Progress in the Application of Natural Language Processing to Information Retrieval Tasks

ALAN F. SMEATON
School of Computer Applications, Dublin City University, Glasnevin, Dublin 9, Ireland
E-mail: Smeaton@dcu.ie.

Techniques of automatic natural language processing have been under development since the earliest computing machines, and in recent years these techniques have proven to be robust, reliable and efficient enough to lead to commercial products in many areas. The applications include machine translation, natural language interfaces and the stylistic analysis of texts but NLP techniques have also been applied to other computing tasks besides these. In this paper we will examine and review recent progress in using the lexical, syntactic, semantic and discourse levels of the language analysis for tasks like automatic and semi-automatic indexing of text, text retrieval, text abstracting and summarisation, thesaurus generation from text corpus and conceptual information retrieval. Our own work on the application of syntactic analysis to the matching and ranking of phrases using structured representations of texts, will be included in the overview. Finally, the prospects for gains in terms of overall retrieval effectiveness or quality will be discussed.

Received December 1991

1. LANGUAGE, INFORMATION AND INFORMATION RETRIEVAL

Information is an abstract and ethereal entity which can exist in many forms and media but no matter where information is kept permanently it only becomes useful if it is made available to the right person at the right time. Providing timely access to relevant information has always been difficult and since the explosion in the volume of information especially in recent decades, effective access to information has now become a critical task. In order for human beings to communicate information to each other and to record it, we use either formal artificial languages like computer programming languages or mathematical logic, or more commonly, we use natural language.

Natural language originally evolved as a spoken communication mechanism and the evolution from spoken form to written has not really changed the form of the language much. Written natural language does have some differences from spoken, but in general these are not significant ones as we tend to write the same way as we would speak in terms of vocabulary and the grammatical constructs we would use. Even in its written form however, there exist a number of different styles of language which can be distinguished. Technical documentation as in repair manuals and software installation guides, tends to be terse and contain tight prose. Usually there are complex phrases and complex sentences needed as difficult technical information is often being conveyed. Sentences are mostly unambiguous and declarative in nature. Journalistic pieces like newspaper articles usually contain shorter sentences, mostly quite simple and easy to read. Story book prose as used in novels and books, can be complex but is usually halfway between newspaper and technical documentation in terms of complexity of the language. Formal language which is very terse and difficult to read is often used in legal documents like contracts and covenants. Finally, electronic mail messages may be ungrammatical, full of abbreviations and mis-spellings and may not contain full sentences at all. This variety in the type of language used in different applications means is that the term *natural language* can actually refer to a large number of types of natural language, depending on the application.

Information retrieval is a discipline dedicated to the development of effective means of accessing textual information of any type by using a computer. If a user has a vague information need then it can be expressed even imprecisely as a statement in natural language or as a boolean combination of keywords. The user thus requires access to information which itself has been encoded imprecisely in an ambiguous language, natural language. Simple retrieval methods like string searching, keyword searching or searches using keyword frequency information can be computationally efficient but they can be quite ineffective as well. For example, keyword-based retrieval cannot handle the following properties of natural language[7]

1. Different words may be used to convey the same meaning: 'Stomach pain after eating' and 'post-prandial abdominal discomfort' mean the same thing.
2. The same words may be used but they can have different meanings: 'Venetian blinds' and 'blind venetians'.
3. Different people may have different perspectives on the same single concept: 'The accident' v. 'the unfortunate incident' could be describing the same thing in a court case depending on whether you are for the defence or the prosecution.
4. The same words may have different meanings in different domains: *sharp* can be a measure of pain intensity in medicine or the quality of a cutting tool in a gardener's handbook.

Providing effective access to information expressed as natural language can only be successfully done by processing the actual language of the text rather than just the individual words that have occurred. However, arguably the most extensive aspect of natural language is

the problem of ambiguity of interpretation, which occurs at all levels of NLP as we shall see in the next section. Ambiguity and the kind of features mentioned above are inherent properties of natural language and make automatically processing it very difficult but not impossible.

Computational linguistics is the study and development of computer systems for performing automatic natural language processing. Rather than attempting to define a single universal grammar for natural language as would be done with theoretical linguistics, computational linguistics is concerned with the development of procedures for handling most cases of natural language and coping with occasional failures in the processing. The goals of computational linguistics are to develop actual systems for applications like machine translation and man–machine interfaces.

With decades of research behind it, practical, efficient and effective computational processing of natural language is now becoming commonplace in many systems. Until recently, information retrieval was like other potential application areas for NLP in that it could not use NLP techniques as they were neither robust, efficient nor reliable enough. Now that has changed and information retrieval research, which has expended so much effort over the last 30 years developing statistical and keyword based approaches which have always had obvious limitations, is now starting to use NLP approaches to the processing of text in a constructive fashion.

In this paper we will examine and review recent progress in using the lexical, syntactic, semantic and discourse levels of language analysis for tasks like automatic and semi-automatic indexing of text, text retrieval, text abstracting and summarisation, thesaurus generation from text and conceptual information retrieval. The following section gives a very brief overview of how natural language can be processed at different levels. In section 3 we look at how the lexical level of language analysis can be used in information retrieval. Section 4 looks at the same thing for syntactic level analysis and section 5 the same for semantic analysis. In section 6 we look at some discourse level phenomena in language, in particular anaphora and sublanguages, and how they affect information retrieval. Finally we conclude with a look at recent trends in NLP research and where we believe the blending of these two disciplines is headed.

2. AN OVERVIEW OF NATURAL LANGUAGE PROCESSING

In order to build the complex systems needed to process natural language the operation is usually divided into independent but co-operating tasks working at different levels of language comprehension. Originally the divide between the levels was clear-cut but now this distinction is very blurred and fuzzy. For information retrieval purposes, the levels of language processing that we are interested in are the lexical, syntactic, semantic and discourse levels. Detailed overviews of NLP techniques can be found in refs 11 and 12.

In order to process a sentence of a language, the elements or tokens of the language must be isolated and identified. For NLP, lexical processing operates at the single word level and involves identifying words and determining their grammatical classes or parts-of-speech so that higher levels of language analysis can take place. This usually consists of looking up a dictionary or lexicon, essentially a list of known words and their legitimate morphological variants like plurals for nouns, participles for verbs, etc. Morphological analysis, which we group as part of lexical processing, involves breaking down a word into morphological or sub-word components. Thus the word 'covering' would be broken into 'cover' and the affix '-ing'. Lexical processing would then legitimately determine this word as either the present participle of the verb to cover ('He was *covering* the court when the rain fell') or a singular noun ('The *covering* was torn so the pitch got wet').

Ideally, lexical processing determines one base form for each word, and one syntactic tag also but this does not always occur in English as many nouns can act as verbs and most noun plurals are created by adding -s, in the same way as the third person singular form of a verb is formed. These ambiguities are passed on to syntactic analysis for resolving.

