supplement the use of formal query languages. It has been investigated for three decades and there are well-established commercial systems [4]. Natural language clearly offers advantages in convenience and flexibility, but also involves major challenges in query interpretation precisely because query expression is decoupled from search formulation. Input queries can require extensive transformation to map onto file categories, and this transformation may have to be mediated by a rich domain model. Ill-formed input can complicate the process. Thus, natural language front ends can be effective, but normally only after significant customization.

Thus NLP techniques are challenged by the basic methods of statistical IR, *which has apparently picked some of the low-hanging fruit off the tree.*

The specific difference between DR and data retrieval is that in data retrieval, the set structure for the query is critical and must be specified precisely. The quantificational structure of the input must therefore be identified through natural-language analysis. The user may have a vague query, but the query must still be interpreted in one or more definite ways for searching, like a DR Boolean query. Post hoc set specification, as when the user of a ranked retrieval DR system chooses how far to go down a ranking, is not allowed.

It is not clear that data retrieval experience is directly applicable to DR; the nature of the information base and type of need differ fundamentally, though development of natural-language analyzers for resolving predication structures in data retrieval is relevant to compound-term identification. However, DR techniques might be applied in data retrieval to provide "relaxed" queries automatically if the initial queries do not produce an answer. DR techniques might also be used to generate substitute or "partner" queries for searching accompanying text files. Finally, it is possible that the DR and TR techniques described in this article may be appropriate for databases with free-text field values and even more for so-called *record bases*, such as museum catalogues, which can include several free fields (containing up to a paragraph of text), as well as coded or controlled fields.

DR vs. Knowledge Retrieval

The relationship between DR and *knowledge retrieval*, or "question-answering," is especially interesting because knowledge retrieval is direct (like data retrieval) but uses less rigorous precoding. Knowledge retrieval thus requires more powerful inference capabilities than either data retrieval or DR.

It is sometimes supposed that replacing a document file with the knowledge base it embodies obviates the need for DR while allowing better IR. This scheme is useful in some contexts, though with high start-up effort for even limited types of texts, such as banking telexes [25]. But it is still desirable to access the writer's own presentation, which is one aspect of document content. Presentation is increasingly important for longer texts, and complete replacement with a knowledge-base version is much less feasible.

A potentially more useful strategy is to give DR more depth and integration through an organized superstructure over the file, which would be exploited as a knowledge base during initial searching. Document frames, or templates, supported by AI-type inference capabilities would give detailed, consistent, and linkable document characterizations. These structures could be used to regulate query-document matching, guide query modification, and focus browsing. Going further and using a propositional knowledge base would give a unified, high-level collection model, allowing more intensive inference. Many conventional DR approaches, such as faceted indexing and hypertext [16], can be viewed as gestures in this direction. The putative difference would be the explicit automatic inference.

For example, EP-X [16], a search intermediary system using a concept frame hierarchy, is an advance along these lines, but it is based on a controlled language and its knowledge base is constructed manually. Building such bases automatically from documents is very difficult, especially in a way that maximizes the derivation of information from the documents themselves, selects the important information in documents, and manages backup from base to individual documents. Some work has begun in this area [2, 9, 13], but in limited domains and by taxing NLP to its limits. Processing is also knowledge-heavy, so for wider and larger files, bootstrapping the lexicon is needed.

Although the knowledge base is supposed to encourage query development, which could include question-answering on the base itself, DR suggests that the right approach to knowledge base design is a simple structure embedding natural language, with rich text pointers. The knowledge base would contain semantic structures supporting inferences, but also include pointers to texts:

```
BUILDING
(TYPE: hut –> text 1
UNIT: mat –> text 1
MATERIAL: reed –> text 1
PLACE: madagascar –> text 1)
```

Such a structure would be hospitable to user queries and not too constraining. A good case can be made for the same type of structure as a means of linking different bases and base types within global systems. Different bases in such hybrid systems would all be treated as if they were document, or text, collections tied together to support "travels in information space" through associative lexical indexing [23].

Conclusions

Although conventional DR services continue to make heavy use of strongly controlled indexing languages (like the National Library of Medicine's *Medical Subject Headings*), indexing increasingly involves terms drawn from the natural language of documents. These simple natural-language indexing techniques have been shown adequate in many experiments, though not on a really large scale. These techniques are also beginning to be used for TR.

However, the greater information detail in full text apparently calls for more sophisticated NLP-based approaches to indexing and retrieval. We suggest that appropriate strategies for this new situation should follow the simple DR methods, extending them to handle compound terms and similar descriptive units. The required NLP technology is being established, and work on applying it to TR is beginning. There are major challenges in making the technology operate efficiently and effectively on the appropriate scale and in conducting the evaluation tests essential to determining whether the approach works and what specific form of it works best [22], especially when the tests involve interactive searching of large files. It is particularly necessary to show whether NLP-derived compound terms are significantly better than, say, simple collocational compounds.

The present surge in TR research, stimulated in part by the ARPA-sponsored Text Retrieval Conferences (TRECs) [10, 11], is a welcome effort. TREC is a major evaluation study with much more data than earlier experiments and comparisons of many different strategies—with and without NLP. However, it is too early to draw conclusions on relative merits, especially since tailoring to the TREC application must be discounted. The retrieval needs covered in the TRECs are by no means typical of many (or most) DR or TR contexts, so care is needed when transferring results, especially since interactive searching is not a primary object of study. Although these tests are on a larger scale than earlier tests, they still involve limitations. More importantly, it is too easy in DR, and hence in TR, to intuit wrongly that things do or will work well, whether these are old approaches, old approaches dressed up in shiny modern technological guises, or truly new approaches. It is essential to test, test, and test again.

