Chapter 7

Document Processing

7.1 Overview

Per-Kristian Halvorsen

Xerox-PARC, Palo Alto, California, USA

7.1.1 The Document

Work gets done through documents. When a negotiation draws to a close, a document is drawn up, an accord, a law, a contract, an agreement. When a new organization is established it is announced with a document. When research culminates, a document is created and published. And knowledge is transmitted through documents: research journals, text books and newspapers. Documents are information organized and presented for human understanding. Documents are where information meets with people and their work. By bringing technology to the process of producing and using documents one has the opportunity to achieve significant productivity enhancements. This point is important in view of the fact that the derivation of productivity increases and economic value from technological innovation in information technologies has proven difficult. In the past decade we have seen unsurpassed innovation in the area of information technology and in its deployment in the general office. Provable increases in the effectiveness of work have been much harder to come by (David, 1991; Brynjolfsson, 1993). By focusing on the work practices that surround the use of documents we bring technology to bear on the pressure points for efficiency. While the prototypical document of the present may be printed, the document is a technology with millennia of technological change behind it. An important change vector for the document concerns new types of content (speech and video in addition to text and pictures) and non-linear documents (hyper-media). Of equal importance is the array of new technologies for processing, analyzing and interpreting the content, in particular the natural language content, of the document. Language, whether spoken or written, provides the bulk of the information-carrying capacity of most work-oriented documents. The introduction of multi-media documents only extends the challenge for language technologies: analysis of spoken as well as written language will enhance the ability to navigate and retrieve multi-media documents.

7.1.2 Document Work Practices

The utility of information technology is amplified when its application reaches outside its native domain—the domain of the computer—and into the domain of everyday life. Files are the faint reflections in the domain of the computer of documents in the domain of everyday life. While files are created, deleted, renamed, backed up, and archived, our involvement with documents forms a much thicker fabric: Documents are read, understood, translated, plagiarized, forged, hated, loved and emasculated. The major phases of a document's life cycle are creation, storing, rendering (e.g., printing or other forms of presentation), distribution, acquisition, and retrieving (Figure 7.1). Each of

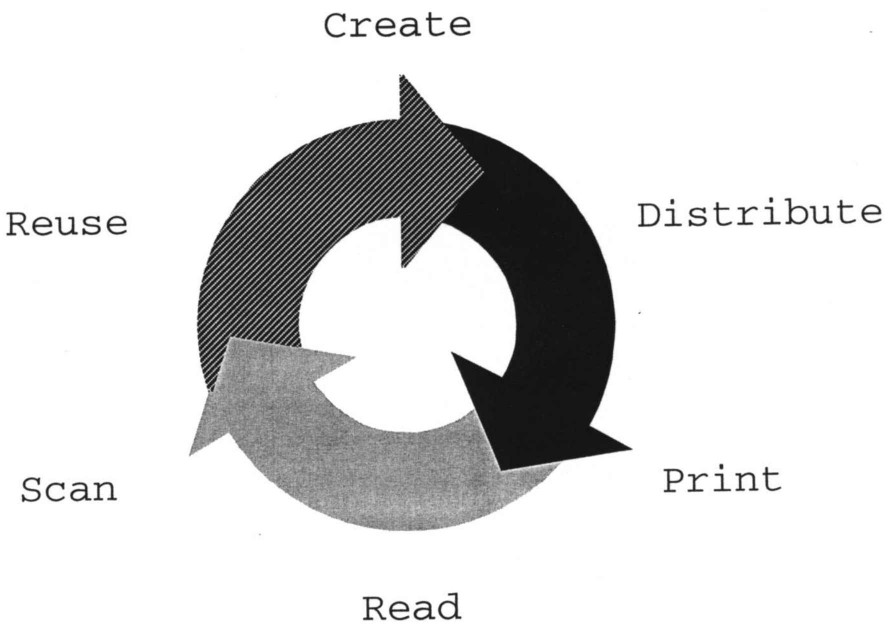

Figure 7.1: The life cycle of a document.

these phases is now fundamentally supported by digital technology: Word processors and publishing systems (for the professional publisher as well as for the desktop user) facilitate the creation phase, as do multi-media production environments.

Document (text) databases provide storage for the documents. Rendering is made more efficient through software for the conversion of documents to page description languages (PDLs) and so-called imagers which take PDL representations to a printable or projectable image. Distribution takes place through fax, networked and on-demand printing, electronic data interchange (EDI) and electronic mail. Acquisition of documents in print form for the purpose of integration into the electronic domain takes place through the use of scanners, image processing software, optical character recognition (OCR) and document recognition or reconstruction. Access is accomplished through document databases. Natural language technologies can yield further improvements in these processes when combined with the fundamental technologies in each phase to facilitate the work that is to be done.

Creation: Authoring aids put computing to the task of assisting in the preparation of the content and linguistic expression in a document in the same way that word processors assist in giving the document form. This area holds tremendous potential. Even the most basic authoring aid—spelling checking—is far from ubiquitous in 1994: The capability and its utility has been proven in the context of English language applications, but the deployment in product settings for other languages is just beginning, and much descriptive linguistic work remains to be done. Grammar and style checking, while unproven with respect to their productivity enhancement, carry significant attraction as an obvious extension to spelling checking. The dependence on challenging linguistic descriptive work is even more compelling for this capability than for the spelling checking task. Authoring tools do not exhaust the range of language-based technologies which can help in the document creation process. Document creation is to a large extent document reuse. The information in one document often provides the basis for the formulation of another, whether through translation, excerpting, summarizing, or other forms of content-oriented transformation (as in the preparation of new legal contracts). Thus, what is often thought of as access technologies can play an important role in the creation phase.