Traditionally syntax is regarded as being either the structure of a sentence with the semantics meaning the actual content, i.e. the parts-of-speech and the set of rules acting on them in order to determine grammaticality, or the set of rules which determine which orderings of words are allowed. Research into syntactic analysis of natural language has primarily been concerned with the construction of wide-coverage grammars and the development of efficient parsing strategies. Grammar formalisms have also been studied which has led to a proliferation of suggested standards for building grammars but natural language has proved notoriously difficult to capture in its entirety as a set of rules as there are always exceptions to rules. This makes wide-coverage grammars huge and complex.

The aim of syntactic processing is to determine the structure of a sentence but in natural language the structure itself can be ambiguous which is sometimes due to lexical ambiguity in the earlier language processing. In the sentences 'I saw her duck' and 'Sheep attacks rocket' the structural ambiguities are caused by the underlying ambiguities with the words 'duck', a singular noun or a verb form, and 'attacks', a plural noun or the 3rd person singular verb form. However in the sentence 'I recognised the boy with the telescope' the structural ambiguity is a pure structural ambiguity as there is no lexical uncertainty with any of these words.

The main sources of structural syntactic ambiguity in English are due to the attachment of prepositional phrases, the construction of nominal compounds and the scope of co-ordination and conjunction. Prepositional phrases can be attached to almost any syntactic category in order to act as modifiers but the problem with prepositional phrases is in determining what they are supposed to modify. For example these two sentences

Remove the bolt with the square head.
Remove the bolt with the square wrench.

are both lexically and syntactically identical but there is genuine structural ambiguity as we do not know to what the prepositional phrases 'with the square head' or 'with the square wrench' should be attached. In the first sentence the attachment should be to the object, the bolt, and in the second it should be to the verb, remove. It

would be important to distinguish these in an information retrieval context for queries about square-headed bolts or about removing bolts using a wrench.

The case of nominal compounds occurs when a noun or nouns are used as modifiers of another noun, making a compound structure as in the phrase 'computer performance evaluation'. Here 'performance' which is a noun, modifies 'evaluation', another noun. 'Computer', a noun modifies either 'performance' or 'evaluation', but we don't know which and this problem with nominal compounds creates the structural ambiguity. Another source of difficulty with processing such compounds is determining what kind of relationship exists between the nouns. 'Fighter plane' is a plane made for fighting but a 'garden party' is a party held in a garden and a 'timber house' is a house made from timber.[10]

Nominal compounding is very common in formal and in technical English as a nominal compound is usually expressing something which is too complex to be expressed as a single word. The phrase 'judiciary plea bargain settlement account audit'[11] has 42 distinct structural interpretations so there is real ambiguity there.

Conjunction is one of the most frequently used constructions in natural language but the scope of the conjuncts, i.e. what is being conjoined, can almost always be ambiguous. We can get conjunction among the heads of a noun phrase as in 'Inspect the bearing cups and cones' and 'Inspect the hub and bearing components', but in the first case there is a structural ambiguity with respect to the existence of 'bearing cones' and in the second case with respect to the existence of 'hub components'. Similarly we can have conjunction among modifiers, among prepositional phrases, among clauses, among almost all constructs. Conjunctions are used in language to make it more concise but the price for this conciseness is the ambiguity which must be resolved at higher levels of language processing. Unfortunately ambiguity in sentences is potentially multiplicative rather than additive when it occurs more than once. This means that texts containing long and complex sentences as in technical and formal writing, likely to have many of these ambiguities.

Despite the aforementioned negative aspects, syntactic level language processing has a number of attractive features including the fact that it determines sentence structure and it can be made efficient but most importantly, the rules of syntax are general in nature and concepts like word class are abstract; this means that the process is domain-independent except for lexical input. The disadvantages with syntactic processing are with the unresolved ambiguities and the fact that it is not inherently robust at handling ill-formed input.

The semantic level of language analysis is concerned with meaning and focuses on broad questions like what type of knowledge representation framework should be used. This level of language analysis interprets things like

John only introduced Mary to Sue

as either

John did nothing else with respect to Mary except the introduction
John introduced Mary to Sue but to nobody else
John introduced Mary and no one else to Sue.

Generally, semantic level NLP has involved defining a formal language into which input text can be turned. Such a language should be unambiguous, have simple rules for interpretation and have a logical structure. These properties are exemplified by mathematical logic and the earliest attempts at representing meaning were in terms of this. More recently, artificial intelligence has tended to represent knowledge by specifying primitive or simple concepts and then combining or structuring them in some way to define more complex, real-life concepts.

One of the most commonly used representations of meaning in NLP applications are semantic networks of which there is no standard but many variants.[11] Essentially a semantic net consists of labelled nodes and labelled arcs. Nodes usually represent objects and arcs, whose labels come from a small set of types, represent connections or relationships. The attempt at capturing meaning by decomposing into a fixed 'vocabulary' of elementary predicates was taken to extreme by Schank's conceptual dependency representation which defined only a dozen primitive action types and all words in natural language were defined in terms of these primitives.[11]

Although semantic networks adequately capture permanent, universal objects and their relationships quite well, there are other more subtle features of natural language which must be addressed. For example notions of modality, possibility, necessity, belief and time are somewhat difficult to capture in this formalism. On another level, there are semantic constraints on what should make semantically sensible natural language statements. The following two sentences are both lexically and syntactically correct, but are semantic nonsense.

Freedom is dark green.
My closet is well-behaved.

The semantic level language analysis should be able to analyse grammatically parsed text into a knowledge representation formalism but should also 'parse' the semantics of the input and note and respond to semantic nonsense. This is because a sentence may have a number of semantic interpretations, possibly arising from a number of syntactic interpretations, and we should eliminate as many of these as possible. The sentence

I noticed a man on the road wearing a hat.

has two syntactic interpretations with the participial phrase 'wearing a hat' modifying either the man or the road. Semantic level processing should tell us that hats are worn by animate objects like men and donkeys, and the latter of the interpretations should be discarded. The difficulty with semantic processing however is that a large amount of domain knowledge is needed in order to eliminate the latter interpretation of the above sentence and to process the meaning of that sentence. We need to know all the properties of all objects, the legitimate arguments of all verbs and building a knowledge base to support this is a huge task, so much so that systems which do process language at this level tend to operate in very restricted and narrow domains in order to make the associated knowledge base manageable in size and complexity. With the exception of the Cyc project at MCC in the US,[18] there are no attempts to build large, freely available knowledge bases off-the-shelf to allow

semantic level NLP techniques to move to new domains easily.