References

1. Buckley, C. The importance of proper weighting methods. In *ARPA Workshop on Human Language Technology* (March 21–24, Plainsboro, N.J.). Morgan Kaufmann, San Mateo, CA, 1993, 349–352.
2. Chinchor, N., Hirschman, L., and Lewis, D.D. Evaluating message understanding systems: An analysis of the Third Message Understanding Conference (MUC-3). *Computational Linguistics 19*, 3 (1993), 409–449.
3. Church, K.W. A stochastic parts program and noun phrase parser for unrestricted text. In *Second Conference on Applied Natural Language Processing* (Feb. 9–12, Austin, Tex.). ACL, Morristown, N.J., 1988, 136–143.
4. Copestake, A. and Sparck Jones, K. Natural language interfaces to databases. *The Knowledge Engineering Review 5*, 4 (1990), 225–249.
5. Croft, W.B., Turtle, H.R., and Lewis, D.D. The use of phrases and structured queries in information retrieval. In *Proceedings of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Oct. 13–16, Chicago, Ill.). ACM/SIGIR, New York, 1991, 32–45.
6. Fagan, J.L. *Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods.* Ph.D. dissertation, Department of Computer Science, Cornell University, Sept. 1987.
7. Fuhr, N. Models for retrieval with probabilistic indexing. *Inf. Process. Manage., 25*, 1 (1989), 55–72.
8. Furnas, G.W., Landauer, T.K., Gomez, L.M., and Dumais, S.T. The vocabulary problem in human-system communication. *Commun. ACM, 30*, 11 (1987), 964–971.
9. Hahn, U. Topic parsing: Accounting for text macro structures in full-text analysis. *Inf. Process. Manage., 26*, 1 (1990), 135–170.
10. Harman, D.K., Ed. *Overview of the Third Text Retrieval Converence.*

National Institute of Standards and Technology Special Publication 500–225, Gaithersburg, Md., 1995.

11. Harman, D.K., Ed. Special Issue on the Second Text Retrieval Conference. *Inf. Process. Manage. 31,* 3 1995, 269–448.

12. Hindle, D. Noun classification from predicate-argument structures. In *28th Annual Meeting of the Association for Computational Linguistics* (June 6–9, Pittsburgh, Pa.). ACL, Morristown, N.J., 1990, 268–275.

13. Jacobs, P.S. and Rau, L.F. SCISOR: Extracting information from Online news. *Commun. ACM, 33,* 11 (1990), 88–97.

14. Krovetz, R. and Croft, W.B. Lexical ambiguity and information retrieval. *ACM Trans. Inf. Syst., 10,* 2 (1992), 115–141.

15. Lewis, D.D. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (June 21–24, Copenhagen, Denmark). ACM/SIGIR, New York, 1992, 37–50.

16. Parsaye, K., Chignell, M., Khoshafian, S., and Wong, H. *Intelligent Databases*. Wiley, New York, 1989.

17. Salton, G. Another look at automatic text-retrieval systems. *Commun. ACM, 29,* 7 (1986), 648–656.

18. Salton, G. and Buckley, C. Improving retrieval performance by relevance feedback. *J. American Society for Information Science, 41,* 4 (1990), 288—297.

19. Salton, G. and Buckley, C. Global text matching for information retrieval. *Science, 253* (1991), 1012–1015.

20. Smadja, F.A. From n-grams to collocations: An evaluation of Xtract. In *29th Annual Meeting of the Association for Computational Linguistics* (June 18–21, Berkeley, Calif.). ACL, Morristown, N.J., 1991, 279–284.

21. Sparck Jones, K. Search term relevance weighting—some recent results. *J. Information Science 1* (1980), 325–332.

22. Sparck Jones, K. *Information Retrieval Experiment*. Butterworths, London, 1981.

23. Sparck Jones, K. Fashionable trends and feasible strategies in information management. *Inf. Process. Manage., 24,* 6 (1988), 703–711.

24. Willett, P., Ed. *Document Retrieval Systems* Taylor Graham, London, 1988.

25. Young, S.R. and Hayes, P.J. Automatic classification and summarization of banking telexes. In *Second Conference on Artificial Intelligence Applications* (Dec. 11–13, Miami Beach, Fla.). IEEE Computer Society Press, Los Alamitos, Calif., 1985, 402–408.

About the Authors:

DAVID D. LEWIS is a member of the technical staff at AT&T Bell Laboratories. His research interests include information retrieval, machine learning, NLP, and evaluation. **Author's Present Address:** AT&T Bell Laboratories, 600 Mountain Ave., Murray Hill, N.J. 07974; email: lewis@research.att.com

KAREN SPARCK JONES is Reader in Computers and Information at the Computer Laboratory, University of Cambridge. Her research interests are in natural language and information processing, ranging from linguistic discourse modelling to practical application systems development. **Author's Present Address:** Computer Laboratory, University of Cambridge, New Museums Site, Pembroke Street, Cambridge CB2 3QG, England; email: sparckjones@cl.cam.ac.uk