Storage: Space, speed and ease of access are the most important parameters for document storage technologies. Linguistically based compression techniques (e.g., token-based encoding) can result in dramatically reduced space requirements in specialized application settings. Summarization techniques can come into play at the time of storage (filing) to prepare for easier access through the generation of compact but meaningful representatives of the documents. This is not a fail-safe arena for deployment, and robustness of the technology is essential for success in this application domain.

Distribution: With the geometric increase in electronically available information, the demand for automatic filtering and routing techniques has become universal. Current e-mail and work group support systems have rudimentary capabilities for filtering and routing. The document understanding and information extraction technologies described in this chapter could provide dramatic improvements on these functions by identifying significant elements in the content of the document available for the use of computational filtering and routing agents.

Acquisition: The difficulty of integrating the world of paper documents into the world of electronic document management is a proven productivity sink. The role of natural language models in improving optical character recognition and document reconstruction is highly underexploited and just now being reflected in commercial products.

Access: An organization's cost for accessing a document far dominates the cost of filing it in the first place. The integration of work flow systems with content-based document access systems promises to expand one of the fastest growing segments of the enterprise level software market (work flow) from the niche of highly structured and transaction oriented organizations (e.g., insurance claim processing), to the general office which traffics in *free text* documents, and not just forms. The access phase is a ripe area for the productivity enhancing injection of language processing technology. Access is a fail-soft area in that improvements are cumulative and 100% accuracy of the language analysis is not a prerequisite for measurably improved access. Multiple technologies (e.g., traditional retrieval techniques, summarization, information extraction) can be synergetically deployed to facilitate access.

7.2 Document Retrieval

Donna Harman,[a] Peter Schäuble,[b] & Alan Smeaton[c]

[a] NIST, Gaithersburg, Maryland, USA
[b] ETH Zurich, Switzerland
[c] Dublin City University, Ireland, UK

Document retrieval is defined as the matching of some stated user query against useful parts of free-text records. These records could be any type of mainly unstructured text, such as bibliographic records, newspaper articles, or paragraphs in a manual. User queries could range from multi-sentence full descriptions of an information need to a few words and the vast majority of retrieval systems currently in use range from simple Boolean systems through to systems using statistical or natural language processing. Figure 7.2 illustrates the manner in which documents are retrieved from various sources.

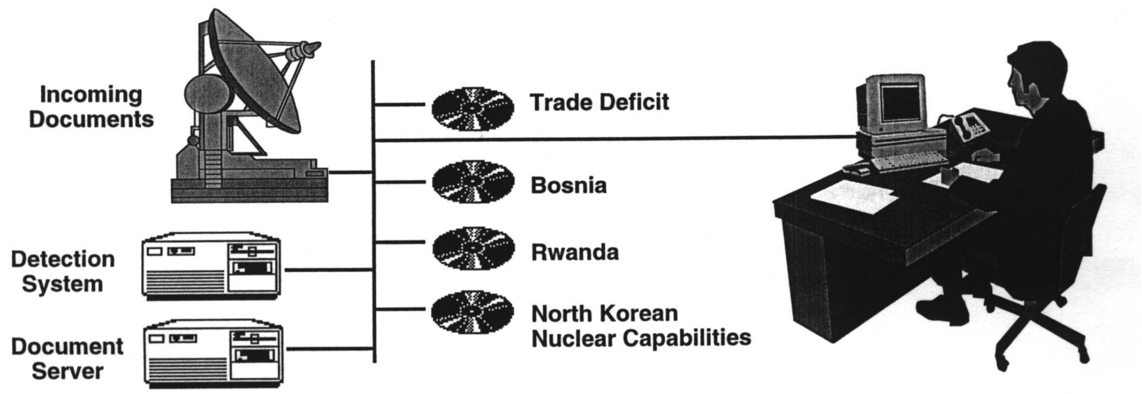

Figure 7.2: The document retrieval process.

Several events have recently occurred that are having a major effect on research in this area. First, computer hardware is more capable of running sophisticated search algorithms against massive amounts of data, with acceptable response times. Second, INTERNET access, such as World Wide Web (WWW), brings new search requirements from untrained users who demand user-friendly, effective text searching systems. These two events have contributed to create an interest in accelerating research to produce more effective search methodologies, including more use of natural language processing techniques.

There has been considerable research in the area of document retrieval for over 30 years (Belkin & Croft, 1987), dominated by the use of statistical methods to automatically match natural language user queries against records. For almost as long there has been interest in using natural language processing to enhance single term matching by adding phrases (Fagan, 1989), yet to date natural language processing techniques have not significantly improved performance of document retrieval, although much effort has been expended in various attempts. The motivation and drive for using natural language processing (NLP) in document retrieval is mostly intuitive; users decide on the relevance of documents by reading and analyzing them and if we can automate document analysis this should help in the process of deciding on document relevance.

Some of the research into document retrieval has taken place in the ARPA-sponsored TIPSTER project. One of the TIPSTER groups, the University of Massachusetts at Amherst, experimented with expansion of their state-of-the-art INQUERY retrieval system so that it was able to handle the 3–gigabyte test collection. This included research in the use of query structures, document structures, and extensive experimentation in the use of phrases (Broglio, Callan, et al., 1993). These phrases (usually noun phrases) were found using a part-of-speech tagger and were used either to improve query performance or to expand the query. In general, the use of phrases as opposed to the use of single terms for retrieval did not significantly improve performance, although the use of noun phrases to expand a query shows much more promise. This group has found phrases to be useful in retrieval for smaller collections, or for collections in a narrow domain.

A second TIPSTER group using natural language processing techniques was Syracuse University. A new system, the DR-LINK system, based on automatically finding conceptual structures for both documents and queries, was developed using extensive natural language processing techniques such as document structure discovery, discourse analysis, subject classification, and complex nominal encapsulation. This very complex system was barely finished by the end of phase I (Liddy & Myaeng, 1993), but represents the most complex natural language processing system ever developed for .

