A Summary of the CLARIT Project

David A. Evans, Steve K. Handerson,
Robert G. Lefferts, Ira A. Monarch

November 1991

Report No. CMU-LCL-91-2

Laboratory for
Computational
Linguistics

139 Baker Hall
Department of Philosophy
Carnegie Mellon University
Pittsburgh, PA 15213

A Summary of the CLARIT Project

Laboratory for Computational Linguistics

Carnegie Mellon University

David A. Evans, Steve K. Handerson, Robert G. Lefferts, Ira A. Monarch

November 1991

Contents

1 CLARIT Overview

The CLARIT[1] Project has focused on the problem of developing a robust, adaptable, and practical text-management system that integrates natural-language processing (NLP) with statistical, numerical, and heuristic techniques for identifying concepts in texts. In terms of functions, the Project aims to perfect (1) concept-based automatic indexing, (2) automatic thesaurus discovery, and (3) automatic identification of concept equivalence classes, in particular, in interpreting users' queries. In this context, the Project claims several innovations that extend the state of the art in information management.

1. **Selective NLP under Hybrid Parsing:** The Project has developed techniques for integrating independent NLP processes to identify selective syntactic structures (e.g., NPs) efficiently and accurately in unrestricted text. The processes currently involve a coordination of comprehensive morphological analysis (extending to classes of regular expressions), probabilistic syntactic tagging, and context-free chart parsing. Previously unseen vocabulary, proper names, noun–noun compounds, alpha-numeric phrases, and language fragments are accommodated (and parsed) gracefully.

2. **Phrasal Indexing using First-Order Thesauri:** The Project has developed techniques for automatically indexing unrestricted documents based on NPs. In addition, the Project has developed algorithms and methods for highly focused identification of *relevant* and *domain-canonical* terminology, guided in part by a simple list of domain terminology—a first-order thesaurus. The process first involves the identification of *candidate NPs* in a text. This is accomplished via selective NLP. The process then involves the filtering of the candidate NPs to produce *candidate index terms*. Filtering is accomplished by evaluating each candidate NP based on several 'scores'. One score is for the distribution characteristics of the NP (relative to the text in which it occurred, the domain corpus of the text, and general English). Another is for the precision of 'fit' between a candidate and terms in the thesaurus (allowing for exact and partial matches). The result is an ordered list of index terms. These are further distinguished as being 'exact' terms—certified terms in the thesaurus—'general' terms—broader terms from the certified set that subsume terms in the text—and 'novel' terms—previously unknown terms that, nevertheless, are prominent in the text.

3. **Automatic Thesaurus Discovery:** The Project has developed algorithms and techniques for clustering phrases in collections of documents to construct first-order thesauri that optimally 'cover' an arbitrary percentage of all the terminology in the domain represented by the document collection. The process involves the decomposition of candidate NPs from the documents to build a term lattice. Nodes are organized hierarchically from words to phrases based on the number of phrases subsumed by the term associated with each node. By selecting nodes that have high subsumption scores and that also satisfy certain structural characteristics (such as being legitimate NPs), it is possible to identify a subset of vocabulary that accurately characterizes the domain. With such a technique, very large numbers of thesauri can be identified quickly. These, in turn, can support numerous functions, such as automatic indexing, filtering, sorting, and routing of texts in information systems.

4. **'Latent Semantic Space' Models of Concepts:** The Project has begun to develop techniques to support the mapping of any query into a collection of index terms that represent

[1] "CLARIT" is an acronym for "Computational-Linguistic Approaches to Indexing and Retrieval of Text." The Project has been supported by grants from the Digital Equipment Corporation. All CLARIT processors, tools, and other resources have been developed in the Laboratory for Computational Linguistics, Carnegie Mellon University. An early description of the Project can be found in (Evans 1990).

the semantic equivalence class of the concepts in the query. The techniques exploit latent-semantic indexing to establish *lexical-item* x *concept* spaces. Extensions of this work promise to obviate the need for the use of semantic networks and knowledge bases to refine queries; and promise to provide a basis for automatically partitioning index terms into appropriate classification sets.