Discourse level language analysis is concerned with the study of context-dependent meaning, the meaning of an entire conversation or text taking into consideration things like who is reading and writing it, knowledge of the world, etc. This level of analysis wrestles with things like presuppositions as in 'The King of America is here' supposing the existence of a King of America and indirect speech acts as in 'Can you sit up?' being either a yes/no question from a hospital visitor asking about a patient's health, or an instruction from a visiting doctor. Discourse analysis tries to ascertain the subtle hidden meanings in spoken and in written texts.

There are a number of discourse-level phenomena which are of interest to information retrieval applications, in particular anaphora. Anaphora is a phenomenon of abbreviated subsequent reference to refer back to an entity introduced with more descriptive phrasing earlier by using a lexically and semantically abbreviated form. It is used to make language more concise and avoid repetition and the most common manifestation of this is in the use of pronouns. Anaphora reminds the reader of something and the more 'distant' the anaphoric reference from the target, the more detail is needed in the reference. For example in the first of the following passages the anaphoric reference *their* refers to the earlier target 'computers' while in the second passage the more expanded reference *such a system* refers to the target 'a centralised computer system'.

> Computers are often mixed up with questions about *their* impact on...
>
> A centralised computer system, on the other hand, can undergo many changes. Every time a new program is added to *such a system* the...

Detecting anaphora and resolving the reference would improve understanding of a text but even detecting anaphora is difficult as there are no indicator phrases or terms. Some words are potentially anaphoric but not always so and anaphoric references can include many constructs. For an information retrieval application, detecting and resolving anaphora is important as an anaphoric reference to subject-1 may occur with a reference to another subject, subject-2, and a user may wish to find information on some combination of subject-1 and subject-2. An example of this would be a query for 'computer system programs' in the second example above. Liddy[19] lists almost 150 words which could be indicators of an anaphoric construct and although many attempts have been made, the problem of reliably resolving anaphora still remains.

Natural language processing techniques are currently used in many applications including machine translation in the METAL system,[15] natural language interfaces[2,3] and text critiquing.[26] NLP can be performed efficiently as has been demonstrated by Siemens with the REALIST system which performs a syntactic analysis on 130 Mbytes of text in 18 hours on a 4.5 MIPS machine, or about 300 words per second[27] which is sufficient for indexing text but not for on-line interactive searching.

Although great strides have been made in the various levels of language analysis in recent times, we do not have fully semantic, interactive, domain-independent language processing of huge volumes of text which we can use for information retrieval, but do we need it for information retrieval functionality? It is believed by some that the problems that NLP research grapples with like anaphora resolution, quantifier scoping, modifier attachment, conjunction and structural ambiguities and others, are unimportant for traditional information retrieval. Sparck Jones has argued[33] that attempting to use AI techniques and natural language understanding for searching large text bases is not feasible at present and it is unclear whether fully-fledged sophisticated NLP would yield the desired payoff in terms of retrieval effectiveness.

However, fully-fledged NLP is being used in information retrieval and has led to the emergence of the application known as *conceptual information retrieval.* Some researchers believe the term conceptual information retrieval as used in the information retrieval literature to be improper as conceptual information retrieval is effectively a question answering system which is a well-established area in artificial intelligence as can be seen in Ref. 35 for example. Nevertheless, conceptual information retrieval should be distinguished from traditional information retrieval. In traditional information retrieval the user requests information and is presented with a list of texts which the system believes will contain the information the user seeks. In conceptual information retrieval the user requests information and is given the information directly, not just a reference to where it may be found. There is a tremendous difference in functionality between the two and conceptual information retrieval, which is more sophisticated, generally requires more sophisticated language processing as we shall see later on.

The most common use for NLP techniques in information retrieval is in indexing as a means to identify content indicators of various forms. This process can be done at the time of filing or data entry so the speeds at which NLP systems can operate are adequate for this task. In the next section we shall look at how lexical level NLP can be used in information retrieval.

3. LEXICAL LEVEL LANGUAGE PROCESSING IN INFORMATION RETRIEVAL

The simplest applications of NLP to information retrieval have been at the word level by indexing based on some normalised or derived form of individual words occurring in the input. An alternative to the popular stemming and conflation algorithms is to determine the base forms of words from a lexicon lookup. This has appeal in that it would always give the correct stem or base form provided that all words from the text are in the lexicon. Building such a lexicon is expensive however and has only given marginal improvements over mechanical stemming and for those reasons the idea has never really been pursued.

However lexical level language analysis has had a surge of interest recently with the increased availability of machine-readable dictionaries (MRDs). Originally derived from the typesetting tapes of published dictionaries, there are now several MRDs available for research purposes including the Oxford English Dictionary on CD-ROM. An MRD includes a definition for each sense or interpretation of a word usually including syntactic category, short textual description of the

meaning, morphology and perhaps semantic information like restrictions on verb arguments or subject classifications.

The obvious use of MRDs in information retrieval is to index texts and queries by word senses rather than by base forms or word stems and if this can be done accurately it would yield a more accurate description of the concepts in a text. Thus the word *bar* in the sentence 'The prisoner stood at the bar and awaited his sentence' would mean the court-room sense of bar and not a long piece of metal, a distinction on a medal, a fastener on a window, an immaterial restriction, a pub counter or place for refreshments, a large Mediterranean fish or a unit of atmospheric pressure. A similar criminal-related sense would be used for the word *sentence*. Word sense disambiguation however is quite difficult and despite the availability of MRDs no definitive technique has yet been discovered to do this.

In information retrieval experiments, indexing by word senses using MRDs initially gave disappointing results in terms of retrieval effectiveness. Because of this it is now believed by researchers that it may not be necessary to determine the single correct sense of a word but sufficient to rule out unlikely senses and weight likely senses highly. Krovetz and Croft have experimented with this approach and have included statistical term weighting but have not yet obtained the expected improvements in retrieval quality.[16] Zernick has tried clustering word sense signatures and used this in retrieval[36] but still no breakthrough has been obtained for multi-word or long queries although Zernick has obtained improvements in retrieval effectiveness for single word queries. It may be that indexing texts and queries by word senses will only improve the effectiveness of short queries but it is clear at this point that much more experimentation is needed in order to determine whether or not this is true.

4. SYNTACTIC LEVEL LANGUAGE PROCESSING IN INFORMATION RETRIEVAL

NLP techniques have been used to help index texts by elements more complex than word forms. Sacks-Davis *et al.*[28] have parsed texts and indexed them by syntactic labels indicating whether a word is a head of a clause or a modifier but they have not obtained significant improvements in retrieval effectiveness. However parsing of texts can also be used to generate more complex representations. It has always been assumed by researchers that in language it is the noun phrases that are the content-bearing units of information. This is not true for a full representation of meaning but noun phrases are good indicators of text content and for traditional information retrieval, that is what is wanted.