The TIPSTER project has progressed to a second phase that will involve even more collaboration between NLP researchers and experts. The plan is to develop an architecture that will allow standardized communication between document retrieval modules (usually statistically based) and natural language processing modules (usually linguistically based). The architecture will then be used to build several projects that require the use of both types of techniques. In addition to this theme, the TIPSTER phase II project will investigate more thoroughly the specific contributions of natural language processing to enhanced retrieval performance. Two different groups, the University of Massachusetts at Amherst group combined with a natural language group at BBN Inc., and a group from New York University will perform many experiments that are likely to uncover more evidence as to the usefulness of natural language processing in document retrieval.

The same collection used for testing in the TIPSTER project has been used by a much larger worldwide community of researchers in the series of Text REtrieval Conference (TREC) evaluation tasks. Research groups representing very diverse approaches to document retrieval have taken part in this annual event and many have used NLP resources like lexicons, dictionaries, thesauri, proper name recognizers and databases, etc. One of these groups, New York University, investigated the gains for using more intensive natural language processing on top of a traditional statistical retrieval system (Strzalkowski, Carballo, et al., 1995). This group did a complete parse of the 2-Gbyte texts to locate content-carrying terms, discover relationships between these terms, and then use these terms to expand or modify the queries. This entire process is completely automatic, and major effort has been put into the efficiency of the natural language processing part of the system. A second group using natural language processing was the group from General Electric Research and Development Center (Jacobs, 1994). They used natural language processing techniques to extract information from (mostly) the training texts. This information was then used to create manual filters for the routing task part of TREC. Another group using natural language processing techniques in TREC was CLARITECH (Evans & Lefferts, 1994). This group used only noun phrases for retrieval and built dynamic thesauri for query expansion for each topic using noun phrases found in highly ranked documents. A group from Dublin City University derived tree structures from texts based on syntactic analysis and incorporated syntactic ambiguities into the trees (Smeaton, O'Donnell, et al., 1995). In this case document retrieval used a tree-matching algorithm to rank documents. Finally, a group from Siemens used the WordNet lexical database as a basis for query expansion (Voorhees, Gupta, et al., 1995) with mixed results.

The situation in the U.S. as outlined above is very similar to the situation in Europe. The European Commission's Linguistic Research and Engineering (LRE) sub-programme funds projects like CRISTAL which is developing a multilingual interface to a database of French newspaper stories using NLP techniques and RENOS which is doing similar work in the legal domain. The EC-funded SIMPR project also used morpho-syntactic analysis to identify indexing phrases for text. Other European work using NLP is reported in Hess (1992); Ruge (1992); Schwarz and Thurmair (1986); Chiaramella and Nie (1990) and is summarized in Smeaton (1992).

Most researchers in the information retrieval community believe that retrieval effectiveness is easier to improve by means of statistical methods than by NLP-based approaches and this is borne out by results, although there are exceptions. The fact that only a fraction of information retrieval research is based on extensive natural language processing techniques indicates that NLP techniques do not dominate the current thrust of information retrieval research as does something like the Vector Space Model. Yet NLP resources used in extracting information from text as describes by Paul Jacobs in section 7.3, resources like thesauri, lexicons, dictionaries, proper name databases, are used regularly in information retrieval research. It seems therefore that NLP *resources* rather than NLP techniques are having more of an impact on document retrieval effectiveness at present. Part of the reason for this is that natural language processing techniques are generally not designed to handle large amounts of text from many different domains. This is reminiscent of the situation with respect to information extraction which likewise is not currently successful in broad domains. But information retrieval systems do need to work on broad domains in order to be useful and the way NLP techniques are being used in information retrieval research is to attempt to integrate them with the dominant statistically-based approaches, almost piggy-backing them together. There is, however, an inherent granularity mismatch between the statistical techniques used in information retrieval and the linguistic techniques used in natural language processing. The statistical techniques attempt to match the rough statistical approximation of a record to a query. Further refinement of this process using fine-grained natural language processing techniques often adds only noise to the matching process, or fails because of the vagaries of language use. The proper integration of these two techniques is very difficult and may be years in coming. What is needed is the development of NLP techniques specifically for document retrieval and vice versa the development of document retrieval techniques specifically for taking advantage of NLP techniques.

Future Directions

The recommendations for further research are therefore to continue to pursue this integration, but with more attention to how to adapt the output of current natural language methods to improving information retrieval techniques. Additionally natural language processing techniques could be used directly to produce tools for information retrieval, such as creating knowledge bases or simple thesauri using data mining.

7.3 Text Interpretation: Extracting Information

Paul Jacobs

SRA International, Arlington, Virginia, USA

The proliferation of on-line text motivates most current work in text interpretation. Although massive volumes of information are available at low cost in free text form, people cannot read and digest this information any faster than before; in fact, for the most part they can digest even less. Often, being able to make efficient use of information from text requires that the information be put in some sort of structured format, for example, in a relational database, or systematically indexed and linked. Currently, extracting the information required for a useful database or index is usually an expensive manual process; hence on-line text creates a need for automatic text processing methods to extract the information automatically (Figure 7.3).