2 CLARIT Rationale and Methodology

Our hypothesis is that units of text greater than 'strings between white space' or keywords are required to capture concepts and extend information mangement beyond current limitations. The role of *selective* NLP is critical: it not only defines the units of information that are used in other processes but circumscribes the NLP task to insure that it is manageable and robust. However, we believe it is important to merge selective NLP with statistical, numerical, and heuristic techniques for text management. Moreover, other resources—such as thesauri that capture the relevant and characteristic terminology of a domain—and other techniques—such as latent semantic indexing to establish semantic relations without having to construct semantic networks—are equally important in large-scale text processing.

In the sections that follow we discuss some of the arguments and preliminary positive results that support our approach.

2.1 The **Utility of Phrase-Based Indexing**

Phrases approximate concepts more closely than words in isolation or in Boolean combination. Phrases based on natural-language structures express implicit semantic relations. Phrasal indexes to documents are less ambiguous and therefore easier for users to interpret than simple lists of 'words'.

Our results show that automatic phrase based indexing can equal (indeed, surpass) traditional human full-text indexing in consistency, completeness, and accuracy (Evans, et al. 1991b). In addition, automatic indexing is much less expensive than human indexing and has potential to scale up and support additional applications such as document profiling, sorting, classifying, and routing.

The essential facilities and resources to support phrase based indexing—parsers and grammars for selective NLP and first-order thesauri—are easy to develop and maintain.

2.2 The **Need for Thesauri**

Ideally a thesaurus is a treasury of terms that express the important concepts of a domain, identifying which terms are expressions of the same concept or similar concepts and indicating **along** which dimensions they are similar. It is clear that having a thesaurus just identifying the **terms** expressing the important concepts of a domain is key both to effective indexing and to document retrieval. Currently there is no well-established method or selection criterion for thesaurus construction. This is especially true in the case of large comprehensive thesauri for indexing **and** retrieval. The CLARIT project has been developing automatic techniques that seek to discover the important concepts of a domain by identifying terms both well-distributed in that domain and yet also involved in distinguishing particular aspects of it. We plan to extend this approach to build more structured thesauri that contain synonyms and minimal conceptual links.

Under the CLARIT approach, we distinguish among first- and second-order (or higher-order) thesauri. A *first-order thesaurus* consists of syntactically well-formed terms or noun phrases containing one or more morphologically normalized words. A set of certified terminology takes on the characteristics of a first-order thesaurus if all the terms in the set have a consistent decomposition

by lexical items. Thus, first-order thesauri can be regarded as having an implicit hierarchical organization by the lexical relations of *broader-than* and *narrower-than.* A *second-order thesaurus* goes beyond a first-order thesaurus in attempting to represent the semantic structure of concepts explicitly. Such semantic structure is captured via the semantic typing of terms, the identification of equivalence classes of terms, and the definition of explicit relations among terms. In general, second-order thesauri are difficult to construct and maintain. For automatic document processing, where one can expect to encounter numerous domains and sub-domains—and hence require numerous thesauri—there are clear advantages to being able to use first-order thesauri.

A major problem for a controlled vocabulary thesaurus is that it must be broad and deep enough to represent the content of a document adequately and concisely, but not so large that it ceases to be selective. There are currently no standard procedures for generating such collections of terminology, especially those that are responsive to many types of users and uses. The CLARIT Project is addressing this problem. We propose that natural-language processing on very large text corpora can be used to nominate terms and that a combination of techniques—including term clustering and term scoring based on distribution statistics—can be used to generate such vocabularies automatically.

CLARIT NP Clustering is a general method for empirically discovering how terms are distributed in texts by combining output of NLP and distribution heuristics to identify a subset of terms highly characteristic of arbitrary textual domains. Terms that share single or contiguous sets of words are grouped or clustered together. We believe this method is useful in identifying important domain concepts. In the following sections we give additional details of CLARIT theasurus construction techniques.

2.3 The Need to Identify Concept Equivalence Classes

Traditional word-based retrieval depends on finding a match between a word and a document. When using different words (even when they have more or less the same meaning) it has been shown that users will retrieve different documents thereby missing documents that may be important to them. Latent semantic indexing (LSI), for example, attempts to circumvent this problem by indexing documents based on secondary and tertiary associations of words recovering sematic relations that discriminate among alternative word meanings, as revealed by the co-occurence patterns of words in a document. The present CLARIT approach involves an adaptation of LSI, aimed at mapping users' expressions into appropriate sets of index terms.