Syntactic analysis can be used to analyse text in order to determine the boundaries of noun phrases which could then be used as internal representations. Indexing texts on a noun phrase basis using NLP techniques was done in the IOTA system[7] but one of the major problems of indexing by noun phrase units is the variety of ways of representing a complex concept in natural language. For example, the following phrases all mean more or less the same thing but use different syntactic constructs:

Design *issues* for the performance of systems

System design performance *issues*

Issues on the design and performance of systems.

A word-word match would identify strong similarity between the above phrases but would also identify strong similarity with the following phrase

A *system* for the design of performance issues

which has a similar set of words but has little semantic overlap with anything, let alone the three earlier phrases.

Instead of just marking noun phrases in text, syntactic analysis could be used to identify the heads of each clause (italicised in the above) but that would leave us with problems of syntactic ambiguity as we discussed earlier. To address the issue of ambiguity in syntactic analysis of texts for indexing purposes there have been three approaches tried; ignore the ambiguity, normalise the identified phrases or index by structures which incorporate the ambiguities.

Ignoring the ambiguity allows texts to be indexed by phrases taken directly from the text. A large amount of work in this area has been done by Salton and others at Cornell University.[30] Here a parse of a text is used to identify head-modifier relationships between words from which indexing phrases are generated and used as index terms. In terms of retrieval using this representation, because document texts and queries are both indexed by phrases, the phrases can be used in the same way as single word index terms in that statistical approaches or vector space retrieval may be used.[29]

The approach of normalising indexing phrases from texts and from queries into some standard form is being taken by the CLARIT project at Carnegie Mellon University. A first order thesaurus for a domain, essentially a phrase list, is first generated automatically. Input texts are parsed and candidate noun phrases are identified. These are then compared to the thesaurus and classified as either exact (identical to some phrase in the list), general (terms in the list are constituents of those in the candidate set) or novel (new terms not in the list). This approach always uses terms from the list as the indexing units and thus always yields the same syntactic form for a concept which could have been expressed in a number of different ways. This means that the indexing vocabulary is quite small and conventional retrieval techniques, vector spaced or statistically-based could be used.[8] The CLARIT project approach to indexing has not yet been evaluated in terms of the effectiveness of the resulting retrieval performance and we await results from that.

The third approach to handling syntactic ambiguity in syntactically based indexing is to encode the ambiguity in some structure and allow the retrieval or matching operation to make allowances for this. The technique has been adopted by Siemens in the TINA/COPSY project, by Metzler in the COP project and by the present author in the SIMPR project. The TINA/COPSY project at Siemens builds dependency trees from noun phrases identified from a shallow parse where the dependency trees identify explicit links between words. The links are all of equal importance and optimistically represent all possible head-modifier relationships. Thus the input phrase 'Problems of fresh water storage and transport in containers or tanks' taken from Ref. 31 would be represented as the dependency tree in Fig. 1.

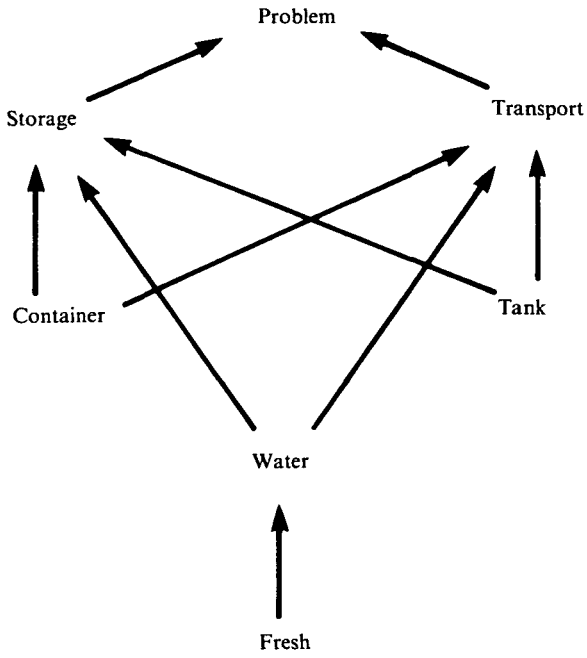

Figure 1. Dependency tree for phrase.

The dependency trees may be used in a number of ways. For example, the matching of two phrases and their associated dependency trees would be needed in retrieval where one dependency tree could be from a query, and another used as part of the index representation for a document. The phrases 'water storage' and 'storage problems' should generate exact matches with the dependency tree in Fig. 1. 'Water transport' generates an exact match in the Siemens system but this assumes that the original phrase deals with water transport and there is a genuine structural syntactic ambiguity here which the dependency trees ignore. The phrase 'storage problems' generates a match with the original phrase but an inexact match as the dependency is transitive via either 'storage' or 'transport'. Phrase matching generates a partial ranking of phrases based on the degree of match although Siemens have not published details on how this is done.

Another application of the Siemens dependency trees is in helping a user to formulate a query by using frequency information about the distribution of dependency links in the indexed database to allow the user to home in on a set of dependencies known to be in the text base, but as with phrase matching, this still has to be fully evaluated.[31]

The constituent object parser also builds dependency trees from a syntactic analysis where the trees are binary and the dominant branch at each node containing the head is marked.[22,23] The COP dependency trees cater for syntactic variants of the same concept or for identifying a simple concept embedded in a complex phrase. The phrase matching in COP is also a graph isomorphism exercise which looks for the same words with the same dependency relationships with the dominance relationship assumed to be transitive, which is not always true for natural language. Dependency links in COP also consider distance and the nature of the path along the tree to quantify the degree of strength of the relationship between terms.

The COP matching procedure can become quite complex if there are many words in common between two dependency trees and at present the best way to score and rank tests from which dependency trees have been generated, has not been determined and so its real evaluation remains to be done.

At Dublin City University we have been working on scoring the degree of match between phrases which is an operation ultimately needed in document retrieval. Working as part of the CEC's ESPRIT SIMPR project, we analyse text at the syntactic level and from the analysis we build tree-structured analytics (TSAs). TSAs are binary trees which encode rather than enumerate structural syntactic ambiguities as markers on non-leaf nodes. For example, the phrase 'Remove the fuel pump sediment bowl and filter from the top of the pump unit' would generate the TSA shown in Fig. 2.

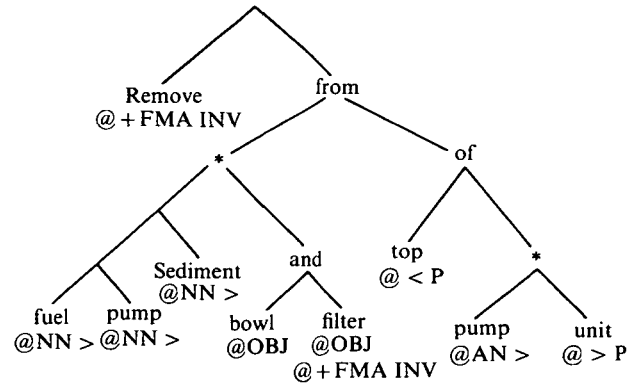

Figure 2. An example TSA.