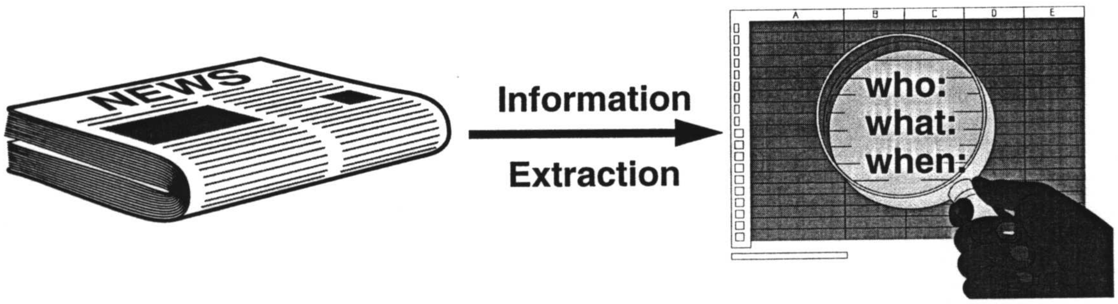

Figure 7.3: The problem of information extraction from text.

Current methods and systems can digest and analyze significant volumes of text at rates of a few thousand words per minute. Using *text skimming*, often driven by finite-state recognizers (discussed in chapters 3 and 11 of this volume), current methods generally start by identifying key artifacts in the text, such as proper names, dates, times, and locations, and then use a combination of linguistic constraints and domain knowledge to identify the important content of each relevant text. For example, in news stories about joint ventures, a system can usually identify joint venture partners by locating names of companies, finding linguistic relations between company names and words that describe business tie-ups, and using certain domain knowledge, such as understanding that ventures generally involve at least two partners and result in the formation of a new company. Other applications are illustrated in Ciravegna, Campia, et al. (1992); Mellish et al. (1995). Although there has been independent work in this area and there are a number of systems in commercial use, much of the recent progress in this area has come from U.S. government-sponsored programs and evaluation conferences, including the TIPSTER Text Program and the MUC and TREC evaluations described in chapter 13. In information extraction from text, the TIPSTER program, for example, fostered the development of systems that could extract many important details from news stories in English and Japanese. The scope of this task was much broader than in any previous project.

The current state of the art has produced rapid advances in the robustness and applicability of these methods. However, current systems are limited because they invariably rely, at least to some degree, on domain knowledge or other specialized models, which still demands time and effort (usually several person-months, even in limited domains). These problems are tempered somewhat by the availability of on-line resources, such as lexicons, corpora, lists of companies, gazetteers, and so forth, but the issue of how to develop a technology base that applies to many problems is still the major challenge.

In recent years, technology has progressed quite rapidly, from systems that could accurately process text in only very limited domains (for example, engine service reports) to programs that can perform useful information extraction from a very broad range of texts (for example, business news). The two main forces behind these advances are: (1) the development of robust text processing architectures, including finite state approximation and other shallow but effective sentence processing methods, and (2) the emergence of weak heuristic and statistical methods that help to overcome knowledge acquisition problems by making use of corpus and training data.

Finite-state approximation (Jacobs, Krupka, et al., 1993; Pereira, 1990) is a key element of current text interpretation methods. Finite-state recognizers generally admit a broader range of possible sentences than most parsers based on context-free grammars, and usually apply syntactic constraints in a weaker fashion. Although this means that finite-state recognizers will sometimes treat sentences as grammatical when they are not, the usual effect is that the finite state approximation is more efficient and fault tolerant than a context-free model.

The success of finite-state and other shallow recognizers, however, depends on the ability to express enough word knowledge and domain knowledge to control interpretation. While more powerful parsers tend to be controlled mainly by linguistic constraints, finite state recognizers usually depend on lexical constraints to select the best interpretation of an input. In limited domains, these constraints are part of the domain model; for example, when the phrase *unidentified assailant* appears in a sentence with *terrorist* *attack*, it is quite likely that the assailant is the perpetrator of the attack.

In broader domains, successful interpretation using shallow sentence processing requires lexical data rather than domain knowledge. Such data can often be obtained from a corpus using statistical methods (Church, Gale, et al., 1991). These statistical models have been of only limited help so far in information extraction systems, but they show promise for continuing to improve the coverage and accuracy of information extraction in the future.

Much of the key information in interpreting texts in these applications comes not from sentences but from larger discourse units, such as paragraphs and even complete documents. Interpreting words and phrases in the context of a complete discourse, and identifying the discourse structure of extended texts, are important components of text interpretation. At present, discourse models rely mostly on domain knowledge (Iwanska, Appelt, et al., 1991). Like the problem of controlling sentence parsing, obtaining more general discourse processing capabilities seems to depend on the ability to use discourse knowledge acquired from examples in place of detailed hand-crafted domain models.

Future Directions

We can expect that the future of information extraction will bring broader and more complete text interpretation capabilities; this will help systems to categorize, index, summarize, and generalize from texts from information sources such as newspapers and reference materials. Such progress depends now on the development of better architectures for handling information beyond the sentence level, and on continued progress in acquiring knowledge from corpus data.

7.4 Summarization

Karen Sparck Jones

University of Cambridge, Cambridge, UK

Automatic abstracting was first attempted in the 1950s, in the form of Luhn's auto-extracts, (cf. Paice, 1990); but since then there has been little work on, or progress made with, this manifestly very challenging task. However the increasing volume of machine-readable text, and advances in natural language processing, have stimulated a new interest in automatic summarizing reflected in the 1993 Dagstuhl Seminar, *Summarizing text for intelligent communication* (Endres-Niggemeyer, Hobbs, et al., 1995). Summarizing techniques tested so far have been limited either to general, but shallow and weak approaches, or to deep but highly application-specific ones. There is a clear need for more powerful, i.e., general but adaptable, methods. But these must as far as possible be linguistic methods, not requiring extensive world knowledge, and ones able to deal with large-scale text structure as well as individual sentences.

7.4.1 Analytical Framework

Work done hitherto, relevant technologies, and required directions for new research are usefully characterized by reference to an analytical framework covering both factors affecting summarizing and the essential summarizing process. I shall concentrate on text, but the framework applies to discourse in general including dialogue.