As (Deerwester et al. 1990) points out, most people want to retrieve documents based on their meaning, not their exact language. The need for query expansion or elaboration is clear to anyone who has used a keyword-based retrieval system. The variety of potential query terms is not only intuitively obvious, but research has shown it to be more than perhaps commonly expected. Even well-known objects axe given the same name by different people only 20% of the time. (Cf. (Furnas et al. 1987).) Similar results—suggesting the persistence of variability—have been found for indexers and expert intermediaries, as well as less experienced searchers.

In addition to the synonymy problem, which affects recall, the same word may have different meanings (polysemy). This becomes important for larger document collections covering a variety of fields; a term may come into usage without its prior use being known. Even worse, for a variety of psychological reasons, people may choose or combine already known or partially known terminology (the alternative, creating new words, is not often done). Polysemy affects precision. Because the polysemy problem is more tractable when the field to which a document belongs can be identified, the CLARIT approach involves the creation of multiple thesauri representing all the fields and subfields of a given document collection, classifying a document according the thesaurus it 'matches' best.

Clearly, increased recall demands the use of some sort of synonym-based or higher-level the-

saurtis. In order to be designated as relevant, a document must match the query in some sense. In addition, some **sort** of recall is required to provide user feedback, and to prompt feedback from the user. Hand-constructed thesauri have a number of problems. In addition, automatically applied (not necessarily generated) thesauri tend to decrease precision in large databases (Sparck Jones 1972). Thus, our approach is to automatically construct, but not automatically apply, thesauri to the interpretation of queries.

3 Description of Underlying Modules and Capabilities

3.1 Lexicon

The CLARIT Lexicon for general English consists of approximately 100,000 root forms and hyphenated phrases, tagged for syntactic category and irregular morphological variation. The lexicon was developed from several non-proprietary sources and is wholly original with the project.

Because the lexicon and morphological processor together recognize on the order of 1,000,000 string forms of general English, it is possible to label any *unrecognized* lexical item as a *candidate proper noun.* This technique has proved extremely reliable and obviates the need to maintain a separate lexicon of proper names when parsing, e.g., news articles or documents with 'alpha-numeric' names, such as technical manuals.

3.2 Morphological Analyzer

The CLARIT Morphological Analyzer is capable of returning a set of legal root-forms and syntactic categories for every word or lexical phrase in a text, relatively quickly (approximately 1,000 words per second on a DECstation 3100) and virtually exhaustively for inflectional, derivational and phrasal lexical forms of English.

3.3 Lexical Disambiguator

The Lexical Disambiguator implements a stochastic grammar for English based on the Brown Corpus. (The project version of the Corpus is a refined, retagged set of sentences.) The disambiguator performs multi-valued lexical disambiguation using a time linear algorithm. It analyzes possible lexical tags based on simple word frequency and pattern frequency statistics.

3.4 Multi-Stage Parser

The Multi-Stage Parser implements a context-free grammar focused on English NP constructions. The parser employs a modified left-corner chart parsing algorithm that uses several techniques to recognize and ignore illegitimate parses as early as possible, thereby improving parser performance in the expected case. Also, the algorithm has been modified to make it more robust. In situations where the parser is unable to identify a full cover for a given input, it will attempt to return as many useful pieces as it can identify. The parse trees generated by the parser are then mapped into frame-like representations that allow the system to work with the gross structure of the noun phrase, rather than being forced to work with full parse trees. The frames represent a noun phrase as composed of determiner, modifier, and head structures. It is possible to reduce the number of ambiguous parses by combining separate frames which do not differentiate two unique overall parse structures. For CLARIT processing purposes, the modifier and head portions of the noun phrase frames are displayed in a simple list format, resulting in a data structure that is useful as a possible index term.

The multi-stage parser is capable of processing approximately 50 words per second in the current Common Lisp implementation (running on a DECstation 5000 workstation). Parser output

identifies modifier head combinations that are possible index terms, and almost always (at least in 99% of parses) manages to constrain possible ambiguities enough to produce only one parse per noun phrase.

3.5 Noun Phrase Grammar

The CLARIT Noun Phrases Grammar was developed originally based on a corpus of over 700 NP types. In developing the CFG rules for the corpus, of course, the grammar has been extended to over an 'infinite' set of NP types. The grammar is designed to identify constituents in their roles as heads and modifiers. While a 'complex-NP' version of the grammar exists, our experience has shown the utility of processing with a 'simplex-NP' grammar—including all the NP structure up to and including the head, but no complement structure.