The leaf nodes of TSAs contain the function words from the input, plus their syntactic labels. The label @NN > assigned to the word 'sediment' indicates that that word is a noun modifying a noun to its right. We don't know from this level of language analysis whether the modification is to the word 'filter' or not i.e. we cannot determine the scope of the quantifier 'sediment', so the labels and the TSA structure encode this ambiguity.

The matching algorithm we have developed for TSAs allows us to get an exact match on phrases like

- sediment bowls
- fuel pump sediment bowls
- pump units

but successively lesser degree of match on phrases like

- sediment filters
- pump filters
- fuel filters
- sediment units
- fuel units

In the algorithm we have developed for this we consider, quantitatively, the words, their syntactic label(s), their roles in phrases as heads or modifiers, the strength of evidence for those roles by checking ambiguity markers, and the residual structures occurring in TSAs between query term occurrences. In terms of evaluation and effectiveness our TSA match procedure performs well statistically in tests of phrase ranking,[32] but has yet to be scaled up to full document indexing and retrieval.

While most of the work on incorporating language

processing techniques into information retrieval tasks has concentrated on indexing and retrieval, some other IR-related tasks have also received this kind of attention. Automatic abstraction or summarisation of large texts into smaller executive summaries would be a great time-saver for many professions. At the Universität Passau an input text can be parsed into a representation which recognises noun phrases, heads and modifiers and from this analysis the dominant concepts in a text are statistically determined.[25] These concepts are related to each other and then 'verbalised' to generate an abstract. Because it operates at the syntactic level only, and does not consider verbs which would determine the relationships between objects, this approach to automatic abstracting will always be crude because the application, automatic abstracting, requires consistently accurate representation.

The automatic generation of a thesaurus from a body of text which determines phrasal relationships would be useful in automatic or semi-automatic query expansion. One of the features of language which makes information retrieval so difficult is that a single concept may be expressed in a number of apparently unrelated ways. If a user asks for information on 'prenatal ultrasonic diagnosis' then documents containing the following texts should also be retrieved:[13]

in utero sonographic diagnosis,
sonographic detection of fetal ureteral obstruction,
obstetric ultrasound,
ultrasonics in pregnancy,
midwife's experience with ultrasound screening,

...etc. A phrasal thesaurus which identifies relationships between phrases which have few or no words in common, would obviously be of use in query formulation and/or retrieval. The Siemens NLP group have experimented with syntactic methods of thesaurus generation by identifying heads and modifiers from a corpus of text and using a variety of similarity measures to identify term associations. Results to date indicate that this method finds semantically similar terms given an initial term, but evaluating the quality of the generated thesaurus is very difficult.

The big assumption made by those who use syntactic level NLP for any information retrieval tasks is that the syntax or structure of the language is indicative of the semantics or meaning, i.e. structure implies content. This is obviously not true for many cases, especially when the structure is ambiguous, but the supposition is made that it is true for much language. Because of that belief, using syntactic structure in indexing or retrieval in whatever way is suitable is a good thing to do provided the expectations in terms of retrieval effectiveness are tempered by this realisation.

In applications where aspirations of near-perfect knowledge representation are sought then a higher level of language analysis like semantic processing is needed to eliminate whatever syntactic ambiguities remain and to provide a deeper and richer representation. We shall see how this is done in information retrieval in the next section.

5. SEMANTIC LEVEL LANGUAGE PROCESSING IN INFORMATION RETRIEVAL

Any piece of text or dialogue which contains information essentially consists of a description of objects and actions on those objects. In order to capture the true meaning of text both the objects and the actions should be encoded and single keywords, word senses, syntactic labels and structured representations of noun phrases cannot do this. Representations like these capture indicators of content in the representation, but not meaning.

Building accurate semantic representations of information in most applications is a critical task and is usually done by hand. In information retrieval we cannot afford to have hand-built semantic representations of all texts so we do this dynamically during indexing. The most commonly used semantic representation of text in information retrieval is based on frames[11] but the disadvantage with frames as with most semantically-based representations is the large, domain-specific knowledge base needed to support their construction. An example of this type of knowledge would be domain-dependent scripts which describe typical sequences of events in the domain. These scripts help the slot-filling in frames by determining what kinds of things to look for to fill the frames. For example, a script for a person making an aircraft journey would search for fillers for frame slots like origin, destination, airline carrier, flight number, etc.

Because of the effort needed to encode the knowledge base needed to support semantic level NLP even in a narrow and restricted domain, systems which include this level of language processing tend to exploit the representation as much as possible and provide conceptual rather than traditional information retrieval, as we mentioned earlier. An exception to this is the MedInEx system from the National Library of Medicine in the US. This system uses a knowledge base of domain-specific information about medical terminology to process input texts and assist manual indexers in subject analysis of texts into a prescribed restricted vocabulary.[13] This ultimately leads to traditional rather than conceptual information retrieval which is unusual for the level of language processing used but in this particular application the sheer volume of text being indexed makes conceptual information retrieval impracticable at present. Another example of providing traditional information retrieval functionality using semantic level NLP can be seen in the RIME system which operates on the domain of medical reports.[2]

There have been a number of conceptual information retrieval systems described in the literature in the last few years. These include SCISOR, RESEARCHER and OpEd. SCISOR[14] is possibly the most well-known of these systems. It reads news stories about company mergers and acquisitions from the newswire, extracts information and stores it in a knowledge base and then it answers users' questions about the content. Input stories are parsed and analysed into the knowledge base using domain-dependent scripts built by hand for each application area, though the authors claim to have ported SCISOR to a new domain of military messages in 40 person-days.

The kind of user–system interaction that SCISOR is designed for is the following:

USER: Did ACE hardware take over the ACME food company?
SYSTEM: Yes, last Friday.
USER: What were the events in the ACE-ACME deal?
SYSTEM: Rumours that ACME was to be taken over started on 13 May. The stock rose $8 a share. On 16 May the company announced that...

As we can see, the interaction in SCISOR is a dialogue. The system has retrieved individual pieces of information from the texts it has processed, put them together and generated natural language responses to queries. The information needed for the second of the system responses above may have been scattered among more than one news story. Obviously in order to provide this kind of functionality the system must know and use discourse level phenomena in the dialogue part as above, and must also be able to process journalistic text when reading stories. SCISOR can read stories and add their information content to its knowledge base at a rate of about 6 per minute.