A summary text is a derivative of a source text condensed by selection and/or generalization on important content. This is not an operational definition, but it emphasizes the crux of summarizing, reducing whole sources without requiring pre-specification of desired content, and allows content to cover both information and its expression. This broad definition subsumes a very wide range of specific variations. These stem from the *context factors* characterizing individual summarizing applications. Summarizing is conditioned by *input factors* categorizing source form and subject; by *purpose factors* referring to audience and function; and also, subject to input and purpose constraints, by *output factors* including summary format and style.

The global process model has two major phases: *interpretation* of the source text involving both local sentence analysis and integration of sentence analyses into an overall *source meaning representation*; and *generation* of the summary by formation of the *summary representation* using the source one and subsequent synthesis of the summary text. This logical model emphasizes the role of text representations and the central transformation stage. It thus focuses on what source representations should be like for summarizing, and on what condensation on important content requires. Previous approaches to summarizing can be categorized and assessed, and new ones designed, according to (a) the nature of their source representation, including its distance from the source text, its relative emphasis on *linguistic*, *communicative* or *domain information* and therefore the structural model it employs and the way this marks important content; and (b) the nature of its processing steps, including whether all the model stages are present and how independent they are.

7.4.2 Past Work

For instance, reviewing past work (see Paice, 1990; Sparck Jones, 1993), source text extraction using statistical cues to select key sentences to form summaries is taking both source and summary texts as their own linguistic representations and also essentially conflating the interpretation and generation steps. Approaches using cue words as a base for sentence selection are also directly exploiting only linguistic information for summarizing. When headings or other locational criteria are exploited, this involves a very shallow source text representation depending on primarily linguistic notions of text grammar, though Liddy et al. (1993) has a richer grammar for a specific text type.

Approaches using scripts or frames on the other hand (Young & Hayes, 1985; DeJong, 1979) involve deeper representations and ones of an explicitly domain-oriented kind motivated by properties of the world. DeJong's work illustrates the case where the source representation is deliberately designed for summarizing, so there is little transformation effort in deriving the summary template representation. In the approach of Rau (1988), however, the hierarchic domain-based representation allows generalization for summarizing.

There has also been research combining different information types in representation. Thus Hahn (1990) combines linguistic theme and domain structure in source representations, and seeks salient concepts in these for summaries.

Overall in this work, source reduction is mainly done by selection: this may use general, application-independent criteria, but is more commonly domain-guided as in Marsh, Hamburger, et al. (1984), or relies on prior, inflexible specification of the kind of information sought, as with DeJong (1979), which may be as tightly constrained as in MUC. There is no significant condensation of input content taken as a whole: in some cases even little length reduction. There has been no systematic comparative study of different types of source representation for summarizing, or of context factor implications. Work hitherto has been extremely fragmentary and, except where it resembles indexing or is for very specific and restricted kinds of material, has not been very successful. The largest-scale automatic summarizing experiment done so far has been DeJong's, applying script-based techniques to news stories. There do not appear to be any operational summarizing systems.

7.4.3 Relevant Disciplines

The framework suggests there are many possibilities to explore. But given the nature and complexity of summarizing, it is evident that ideas and experience relevant to automatic summarizing must be sought in many areas. These include human summarizing, a trained professional skill that provides an iterative, processual view of summarizing often systematically exploiting surface cues; discourse and text linguistics supplying a range of theories of discourse structure and of text types bearing on summarizing in general, on different treatments suited to different source types, and on the relation between texts, as between source and summary texts; work on discourse comprehension, especially that involving or facilitating summarizing; library and information science studies of user activities exploiting abstracts e.g., to serve different kinds of information need; research on user modeling in text generation, for tailoring summaries; and NLP technology generally in supplying both workhorse sentence processing for interpretation and generation and methods for dealing with local coherence, as well as results from experiments with forms of large-scale text structure, if only for generation so far, not recognition. Some current work drawing on these inputs is reported in IPM (1995); it also illustrates a growing interest in generating summaries from non-text material.

7.4.4 Future Directions

The *full text revolution*, also affecting indexing, implies a pressing need for automatic summarizing, and current NLP technology provides the basic resource for this. There are thus complementary shorter and longer term lines of work to undertake, aimed at both practical systems and a scientific theory of summarizing, as follows:

1. Develop shallow-processing techniques that exploit robust parsing, and surface or statistical pointers to key topics and topic connections, for simple indexing-type information extracts and summaries.

2. Seek generalizations of deep, domain-based approaches using e.g., frames, to reduce tight application constraints and extend system scopes.

3. Carry out systematic experiments to assess the potentialities of alternative types of source representation both for any summarizing strategy and in relation to different context factor conditions.

4. Engage in user studies to establish roles and hence requirements for summaries as leading to or providing information, and to determine sound methods of evaluating summaries.

5. Explore dynamic, context-sensitive summarizing for interactive situations, in response to changing user needs as signaled by feedback and as affected by ad hoc assemblies of material.

7.5 Computer Assistance in Text Creation and Editing

Robert Dale

Microsoft Institute of Advanced Software Technology, Sydney, Australia

On almost every office desk there sits a PC, and on almost every PC there resides a word processing program. The business of text creation and editing represents a very large market, and a very natural one in which to ask how we might apply speech and natural language processing technologies. Below, we look at how language technologies are already being applied here, sketch some advances to be expected in the next 5–10 years, and suggest where future research effort is needed.

Information technology solutions are generally of three types: accelerative, where an existing process is made faster; delegative, where the technology carries out a task previously the responsibility of a person; and augmentative, where the technology assists in an existing task. The major developments in the next 5–10 years are likely to be of an augmentative nature, with increasingly sophisticated systems that have people and machines doing what they each do best. The key here is to add intelligence and sophistication to provide *language sensitivity*, enabling the software to see a text not just as a sequence of characters, but as words and sentences combined in particular structures for particular semantic and pragmatic effect.