Because the information content of an NP may not always be found in its literal syntactic head—e.g., *many* in *many of the soldiers* and *development* in *the development of writing skills in first-graders*—the CLARIT grammar has been developed to provide 'information-theoretically useful' parses. In particular, the grammar provides for classes of *pre-determiner* quantifiers (such as *many of)* and for *de-verbal nominals* (such as *development)* and specifically demotes them from 'head' positions. (In the above examples, *soldiers* and *writing skills,* respectively, would be identified as heads.)

The grammar also treats as special cases a variety of participle modifiers and noun-noun compounds; conjunctions; and modifier and complement attachment. Because the process is designed to extract terms in their coarse roles as heads and modifiers of NPs, the possibly ambiguous substructure of NPs can often be ignored, greatly reducing variant parses.

3.6 Indexing Algorithms

The Project has developed and implemented varieties of algorithms for scoring NPs as index terms; for building index-term lattices; for finding subsets of terminology to use as first-order thesauri; for computing latent semantic spaces for lexical items. We describe several of the most important algorithms below.

3.6.1 Ranking Index Terms

Term ranking is accomplished through the interaction of two modules: SCORE and MATCH. SCORE evaluates the noun phrases discovered in an article based on statistics extracted from a large sample of text representative of a given domain. It measures terms using statistical parameters of frequency, distribution, and linguistic distinctiveness. Each possible index term is evaluated in comparison to its expected occurrence within the chosen domain. The MATCH module identifies good index terms based on a list of certified concepts for the domain (a first-order thesaurus). It eliminates some noun phrases from the list of index terms and produces a categorized listing of terms. The new listing contains some terms extracted from the thesaurus that are either present in the article or are generalization of terms present in the article. It also contains terms that are not directly related to the thesaurus, but which may be still be useful for indexing. The combined effect of the SCORE and MATCH modules is to produce a ranked listing of index terms that are organized according to their relationships to certified domain terminology.

3.6.2 Index Term Clustering

Index term lattices are constructed using the CLUSTER module. CLUSTER decomposes a set of terms into all possible subsequences, and identifies interesting subsequences based upon their occurrence in the input set. So, for example, the terms "red fire truck," "green fire truck," and

"blue fire truck," all share the sequence "fire truck." Therefore, these terms may be referred to as being in the "fire truck" cluster. Note that the cluster "truck" may also be identified, and that these two clusters form the beginnings of a lattice, since the cluster "truck" contains all of the cluster "fire truck." CLUSTER incorporates several techniques for eliminating useless clusters from this lattice, based strictly on statistical information about the occurrences of subsequences. The resulting lattice contains a large amount of information about word occurrence and patterning characteristics in a given set of terms. Therefore, it is used within other algorithms (such as thesaurus extraction) and is also useful for browsing purposes.

3.7 Automatic Term Clustering and Thesaurus Discovery

For the last several years the CLARIT team has been producing first-order thesauri using a variety of computational techniques, including selected NLP, automated clustering of noun phrases or terms, and automated thesaurus discovery, all developed by the CLARIT team. The automated clustering of terms extracted using selected NLP has not only been used in thesaurus discovery, but also to produce results found useful in themselves.

A first-order thesaurus should be a collection of the useful and interesting terms in a domain. Intuitively, therefore, that set of terms should have the following characteristics:

- Each term in the thesaurus should be syntactically well-formed. It should be a term that is 'recognizable' language and can be matched against other terms recognized by the parser.
- The terms in the thesaurus should serve to identify the distinctive concepts in the domain. Terms such as "very large piece"—because of their 'generality'—should be eliminated; terms such as "latent semantic indexing" should be selected.
- Each term in the thesaurus should be relevant to the domain. Terms extracted from text describing an example are not necessarily relevant to the domain. Also, terms that are not shared among different articles in the domain should not be nominated as concepts.

In order to locate a set of terms having all the desirable characteristics, we have developed a multi-phase procedure which takes as input a large amount of text (preferably as much as 10 megabytes) from the domain. The first step of the process is to parse the text to extract all the candidate noun phrases. This provides a set of syntactically well-formed phrases from which a subset of interesting terms can be chosen. The second phase involves the evaluation of the noun phrases using several different criteria that are designed to measure the distinctiveness and relevancy of possible terms for inclusion in a thesaurus. Finally, a subset of the noun phrases is selected by identifying the set of terms that optimizes the evaluation criteria.