RESEARCHER operates in the domain of US patent applications. It processes the text of applications and is able to answer questions on their content.[17] The interesting part of RESEARCHER is that the NLP uses limited semantics to resolve syntactic ambiguity and then uses the knowledge assimilated from the whole of the patent application it is processing to try to resolve outstanding ambiguities. Texts are analysed into a frame-based representation and functionally, RESEARCHER is the same as SCISOR except it operates in a different domain and has a different knowledge base to support it.

SCISOR and RESEARCHER operate on news stories and patent applications respectively, both text types which contain descriptions of complex but real physical objects. OpEd is an editorial comprehension and question-answering system which answers questions about beliefs, belief relationships and goals of those who have made arguments in the input texts.[1] Thus the questions asked in OpEd are not answered by retrieving relevant facts from the knowledge base, putting them together and generating a natural language response, but by an understanding of the arguments in the input texts. The significant point about OpEd is that it demonstrates conceptual information retrieval from a very complex domain, and it seems to work.

The final example of conceptual information retrieval that we will look at is unusual in that it does not use semantic level NLP at all. The START system indexes text into T-expressions which are triples of the form ⟨subject relation object⟩ which can be recursive in order to handle embedded sentences.[9] T-expressions are normalised in the sense that they handle some syntactic variants. The retrieval in START involves turning the user's query into a T-expression pattern like ⟨Jessica want ⟨computer print???⟩⟩, which would be the pattern for the query 'What did Jessica want the computer to print?', and searching for T-expression patterns. This searching can be quite straightforward in some cases but START also has a set of rules which capture some types of syntactic variants and these rules define legitimate transformations on T-expressions. These transformations are applied during retrieval if there is no exact match on the pattern generated from the user's query.

START is unusual in that it is a syntactically based conceptual information retrieval but it has its limitations compared to such systems as SCISOR and RESEARCHER which use a semantic knowledge base. All of these systems however, are prototypes operating in restricted domains and have small knowledge bases but they do demonstrate that conceptual information retrieval functionality is achievable, though at a cost. When applying NLP techniques to information retrieval the tradeoff exists between the sophistication of the retrieval operation, conceptual or traditional information retrieval, and the level of language processing needed, semantic or lower levels with the ensuing restrictions on the domain.

Besides conceptual information retrieval, semantic level language processing has a role to play in other information retrieval tasks. Chris Paice at the University of Lancaster has been experimenting with automatically producing abstracts or text summaries by building frames or templates for these summaries with slots like 'aim of paper', 'purpose of study', 'results' and 'conclusions'.[24] Processing an input text will only partially fill some of these frames, but not all frame slots need to be filled as some texts will not have a conclusion, others will not state aims explicitly, etc. When the text is analysed, a coherent abstract can be generated automatically. This approach to text summarisation is still very much at the experimental stage, and one of the major difficulties is handling discourse level phenomena like anaphora and ellipses. In the next section of this article we shall look at how some discourse level phenomena can effect information retrieval and how they can be handled. We shall also look at the existence of sublanguages in particular domains and how this affects information retrieval.

6. DISCOURSE PHENOMENA IN INFORMATION RETRIEVAL

Earlier in this paper we looked at anaphora as an example of a discourse level construct which occurs in dialogue and text and which could have implications for information retrieval. Anaphoric constructs in language may hide the real statistical distribution of word usage in text by making abbreviated references to concepts mentioned earlier. An abbreviated reference to a concept may or may not contain some of the original words used to describe the concept, but because the anaphoric reference is present, mention of the concept is being made in the same way as if the full unabridged form were used. Obviously it is more important to determine the number of times a concept is mentioned in a text instead of the number of times a word or combination of words which may reference that concept, occurs.

The most extensive study of anaphora in information retrieval has been carried out at Syracuse University from about 1983 onwards. These studies are summarised in Ref. 19 and have

- examined the basic linguistic assumptions underlying the use of anaphora in text;
- developed a taxonomy of anaphora types;
- developed rules for distinguishing anaphoric from non-anaphoric use of potential terms;

- gathered statistics on the occurrence of anaphora in abstracts;
- examined the impact of anaphora resolution on a variety of term weighting schemes and on document ranking.

The results show that in a piece of text the size of a document abstract there are 12 potential anaphors with an actual use of 3.7 on average, so such constructs do occur quite frequently. The Syracuse team have developed a set of rules for anaphora resolution which do not replicate human cognition but which simply capture most of the linguistic patterns of anaphora occurrence. A simple rule for anaphora resolution would replace each potential anaphoric word occurrence by the nearest preceding word which matches the occurrence in gender and number and this would resolve 70% of potential anaphora of which 60% would be correctly done. Of course this rule would also substitute potential anaphors which are not actual occurrences and this would really corrupt subsequent retrieval.

Manually and thus correctly resolving anaphors in texts and performing retrieval on the resolved texts provided mixed results in terms of retrieval effectiveness. Some queries were improved while others were worse. This strange result is counter-intuitive as resolving anaphora correctly would seem to be a sensible thing to do. The premise on which most of statistically based information retrieval is based is that the more often a concept is discussed in a document the more important that concept is to that document, and revealing suppressed anaphoric references to a concept should expose the true statistical frequencies of concept occurrences. At present we do not know how to resolve anaphora reliably and when we do resolve them we don't know what to do with them. It is now believed that resolving anaphora by syntactic means may not hold the best hope for the future and the consensus seems to be that anaphora resolution should be treated with other discourse level phenomena and should form part of an overall semantically-based NLP on text.

As we have seen earlier, semantic level language analysis requires the encoding of much domain-specific knowledge and when this is done it generally leads to conceptual information retrieval but this is expensive in terms of resources and effort. The existence of sublanguages in certain applications in restricted domains could be exploited for information retrieval purposes. Different domains will generally have different vocabularies and terms, but the form of language, the syntactic structures, etc., will generally be the same. Not only may the vocabulary be different in a sublanguage with some technical terms being restricted to some domains only, but there will also be fewer words senses for other words. For example in a car manual the word 'brake' will mean either the verb to stop a vehicle or the noun referring to the actual device. It will not refer to bracken, brushwood, an instrument for crushing flax, a large wagonette or any other senses of that word. For these reasons, semantic analysis of text may be easier and less costly to achieve in restricted sublanguages and there are a number of examples of this in practice.