7.5.1 Creation and Revision of Unconstrained Text: The Current Situation

Although language technologies can play a part in the process of text creation by providing intelligent access to informational resources, the more direct role is in the provision of devices for organizing text. The degree of organizational assistance that is possible depends very much on the extent to which regularity can be perceived or imposed on the text concerned. Document production systems which impose structure support text creation; the most useful offspring here has been the outliner, now a standard part of many word processing systems. However, in general the model of documenthood these systems embody is too constrained for widespread use in text creation. While relatively structured documents are appropriate in some business contexts, other future markets will focus on home and leisure usage, where concerns other than structure may become relevant to the creation of text. In the following two subsections we focus on unconstrained text, whereas controlled languages are treated in section 7.6.

No existing tools in this area embody any real language sensitivity. Much of the initial exploratory work required here has reached computational linguistics via research in natural language generation; but we are still far away from being able to automatically *interpret* discourse structure in any sophisticated sense. Current models of discourse structure do not mirror the sophistication of our models of sentence structure, and so the scope for assistance in text creation will remain limited until significant research advances are made.

The story is very different for text revision. Here, language technology finds a wide range of possible applications. We already have the beginnings of language sensitivity in spelling correction technology: the techniques used here are now fairly stable, although without major advances (for example, taking explicit account of syntax and even semantics) we cannot expect much beyond current performance.

Grammar checking technology is really the current frontier of the state of the art. Commercial products in this area are still much influenced by the relatively superficial techniques used in the early Unix Writer's Workbench (WWB) system, but some current commercial systems (such as Grammatik and CorrecText) embody greater sophistication: these are the first products to use anything related to the parsing technologies developed in the research field. As machines become more powerful, and as broad-coverage grammars become more feasible, we can expect to see more of the CPU-hungry techniques developed in research labs finding their way into products; IBM's Critique system gives a flavor of what is to come.

Beyond grammar checking, the next important step is stylistic analysis. Anything more than the very simple string and pattern matching techniques first used in the Unix WWB system require the substrate of syntactic analysis, and, indeed, there are many aspects of style for which semantic and pragmatic analyses are required. Here more than anywhere the problem of different perceptions of the shape of the task rears its head: style is a term used to cover many things, from the form in which a date should be written to the overall feel of a text. Some of the simpler problems here are already being dealt with in products on the market, and this is where we can expect to see most developments in the next five years.

7.5.2 Future Directions

Medium-term Prospects

The key to medium-term developments in this area is the productization of parsing and grammar technologies. There are a number of shifts in research focus that are needed to accelerate this process.

1. Linguistic theories need to be assessed for their value in this working context: For example, are some theories more suited than others to the development of a theory of syntactic error detection and correction? Do the standard linguistic distinctions between syntax, semantics and pragmatics stand up in this domain?

2. Parsing mechanisms need to be made far more robust than is usually taken to be necessary: no matter how broad coverage a grammar is, there will always be texts that do not conform. How does a system decide that it is faced with an ungrammatical sentence rather than a correct sentence for which it does not have a grammar rule? How is the handling of unknown words best integrated with the handling of grammatical errors?

3. How do we evaluate these systems? Corpora of errors are needed in order to determine which categories of errors are most frequent and where effort is best applied. A real problem here is knowing how to measure performance: the appropriate metrics have not yet been developed. Underlying these requirements is a need for a properly elaborated theory of textual error: what exactly counts as a spelling error as opposed to a syntactic error, for example?

4. How is the user to understand the basis of the system's proposed revisions? Because of the mismatch between the user's view of the problem and the language technologist's view, there is a need for better means of explaining errors to users in an acceptable way.

5. Finally, and most importantly, if we are to progress beyond rather trivial assistance in stylistic matters, we need a sizable effort directed at research on stylistic issues to build computational theories at that level.

Longer-term Prospects

We have already alluded above to the scope for incorporating sophisticated theories of discourse into the creation task in writing tools; similarly, the acceleration and delegation of language-centered tasks will become increasingly viable as advances are made in speech processing and natural language generation in the longer term.

Looking more broadly, we should be concerned not only with the words themselves, but also how they appear on the page or screen. The fact that, for example, we often have to make our texts fit word limits means that we have to take account of physical space. Systems should be able to reason about graphics as well as words, and systems should know about typographic devices.

Beyond these areas, there are new categories of assistance we might expect in the longer term. Modes of writing themselves are likely to adapt to accommodate the uneven profile of ability offered by existing systems, with currently unpredictable back and forwards effects on the tools that become required. We can't easily foresee what new market possibilities for computer-based writing tools the information superhighway will lead to; but there is a strong possibility that the categories we have previously thought in will no longer be the most appropriate.

7.6 Controlled Languages in Industry

Richard H. Wojcik & James E. Hoard

Boeing Information & Support Services, Seattle, Washington, USA

7.6.1 The Reason Why

Natural language permits an enormous amount of expressive variation. Writers, especially technical writers, tend to develop special vocabularies (jargons), styles, and grammatical constructions. Technical language becomes opaque not just to ordinary readers, but to experts as well. The problem becomes particularly acute when such text is translated into another language, since the translator may not even be an expert in the technical domain. Controlled Languages (CL) have been developed to counter the tendency of writers to use unusual or overly-specialized, inconsistent language.

A CL is a form of language with special restrictions on grammar, style, and vocabulary usage. Typically, the restrictions are placed on technical documents, including instructions, procedures, descriptions, reports, and cautions. One might consider formal written English to be the ultimate Controlled Language: a form of English with restricted word and grammar usages, but a standard too broad and too variable for use in highly technical domains. Whereas formal written English applies to society as a whole, CLs apply to the specialized sublanguages of particular domains.