To achieve completely automatic thesaurus discovery using on-line corpora, we have experimented with a number of techniques. One technique, based almost entirely on clustering, involves selecting the shared elements of clusters (called the 'heads' or 'names' of clusters) just in case they satisfy two constraints: (1) The shared elements must also appear as independent noun phrases (minimally, a single word noun phrase); and (2) the cluster subsumes terms that occur above a specified frequency in the corpus. This technique has worked well.

We are currently experimenting with new techniques. We continue to exploit information based on clustering, but only indirectly, to determine a term's relative distribution in a corpus. Other conditions such as frequency in the corpus and rarity in a general corpus of English (e.g., the **Brown** Corpus) are also taken into account in the selection of a term. Such statistics provide a metric of a term's distribution in a corpus: We aim to select terms that are well distributed, but nevertheless useful in distinguishing one document or one portion of text from another.

3.7.1 Ranking the Noun Phrases

We have developed four statistical parameters, each of which measures a different facet of the composition, distribution, and frequency characteristics of each noun phrase. These parameters can be divided into two basic classes: first, the statistics that evaluate the term as a whole; second, the statistics that evaluate components of the term. ("Components" can include individual words or sequences of words from the noun phrase.) The two statistics used in evaluating a candidate term are:

Term Distribution: The distribution of a term is simply a count of the number of articles in which the term appears. A term that is highly distributed in a domain will represent a concept that is broadly used.

Frequency: The frequency of a term is a count of the total number of times that a term is used in the sample domain. A frequently used noun phrase is necessary to expressing ideas in the domain.

Both frequency and term distribution measure the relevance of a given noun phrase to the sample text. However, without additional information, such measures are not sufficient to evaluate the importance of a noun phrase as a thesaurus concept. We employ the two additional measures of "cluster distribution" and "rarity" to analyze noun phrases based on their compositional structure. Thus, these measures are designed to capture the characteristics of the language that forms the noun phrase.

3.7.2 Cluster Distribution.

The *cluster distribution* statistic measures the distribution of subsequences of words within the noun phrase. Since all the noun phrases have been "clustered", it is possible to identify useful combinations of words that are shared between several different noun phrases. For purposes of thesaurus construction, clustering can be thought of as a tool for identifying all of the significant combinations of words that are shared between different noun phrases.

3.7.3 Rarity

Rarity is a statistic based on an analysis of word frequency in common English. Each word is given a frequency count based on its occurrence in the Brown Corpus of Kučera and Francis. The score defaults to 1 if the word is not present in the corpus. Every word receives a rarity score by the function Median Frequency/Frequency Count. In the Brown Corpus, the median frequency is 2. This score, which is calculated for an individual word, is then summed over every word in a noun phrase, to give a total score for a noun phrase. The rarity statistic of a noun phrase will give high scores to long noun phrases that contain distinctive and unusual words.

Intuitively, the combination of all of these parameters should identify noun phrases that (1) are constructed from interesting words (rare), (2) contain useful combinations of words (cluster distribution), (3) are frequent in the domain, and (4) are well-distributed across the domain.

Additional documentation and examples can be found in (Evans et al. 1991d).

3.8 Latent-Semantic Space Generation

Our interest in LSI is based upon rationale and results given in (Deerwester et al. 1990), which are in turn based on the vector-space retrieval model of Salton. LSI addresses the problem of variability of query language by using language from the document set to automatically build a form of (synonym) thesaurus, based upon word co-occurrence in the document set. The method follows

naturally as a compression/decomposition of a term-document matrix, which can theoretically be represented in terms of a smaller number of 'meaning dimensions': the latent semantics. Although the method handles synonymy fairly well, polysemy remains a problem; we intend to handle this with more precise identification of terms and by dividing document sets into subdomains to limit language variability and therefore polysemy. When unified with a retrieval system, the method should be able to make use of both positive information to increase recall (in the form of new query language and relevant documents) and negative information to increase precision (in the form of irrelevant documents).

In addition, the method can be used to associate query language (words and terms) with indexing language (terms), as well as query language directly with relevant documents.