The TASLINK system[21] reads free-form text descriptions of cases of automobile stalling and transforms each text description into a standard representation. In this application the texts are terse, contain many ambiguities, mis-spellings and grammatical errors, but the system can process the input texts correctly. Liddy has developed a system for the analysis of completed questionnaires on life assurance applications.[20] These were completed by agents in the field and are full of ungrammatical utterances, abbreviations and misspellings, but nevertheless the system developed showed good results in terms of correct analysis of input. Finally, Wood and Sommervile have described a system which processes text descriptions of re-usable software components into frame-like knowledge representations.[34] In each of the three cases above there is specialist vocabulary and there are few senses for many of the words used. Each system demonstrates that the language analysis needed to support conceptual information retrieval can be done but the important thing is that the knowledge bases needed to support this type of retrieval are smaller because of the smaller vocabulary size and the restricted forms of the language used. This suggests that conceptual information retrieval is easier to achieve in narrow domains which have sublanguages.

7. CONCLUSIONS

This paper has outlined the four levels of language which are of interest to information retrieval tasks. At the lexical level it appears that machine readable dictionaries offer some interesting possibilities for indexing and representation of texts, but much experimental work remains to be done in order to determine how MRDs should be used effectively. Indexing by words senses using a MRD should lead to more effective text retrieval than indexing by word stems and the huge amount of research into statistical information retrieval over recent decades could be used on representations consisting of word senses rather than word stems.

Syntactically based approaches to indexing have also been the subject of recent investigation. The domain-independence of the language analysis is an attractive feature for information retrieval applications but one of the drawbacks with techniques based on syntactic analysis as we have seen to date is that they deal with sentence-level texts only and do not address issues like anaphora. A more serious weakness with statistically based information retrieval is that it has not considered the issue of synonymity between words or between phrases. Word–word synonymity, i.e. 'cheerful' is a synonym of 'happy', could be handled by word substitution. Phrasal synonymity as in 'prenatal ultrasonic diagnosis' is a synonym of 'sonographic detection of fetal ureteral obstruction', is more difficult to handle but is a fundamental stumbling block to effective retrieval. In semantically-based information processing, such synonyms would be indexed into the same frame-based representation but syntactically based information retrieval tasks must look at this problem if progress is to be made and as with the lexical level, much experimentation remains to be done.

One of the very interesting developments in computational linguistics over the last few years has been the emergence of statistically based language processing. Brown has run a statistical analysis on 3 million pairs of sentences, one in English and one the manual French translation, which came from the proceedings of the Canadian parliament.[4] The analysis determined the

probability of word adjacencies based on the context and the presence of other words. From these statistics an automatic machine translation system was developed with a 9000-word vocabulary which yielded a 53% success rate in terms of complete sentences translated with the correct meaning. Although this is computationally expensive, this work demonstrates that statistically based language translation is a possibility. Ken Church has developed a procedure for building probabilistic grammars which is being used to try to determine semantics from statistics.[6] These kinds of developments are exciting for information retrieval because of the possibility of really integrating statistically-based language analysis with the already developed methods of statistically-based information retrieval. Progress can be expected in this area in the future.

Regardless of the progress in traditional information retrieval, the ideal information retrieval system is one which provides conceptual information retrieval. Unfortunately, as we have seen, this usually requires semantic level language processing which in turn needs a large domain-specific knowledge base which makes the whole language analysis and conceptual information retrieval process restricted to a narrow domain. Semantic level language processing does not scale up to information retrieval size dimensions because of the difficulty of scaling up the supporting knowledge base and this looks like being the case for the foreseeable future. Until the next great breakthrough in natural language processing arrives, i.e. efficient domain-independent semantic level language processing, we will be stuck with this catch-22 position, but notwithstanding this, we have much experimental work to do to fully realise the potential of the currently available NLP techniques.

Finally, the role that NLP techniques currently play in information retrieval research is more or less an empirical role. NLP techniques are regarded as black boxes or tools to help provide better or richer indexing by phrases instead of by words, to provide graded matching of phrases, etc. This role does not really address issues of retrieving information for users based on the language used in queries or in texts. Fundamental issues and questions dealing with the notion of a retrieval model and document relevance will need to be integrated with what NLP techniques have to offer if really significant progress in retrieval effectiveness is to be expected.

Acknowledgement

The author would like to acknowledge the helpful comments made by Yves Chiaramella on an earlier draft of this paper.

REFERENCES

1. S. Alvarado *et al.*, Argument comprehension and retrieval for editorial text. *Knowledge-Based Systems* **3** (3), 139–162 (1990).
2. C. Berrut, Indexing medical reports: the RIME approach. *Information Processing and Management* **26** (1), 93–110 (1990).
3. J.-L. Binot *et al.*, Natural language interfaces: a new philosophy. *SunExpert Magazine* **2** (1), 67–73 (1991).
4. K. Brown *et al.*, A statistical approach to machine translation. *Computational Linguistics* **16** (2), 77–85 (1990).
5. Y. Chiaramella *et al.*, IOTA: a full text information retrieval system. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, edited F. Rabitti, pp. 207–213. Pisa (1987).
6. K. Church, A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the 2nd Conference on Applied Natural Language processing*, pp. 136–143. Austin, Texas (1988).
7. D. A. Evans, Concept management in text via natural language processing: the CLARIT approach. In *Working Notes for the AAAI Spring Symposium on Text-Based Intelligent Systems*. Stanford (1990).
8. D. A. Evans *et al.*, Automatic indexing using selective NLP and first order thesauri. In *Proceedings of RIAO '91*, pp. 624–643. Barcelona, Spain (1991).
9. J. Gaulding and B. Katz, Using 'Word Knowledge' reasoning for question answering. In *The Society of Text: Hypertext, Hypermedia and the Social Construction of Information*, edited E. Barrett, pp. 403–421. MIT Press (1989).
10. L. S. Gay and W. B. Croft, Interpreting nominal compounds for information retrieval. *Information Processing and Management* **26** (1), 21–38 (1990).
11. G. Gazdar and C. Mellish, *Natural Language Processing in LISP: An Introduction to Computational Linguistics*. Addison-Wesley (1989).
12. R. Grishman, *Computational Linguistics: An Introduction*. Cambridge University Press (1986).
13. S. M. Humphrey, A knowledge-based expert system for computer-assisted indexing. *IEEE Expert* **4** (3), 25–38 (1989).
14. P. S. Jacobs and L. F. Rau, SCISOR: extracting information from online news. *Communications of the ACM* **33** (11) (1990).
15. A. Joscelyne, Pedal to the METAL. *Electric Word* **17**, 33–38 (1990).
16. R. Krovetz and W. B. Croft, Word sense disambiguation using machine readable dictionaries. In *Proceedings of the 12th International SIGIR Conference on Research and Development in Information Retrieval*, edited N. J. Belkin and W. B. Croft, pp. 127–136. Boston (1989).
17. M. Lebowitz, The use of memory in text processing. *Communications of the ACM* **31** (12), 1483–1502 (1988).
18. D. Lenat *et al.*, Cyc: toward programs with common sense. *Communications of the ACM* **33** (8), 30–49 (1990).
19. E. DuRoss Liddy, Anaphora in natural language processing and information retrieval. *Information Processing and Management* **26** (1), 39–52 (1990).
20. E. Liddy *et al.*, Processing natural language for an expert system using a sublanguage. In *Proceedings of RIAO '91*, pp. 707–717. Barcelona, Spain (1991).
21. S. Lytinen, Robust parsing of terse text. In *Working Notes for the AAAI Spring Symposium on Text-Based Intelligent Systems*. Stanford (1990).
22. D. P. Metzler and S. W. Haas, The constituent object parser: syntactic structure matching for information retrieval. *ACM Transactions on Information Systems* **7** (3), 292–316 (1989).
23. D. P. Metzler *et al.*, Conjunction, ellipses and other discontinuous constituents in the constituent object parser. *Information Processing and Management* **26** (1), 53–71 (1990).
24. C. D. Paice, Constructing literature abstracts by computer: techniques and prospects. *Information Processing and Management* **26** (1), 171–186 (1990).
25. U. Reimer and U. Hahn, 'Text condensation as knowledge