The objective of a CL is to improve the consistency, readability, translatability, and retrievability of information. Creators of CLs usually base their grammar restrictions on well-established writing principles. For example, AECMA Simplified English limits the length of instructional sentences to no more than 20 words. It forbids the omission of articles in noun phrases, and requires that sequential steps be expressed in separate sentences.

7.6.2 Results

By now, hundreds of companies have turned to CLs as a means of improving readability or facilitating translation to other languages. The original CL was Caterpillar Fundamental English (CFE), created by the Caterpillar Tractor Company (USA) in the 1960s. Perhaps the best known recent controlled language is AECMA Simplified English (AECMA, 1995), which is unique in that it has been adopted by an entire industry, namely, the aerospace industry. The standard was developed to facilitate the use of maintenance manuals by non-native speakers of English. Aerospace manufacturers are required to write aircraft maintenance documentation in Simplified English. Some other well-known CLs are Smart's Plain English Program (PEP), White's International Language for Serving and Maintenance (ILSAM), Perkins Approved Clear English (PACE), and COGRAM. (See Adriaens & Schreuers, 1992, which refers to some of these systems). Many CL standards are considered proprietary by the companies that have developed them.

7.6.3 Prospects

The prospects for CLs are especially bright today. Many companies believe that using a CL can give them something of a competitive edge in helping their customers operate and service their products. With the tremendous growth in international trade that is occurring worldwide, more and more businesses are turning to CLs as a method for making their documents easier to read for non-native speakers of the source language or easier to translate into the languages of their customers.

One of the factors stimulating the use of CLs is the appearance of new language engineering tools to support their use. Because the style, grammar, and vocabulary restrictions of a CL standard are complex, it is nearly impossible to produce good, consistent documents that comply with any CL by manual writing and editing methods. The Boeing Company has had a Simplified English Checker in production use since 1990, and Boeing's maintenance manuals are now supplied in Simplified English (Hoard, Wojcik, et al., 1992; Wojcik, Harrison, et al., 1993; LIM, 1993). Since 1990, several new products have come onto the market to support CL checking. A number of others exist in varying prototype stages. The Commission of the European Union has authorized a recent program to fund the development of such tools to meet the needs of companies that do business in the multilingual EU.

7.6.4 Future Directions

There are two principal problems that need to be kept in focus in the language engineering area. The first is that any CL standard must be validated with real users to determine if its objectives are met. If some CL aims, say, to improve readability by such and such an amount, then materials that conform to the standard must be tested to ensure that the claim is valid. Otherwise, bearing the cost and expense of putting materials into the CL is not worth the effort. The second problem is to develop automated checkers that help writers conform to the standard easily and effectively. One cannot expect any checker to certify that a text conforms completely to some CL. The reason is that some rules of any CL require human judgments that are beyond the capability of any current natural language software and may, in fact, never be attainable. What checkers can do is remove nearly all of the mechanical errors that writers make in applying a CL standard, leaving the writer to make the important judgments about the organization and exposition of the information that are so crucial to effective descriptions and procedures. The role of a checker is to make the grammar, style, and vocabulary usages consistent across large amounts of material that is created by large numbers of writers. Checkers reduce tremendously the need for editing and harmonizing document sections. Over the next decade the kinds of CL rules that can be checked automatically will expand. With current technology it is possible to check for syntactic correctness. In the coming years it will also be quite feasible to check a text for conformity with sanctioned word senses and other semantic constraints. This will increase the cost effectiveness of providing documents in a CL to levels that can only be guessed at now.

7.7 Chapter References

Adriaens, G. and Schreuers, D. (1992). From COGRAM to ALCOGRAM: Toward a controlled English grammar checker. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 595–601, Nantes, France. ACL.

AECMA (1995). *AECMA Simplified English: A Guide for the Preparation of Aircraft Maintenance Documentation in the International Aerosace Maintenance Language.* AECMA, Brussels.

Belkin, N. J. and Croft, W. B. (1987). Retrieval techniques. In Williams, M., editor, *Annual Review of Information Science and Technology*, volume 22, pages 109–145. Elsevier, New York.

Broglio, J., Callan, J., and Croft, W. (1993). The INQUERY system. In Merchant, R., editor, *Proceedings of the TIPSTER Text Program—Phase I*, San Mateo, California. Morgan Kaufmann.

Brynjolfsson, E. (1993). The productivity paradox of information technology. *Communications of the ACM*, 36(12).

Chiaramella, Y. and Nie, J. (1990). A retrieval model based on an extended modal logic and its applications to the RIME experimental approach. In Vidick, J.-L., editor, *Proceedings of the 13th International Conference on Research and Development in Information Retrieval*, pages 25–44, Brussels, Belgium. ACM.

Church, K., Gale, W., Hanks, P., and Hindle, D. (1991). Using statistics in lexical analysis. In Zernik, U., editor, *Lexical Acquisition: Using On-Line Resources To Build A Lexicon*. Lawrence Earlbaum, Hillsdale, New Jersey.

Ciravegna, F., Campia, P., and Colognese, A. (1992). Knowledge extraction by SINTESI. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 1244–1248, Nantes, France. ACL.

COLING (1992). *Proceedings of the 14th International Conference on Computational Linguistics*, Nantes, France. ACL.

David, P. A. (1991). *Technology and Productivity The Challenge for Economic Policy*, chapter Computer and Dynamo—The modern productivity paradox in a not-too-distant mirror. ODED, Paris.

DeJong, G. F. (1979). *Skimming stories in real time: an experiment in integrated understanding*. PhD thesis, Yale University.