However, LSI has several weaknesses that make it unsuitable for unrestricted IR applications. Most critically, LSI is computationally expensive (hence impractical for use with large document collections) and the value of LSI in discovering 'semantics' is weakened to the extent that polysemous words in any collection will lead to bad results of processing.

The CLARIT approach to LSI seeks to remedy these weaknesses. Instead of relating words and documents, the CLARIT approach involves relating words (or language, more generally) to terms. This becomes the basis for incrementally supplementing and refining a thesaurus' collection of synonyms in a given domain. For example, LSI can be used to derive the common semantic space of terms from different vocabularies. Each term is treated as a 'document' containing words importantly related to the term. An LSI space not only will cluster the terms according to their implicit semantics, but can also be used to map any natural-language variants of the terms to the set of 'best-matching' terms in the space. In effect, the variant expressions are taken as 'queries' and the retireved terms are the 'documents' that best match the query.

The results of preliminary experiments in exploiting 'latent semantic space' models of terms are given in (Evans et al. 1991a; 1991c) and (Chute, Yang, Evans 1991). Additional information about LSI and the associated method of singular-value decomposition can be found in (Deerwester et al. 1990), (Cullum et al. 1983), (Cullum & Willoughby 1985), (Forsythe et al. 1977), and (Golub & Reinsch 1971).

4 Goals and Future Directions

We envision our efforts as laying the foundation for two information systems that could be used in arbitrary work environments of groups engaged in technical activities such as engineering. One of the systems would significantly extend traditional IR *performance* and hone it for use in group-work environments, providing, for example, automated indexing, profiling, and retrieval over large corpora either shared by many teams or indigenous (proprietary) to a single team. The other system (which could be integrated with the first) would perform tasks not normally regarded as part of the repertoire of traditional IR systems, such as message routing, sorting, and classifying for specific domains of message traffic (e.g., notes, memos, reports associated with specific tasks). These capabilities are different from those of traditional IR systems in that the former systems are *provided* with the data to be processed whereas the latter systems must solve the problem of *capturing* the data to be processed. While it is true that IR systems currently depend on scanning printed documents on-line, the capability needed to realize the second product is the capture of transient or uncollected data already on-line.

An essential requirement is that capturing of information must keep up with the pace at which messages are being received. This means, from a CLARIT perspective, that they must be parsed and indexed in virually real time. Moreover, thesaurus updating (for use in the indexing process) must also occur quickly. Although CLARIT processing proceeds reasonably fast at present, it will have to execute much more quickly in the future. Within a year, we plan to be able to index 100 megabytes of text per day; in three years, a gigabyte per day.

Not only must CLARIT processing be fast, it must also extend beyond literal matching and be capable of discovering synonyms or at least to remove the strict dependency on literal matching. We measure this in terms of achieving significantly better results in the tradeoff between precision and recall than is currently being achieved using free text retrieval systems.

Another requirement is that CLARIT modules and tools be transportable across technical domains and groups. Our goal in this regard is that the necessary preprocessing of documents, updates to the lexicon, the construction of a domain thesaurus, parsing, and the indexing of a relatively large corpus of text would occur within a day, so that in a twenty-four-hour period, automatic retrieval, routing, sorting, classifying could be implemented for any given technical domain. Within a month, we plan to be able to extract a thesaurus from a minimal corpus of 10 megabytes in two-hours time. Since larger corpora result in better thesauri, we estimate processing a 100 megabyte corpus in one day. Within two years, we expect to be able to perform the same processes in minutes and hours, respectively.

Not only must CLARIT processing be fast, less literal, and transportable, it must also be packaged in an interactive interface that is easy to use, both for building the *resources* needed for the two information systems and for using them as well. Our goal in this regard is that a maintainer who knows how to use a text processor would be able to learn how to build CLARIT resources within a day; and that a user of a CLARIT-based application would be able to use it at least as quickly, but more effectively, than existing retrieval systems.