base abstraction. Universität Passau Technical Report MIP-8723 (1987).
26. S. Richardson *et al.*, Experiences of developing a large-scale natural language text processing system: CRITIQUE. IBM Research Report RC 13644 (1987).
27. G. Ruge, Experiments in linguistically based term associations. In *Proceedings of RIAO '91*, pp. 528–545. Barcelona, Spain (1991).
28. R. Sacks-Davis *et al.*, Using syntactic analysis in a document retrieval system that uses signature files. In *Proceedings of the 13th International SIGIR Conference on Research and Development in Information Retrieval*, edited J.-L. Vidick, pp. 179–192. Brussels, Belgium (1990).
29. G. Salton, *Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*. Addison-Wesley (1989).
30. G. Salton *et al.*, On the application of syntactic methodologies in automatic text analysis. *Information Processing and Management* **26** (1), 73–92 (1990).
31. C. Schwarz, Content based text handling. *Information Processing and Management* **26** (2), 219–226 (1990).
32. P. Sheridan and A. F. Smeaton, The application of morpho-syntactic language processing to effective phrase matching. *Information Processing and Management* **28** (2).
33. K. Sparck Jones, What exactly should we look to AI, and NLP especially, for? In *Working Notes from the AAAI Spring Symposium on Text Based Intelligent Systems* (1990).
34. M. Wood and I. Sommerville, An information retrieval system for software components. *ACM SIGIR Forum* **22** (3/4), 11–28 (1988).
35. G. P. Zarri, An outline of the representation and use of temporal data in the RESEDA system. *Information Technology: Research and Development* **2** 89–108 (1983).
36. U. Zernik, On 'Diesel Train1 Engines' and 'To Train2 Airline Pilots': tagging word senses in corpus", In *Proceedings of RIAO '91*, pp. 567–585. Barcelona, Spain (1991).

Book Review

MICHAEL L. MAULDIN
Conceptual Information Retrieval – A Case Study in Adaptive Partial Parsing.
Kluwer Academic Publishers, Boston. ISBN 0-7293-9214-0. £43.25.

This book is about a conceptual information retrieval system called FERRET which was created by the book's author. It is stated that the motivation behind building FERRET was to get away from the limitations of traditional keyword-based IR systems and create a system that 'understands' the content of the documents it is retrieving. Using techniques based on the FRUMP news-skimming system (by DeJong), FERRET parses documents and the user's query, trying to fit them to one of a number of predefined scripts. Once a script has been found a case frame is built, and matching is done on these frames. Hence FERRET is restricted to parsing documents it has scripts for, limiting it to a small domain. In FERRET, the author has attempted to improve on FRUMP by using an on-line dictionary to try to give clues about words not in the system lexicon. He has also incorporated learning algorithms to improve the parsing scripts. Based on user reactions to the results of a query, two kinds of algorithm start up, one which tries to generalise a script so that more documents fit a query; and the other which attempts to mutate a script to allow it to understand new concepts. The latter is the most interesting part of FERRET, because if it is genuinely capable of learning to change its parser this approach to IR would have great potential.

The book is made up of seven chapters describing the motivations behind FERRET, a review of the literature, the system itself, testing, conclusions and future work. According to the publisher's comments on the back of the book, chapters 2 and 3 'serve as an excellent reference in the fields of NLP, IR and AI'. I am not really qualified to comment on the NLP and AI review, but I found the review of IR work, especially that of keyword systems, to be rather thin, summing up weighted keyword retrieval research in just under three pages. The description of the FERRET system in chapters 4 and 5 is well done, with many diagrams and examples to explain how everything works. Chapter 6 outlines the testing that was performed on FERRET. The centre piece of the testing is a comparative study between FERRET and a keyword-based IR system. Surprisingly, the author chose to pit FERRET against a Boolean system. Perhaps this is because each system's retrieval output is an unordered set of documents, thus making comparisons easier. However, it is generally accepted that weighted-keyword IR systems with a good stemming algorithm and relevance feedback outperform Boolean. So the significance of the result (FERRET winning easily), is lessened in the light of that knowledge. The more impressive result was the improvement in recall that the learning algorithms produced, raising it by 30%, although all this improvement came from the generalisation algorithm, with the mutation algorithm contributing nothing. Whether this was due to a failing in the algorithm or just an insufficient amount of training data is not clear.

The style, layout and order of the book are all good, making it very readable. Certainly if you want to know about FERRET, this is the book for you. As a book about conceptual information retrieval then, remembering the reservations outlined above, I would say that for someone relatively new to NLP and AI (like me) it provides a good introduction to the subject.

MARK SANDERSON
Glasgow

Announcement

7–9 OCTOBER 1992

11th International Conference on the Entity Relationship Approach, Karlsruhe, Germany

The Entity Relationship (ER) approach is extensively used in many database and information system design methodologies and has become a *de facto* standard of most manual and computerised design tools. Continuing its tradition, the 11th conference will provide an international and interdisciplinary forum in which researchers and practitioners can share novel research, tool developments and management experiences. The conference will consist of presented papers, invited papers, tutorials, tool demonstrations, and panel sessions.

Topics:

- Conceptual modelling and database design
- Federated information bases
- Innovative applications of the ER approach
- Security and integrity techniques
- Practical issues in database development
- Process modelling, characterisation and implementation
- User interfaces and multimedia databases
- Re-engineering of databases
- Quality control of database design aspects
- Automated design of information systems
- Functional design
- Query languages

For further information contact:

Professor Dr W. Stucky, University of Karlsruhe, AIFB, P.O. Box 6980, D-W-7500 Karlsruhe, Germany. Tel.: +49-721-6083-812. Fax: +49-721-6937-17. email: WST@AIFB.UNI-KARLSRUHE.DE.