Endres-Niggemeyer, B., Hobbs, J., and Sparck Jones, K. (1995). Summarizing text for intelligent communication. Technical Report Dagstuhl Seminar Report 79, 13.12-19.12.93 (9350), IBFI, Dagstuhl. http://www.bid.fh-hannover.de/SimSum/Abstract/ (Short and Full versions, the latter only available in electronic form).

Evans, D. and Lefferts, R. (1994). Design and evaluation of the CLARIT–TREC-2 system. In Harman, D., editor, *National Institute of Standards and Technology Special Publication No. 500-215 on the The Second Text REtrieval Conference (TREC-2)*, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Fagan, J. L. (1989). The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, 40(2):115–132.

Hahn, U. (1990). Topic parsing: accounting for text macro structures in full-text analysis. *Information Processing and Management*, 26(1):135–170.

Harman, D., editor (1994). *National Institute of Standards and Technology Special Publication No. 500-215 on the The Second Text REtrieval Conference (TREC-2)*, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Hess, M. (1992). An incrementally extensible document retrieval system based on linguistic and logical principles. In *Proceedings of the 15th SIGIR Conference*, pages 190–197, Copenhagen, Denmarkp.

Hoard, J., Wojcik, R., and Holzhauser, K. (1992). An automated grammar and style checker for writers of simplified English. In Holt, P. and Williams, N., editors, *Computers and Writing*. Kluwer Academic Publishers, Boston.

IPM (1995). Special issue on automatic summarizing. *Information Processing and Management*, 31(3).

Iwanska, L., Appelt, D., Ayuso, D., Dahlgren, K., Glover Stalls, B., Grishman, R., Krupka, G., Montgomery, C., and Riloff, E. (1991). Computational aspects of discourse in the context of MUC-3. In *Proceedings of the Third Message Understanding Conference*, San Diego, California. Morgan Kaufmann.

Jacobs, P. (1994). GE in TREC-2: Results of a Boolean approximation method for routing and retrieval. In Harman, D., editor, *National Institute of Standards and Technology Special Publication No. 500-215 on the The Second Text REtrieval*

Conference (TREC-2), Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Jacobs, P., Krupka, G., Rau, L., Mauldin, M., Mitamura, T., Kitani, T., Sider, I., and Childs, L. (1993). The TIPSTER/SHOGUN project. In *Proceedings of the TIPSTER Phase I Final Meeting*, San Mateo, California. Morgan Kaufmann.

Liddy, E. D. et al. (1993). Development, implementation and testing of a discourse model for newspaper texts. In *Proceedings of the 1993 ARPA Human Language Technology Workshop*, pages 159–164, Princeton, New Jersey. Advanced Research Projects Agency, Morgan Kaufmann.

Liddy, E. D. and Myaeng, S. H. (1993). DR–LINK: A system update for TREC-2. In Merchant, R., editor, *Proceedings of the TIPSTER Text Program—Phase I*, San Mateo, California. Morgan Kaufmann.

LIM (1993). The boeing simplified English checker. *Language Industry Monitor*, (13).

Marsh, E., Hamburger, H., and Grishman, R. (1984). A production rule system for message summarization. In *Proceedings of the National Conference on Artificial Intelligence*, pages 243–246. American Association for Artificial Intelligence.

Mellish, C. S. et al. (1995). The TIC message analyser. *Computational Linguistics.*

Paice, C. D. (1990). Constructing literature abstracts by computer. *Information Processing and Management*, 26(1):171–186.

Pereira, F. (1990). Finite-state approximations of grammars. In *Proceedings of the Third DARPA Speech and Natural Language Workshop*, pages 20–25, Hidden Valley, Pennsylvania. Defense Advanced Research Projects Agency, Morgan Kaufmann.

Rau, L. F. (1988). Conceptual information extraction and information retrieval from natural language input. In *Proceedings of the Conference on User-Oriented, Content-Based, Text and Image Handling*, pages 424–437, Cambridge, Massachusetts.

Ruge (1992). Experiments in linguistically based term associations. *Information Processing and Management*, 28(3).

Schwarz and Thurmair, editors (1986). *Informationslinguistische texterschliessung.* Hildesheim: Georg Olms Verlag.

Smeaton, A. (1992). Progress in the application of natural language processing to information retrieval tasks. *The Computer Journal*, 35(3).

Smeaton, A. F., O'Donnell, R., and Kelledy, F. (1995). Indexing structures derived from syntax in TREC-3: System description. In *National Institute of Standards and Technology Special Publication on the The Third Text REtrieval Conference (TREC-3)*, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Sparck Jones, K. (1993). What might be in a summary? In Knorz, G., Krause, J., and Womser-Hacker, C., editors, *Information retrieval '93: von der modellierung zur anwendung*, pages 9–26. Konstanz, Universitatsverlag Konstanz.

Strzalkowski, T., Carballo, J. P., and Marinescu, M. (1995). Natural language information retrieval: TREC-3 report. In *National Institute of Standards and Technology Special Publication on the The Third Text REtrieval Conference (TREC-3)*, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

TREC (1995). *National Institute of Standards and Technology Special Publication on the The Third Text REtrieval Conference (TREC-3)*, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Voorhees, E., Gupta, N. K., and Johnson-Laird, B. (1995). The collection fusion problem. In *National Institute of Standards and Technology Special Publication on the The Third Text REtrieval Conference (TREC-3)*, pages 95–104, Washington, DC. National Institute of Standards and Technology, U.S. Department of Commerce, U.S. Government Printing Office.

Wojcik, R., Harrison, P., and Bremer, J. (1993). Using bracketed parses to evaluate a grammar checking application. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 38–45, Columbus, Ohio. ACL.

Young, S. R. and Hayes, P. J. (1985). Automatic classification and summarization of banking telexes. In *Proceedings of the Second Conference on Artificial Intelligence Applications*, pages 402–408.