4.1 **Stochastic Noun** Phrase **Recognizer**

The techniques employed by the stochastic lexical disambiguator are general enough to be expanded beyond the simple task of lexical disambiguation. In fact, we plan to use a very similar algorithm to solve the problem of noun phrase recognition. Since a recognition problem can be thought of as a problem of disambiguating between tokens in the structure and tokens outside of the structure, the simple pattern matching operations of the disambiguator can still be employed. If successful, the recognizer would be able to analyze arbitrary texts in strictly linear time. While the current parser written in Common Lisp operates at a speed of 50 words per second, it is expected that the new stochastic recognizer could operate at speeds of 1,000-10,000 words per second. However, such an application would still require a large corpus of text from which to extract statistics about noun phrase occurrences in a given domain. Therefore, a full parser, with a more traditional grammar, will be used with unfamiliar domains. Once a large enough amount of text in a given domain has been analyzed, the stochastic recognizer can be used to process all further texts.

4.2 Hybrid Parser

We are exploring a new design for the parsing module of CLARIT that would realize many improvements over the current system. The new parser employs several different methods of analysis during each phase of the parsing operation. Structures can be identified using stochastic grammars, finite-state grammars, and probabilistic context-free grammars. Furthermore, all levels of description are used productively to constrain the number of ambiguities generated by the system. Thus, for example, the stochastic disambiguation phase of the algorithm does not actually function in situations where no ambiguity exists. This design takes advantage of recent developments in the use of stochastic parsers, probabilistic grammars, and parsing control structures. We expect the new implementation to improve the accuracy and speed of the parser over unrestricted texts, while also making it easier to write and maintain grammars. The actual speed of the parser will depend completely on the type of analysis attempted. For example, strict stochastic disambiguation as described above, should operate at very high speeds. However, even full-scale context-free parsing should show significant improvement over current technology. We hope to achieve speeds of 1000-2000 words per second for full-scale parsing, which is twenty-forty times faster than the current

parser. In fact, a prototype *heuristic* parser now operating in the CLARIT system can process at rates of 2 megabytes of text per minute.

4.3 Thesaurus Extraction Techniques

There are several developmental thrusts that will improve the performance of our thesaurus extraction mechanisms. The complete thesaurus discovery package will have the following facilities:

1. Fast and reliable identification of certified term lists from a large sample of domain specific text (extraction of a first-order thesaurus)
2. Capabilities for incremental updates of the thesaurus, based on new textual information—Such a facility will be extremely useful in contexts where the subject database is constantly changing.
3. Identification of equivalence classes of terms strictly through analysis of the corpus; discovery of relations using various computation techniques including LSI and cluster analysis
4. Semi-automatic and interactive discovery of hierarchical and minimal conceptual links among terms

As recently as one month ago, it took a full day of processing to extract a first-order thesaurus from a 5 megabyte sample corpus. In our deliverable system, this should be fast enough to extract a thesaurus in one hour from a 10 megabyte corpus. Furthermore, incremental updates to a thesaurus should occur in time intervals of under five minutes.

4.4 Automated Indexing Tools

In the future, the automated indexing tools will be re-engineered, and re-implemented in C, in order to increase their efficiency and robustness. Along with re-engineering, the functionality of the algorithms will be modified in order to allow the new modules to interact easily with the proposed information management and filtering systems. Furthermore, the various operational parameters of the system will be made explicit, so that it be much easier to fine-tune indexing for a specific domain.

For indexing, the information management tool will support not only interactive indexing of documents, but also the construction of databases for automatic indexing in the filtering system. This is the larger task, since it includes document preprocessing, parsing, database construction and update, language and domain analysis for indexing terms, as well as indexing itself. Although retrieval also includes some of the same functions, the system need only perform on fixed data.

Given a document or document set using a potentially new markup language, the indexing tool will assist the user in discovering and initially filtering that markup language, identifying new words or morphemes, and assisting in any required lexicon update. It will act as an agent for parsing the text and aid the user in reviewing the results, identifying any problems, and in correcting them efficiently. It will allow the user to select groups of documents and to segment them flexibly, either based on the markup language, hand selections, or perhaps database operations involving the segment's content in conjunction with synonym databases. It will assist the user in creating domain definitions and thesauri based upon such groupings and assist in matching documents against them. The construction of Latent Semantic Concept spaces will be driven by similar selections. The user will be able to determine how well a given set of indexing terminology covers a particular group of documents and perhaps make use of this information to drive further analysis and first-level thesaurus updating.

Retrieval in the information management tool will include not only traditional string-based or at least word- and phrase-based retrieval, possibly in conjunction with a synonym database, but

also various applications of Latent Semantic Concept indexing. Queries of a given document set will be parsed and matched to indexing terminology as well as directly to documents, and perhaps to personal or collective histories of queries and relevant indexing terminology and documents. Feedback can be obtained and given in the form of relevant or irrelevant terminology, documents, or former queries. Some of the analysis tools used in thesaurus discovery may also be useful in retrieval to summarize documents and to characterize their similarity or dissimilarity.

The filtering system will be the culmination of the project research. It will be virtually automatic as well as accurate. It may be used in conjunction with the interactive retrieval system, to allow a natural form of review and profile updating, but it should function (once trained) with minimal updates. As such, it will have to make use of prior knowledge built up over time. It will distinguish high-priority document types (where not missing a message is important) from low-priority document types (such as mailing lists) which can be interactively perused from time to time.

5 Selected References

[Chute, Yang, Evans 1991]: Chute, C.G., Yang, Y., and Evans, D.A. Latent semantic indexing of medical diagnoses using UMLS semantic structures. P.D. Clayton (Editor), *Fifteenth Annual Symposium on Computer Applications in Medical Care (SCAMC),* Washington, DC: IEEE Computer Society, 1991, 185-189.

[Cullum et al. 1983]: Cullum J.K., Willoughby R.A., and Lake, M. A Lanczos algorithm for computing singular values and vectors of large matrices. *SIAM Journal of Scientific and Statistical Computing, 4(2),* 1983, 197-215.

[Cullum & Willoughby 1985]: Cullum, J.K. and Willoughby, R.A. *Lanczos algorithms for large symmetric eigenvalue computations, Vol. I Theory.* Boston, MA: Birkhauser, 1985.

[Deerwester et al. 1990]: Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., and Harshman, R. Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41(6),* 1990, 391-407.

[Evans 1990]: Evans, D.A. Concept management in text via natural-language processing: The CLARIT approach. *Working Notes of the 1990 AAAI Symposium on "Text-Based Intelligent Systems'[9],* Stanford University, March, 27-29, 1990, 93-95.

[Evans et al. 1991a]: Evans, D.A., Chute, C.G., Handerson, S.K., Yang, Y., Monarch, I.A., and Hersh, W.R. 'Latent semantics' as a basis for managing variation in medical terminologies. Submitted to *Medinfo 92.*

[Evans et al. 1991b]: Evans, D.A., Ginther-Webster, K., Hart, M., Lefferts, R.G., and Monarch, LA. Automatic indexing using selective NLP and first-order thesauri. *RIAO '91,* Autonoma University of Barcelona, Barcelona, Spain, April 2-5, 1991, 624-644.

[Evans et al. 1991c]: Evans, D.A., Handerson, S.K., Monarch, LA., Pereiro, J., Hersh, W.R. *Mapping Vocabularies using 'Latent Semantics*.* Technical Report No. CMU-LCL-91-1, Laboratory for Computational Linguistics, Carnegie Mellon University, Pittsburgh, PA, 1991, 16pp.

[Evans et al. 1991d]: Evans, D.A., Hersh, W.R., Monarch, LA., Lefferts, R.G., Handerson, S.K. Automatic indexing of abstracts via natural-language processing using a simple thesaurus. *Medical Decision Making, 11,* (Supplement), 1991, S108-S115.

[Forsythe et al. 1977]: Forsythe, G.E., Malcolm, M.A., and Moler, C.B. *Computer Methods for Mathematical Computations.* Englewood Cliffs, N.J: Prentice-Hall, 1977.

[Furnas et al. 1987]: Furnas, G.W., Landauer, T.K., Gomez, L.M., and Dumais, S.T. The vocabulary problem in human-system communication. *Communications of the ACM, 30(11),* 1987, 964-971.

[Golub & Reinsch 1971]: Golub, G.H., and Reinsch, C. Singular value decomposition and least squares solutions. In: Wilkinson, J., and Reinsch, C. *Linear Algebra* New York, NY: Springer-Verlag, 1971.

[Hersh et al. 1991]: Hersh, W.R., Evans, D.A., Monarch, LA., Lefferts, R.G., Handerson, S.K., Gorman, P.N. Indexing effectiveness of linguistic and non-linguistic approaches to automated indexing. Submitted to *Medinfo 92.*

[Sparck Jones 1972]: Sparck Jones, K. A statistical interpretation of term specificity and its applications in retrieval. *Journal of Documentation,* March 1972, *28(1),* 11-21.