

LIST OF CONTENTS

Vol. 25, No. 1, 1989

Special Issue: Modeling Data, Information and Knowledge

I. Introduction

- | | | |
|----------------------|---|--|
| Edward A. Fox | 1 | Research and development of information retrieval models and their application |
| F. N. Teskey | 7 | User models and world models for data, information and knowledge |

II. Boolean-Based Models

- | | | |
|--|----|---|
| Steven C. Cater
Donald H. Kraft | 15 | A generalization and clarification of the Waller-Kraft wish list |
| Caroline M. Eastman | 27 | Handling incrementally specified Boolean queries: A comparison of inverted and signature file organizations |

III. Probabilistic Models

- | | | |
|--|----|--|
| S. K. M. Wong
Y. Y. Yao | 39 | A probability distribution model for information retrieval |
| Norbert Fuhr | 55 | Models for retrieval with probabilistic indexing |

IV. Artificial Intelligence Models

- | | | |
|------------------------------|-----|---|
| Susanne M. Humphrey | 73 | MedIndEx System: Medical Indexing Expert System |
| Marie-France Bruandet | 89 | Outline of a knowledge-base model for an intelligent information retrieval system |
| | 117 | Announcement |

I Software Survey Section

Vol. 25, No. 2, 1989

- | | | |
|---|-----|---|
| Lesley Trenner | 119 | A Comparative Survey of the Friendliness of Online <i>Help</i> in Interactive Information Retrieval Systems |
| Mark T. Maybury | 137 | Genny: A Knowledge-Based Text Generation System |
| Timo Raita
Jukka Teuhola | 151 | Predictive Encoding in Text Compression |

- | | | |
|--|-----|--|
| Lorraine K.D. Cooper
Alan L. Tharp | 161 | Inverted Signature Trees and Text Searching on CD-ROMS |
| Alan P. Parkes | 171 | The Prototype Cloris System: Describing, Retrieving and Discussing Videodisc Stills and Sequences |
| Stuart Weibel
Michael Oskins
Diane Vizine-Goetz | 187 | Automated Title Page Cataloging: A Feasability Study |
| Patrick Dorelan | 205 | On the Ranking of Psychological Journals
<i>Book Reviews</i> |
| Marcia J. Bates | 215 | <i>Manual of Online Search Strategies</i> by C.J. Armstrong and J.A. Large (Eds.) |
| Thomas Childers | 216 | <i>Education of Library and Information Professionals: Present and Future Prospects</i> by Richard K. Gardner (Ed.) |
| Harold Borko | 218 | <i>Database Design for Information Retrieval: A Conceptual Approach</i> by R. Fidel |
| Karen Butter | 219 | <i>Public Access Microcomputers in Academic Libraries: The Mann Library Model at Cornell University</i> by H. Curtis (Ed.) |
| Pauline A. Cochrane | 220 | <i>The Future of Industrial Information Services; Proceedings of the Library Association Industrial Group 1987 Annual Study Weekend, High Wycombe</i> by L. Wood and R. Haigh (Eds.) |
| Harold Borko | 220 | <i>If You Want to Evaluate Your Library . . .</i> by F.W. Lancaster |
| Linda Z. T. Knowles | 221 | <i>Online Database Search Services Directory: A Reference and Referral Guide to More Than 1,700 Libraries, Information Firms, and Other Sources Providing Computerized Information Retrieval and Associated Services using Publicly Available Online Databases, 2nd ed.</i> by D.M. Maxfield (Ed.) |
| Ethel Auster | 221 | <i>Online Retrieval: Analysis and Strategy</i> by P.J. Vigil |
| Mathilde V. Rovelstad | 222 | <i>International Library Cooperation: Festschrift in Honour of Günther Pflug to His 65th Birthday/10th Anniversary Essen Symposium 19 October-22 October 1987</i> by A.H. Helal and J.W. Weiss (Eds.) |

- | | | |
|--|-----|--|
| Ravi Sharma | 223 | A Generic Machine for Parallel Information Retrieval |
| Christine L. Borgman | 237 | All Users of Information Retrieval Systems Are Not Created Equal: An Exploration Into Individual Differences |
| Susan Gauch
John B. Smith | 253 | An Expert System for Searching in Full-Text |
| U. Güntzer
G. Jüttner
G. Seegmüller
F. Sarre | 265 | Automatic Thesaurus Construction by Machine Learning From Retrieval Sessions |
| Fazli Can
Esen A. Ozkarahan | 275 | Dynamic Cluster Maintenance |
| M.A. Bassiouni
Amar Mukherjee
N. Ranganathan | 293 | Enhancing Arithmetic and Tree-Based Coding |
| Charles V. Packer | 307 | Applying Row-Column Permutation to Matrix Representations of Large Citation Networks |
| Lieve Hamers
Yves Hemeryck
Guido Herweyers
Marc Janssen
Hans Keters
Ronald Rousseau
André Vanhoutte | 315 | Similarity Measures in Scientometric Research: The Jaccard Index Versus Salton's Cosine Formula |
| J. Švarc
A. Švarc | 319 | Application of the Cobb-Douglas Model to the Use of Information Resources by Industry in Croatia, Yugoslavia |

Book Reviews

- | | | |
|--------------------------|-----|--|
| Clare Beghtol | 333 | <i>Subject Access to Visual Resources Collections: A Model for Computer Construction of Thematic Catalogs</i> by K. Markey |
| Naomi Sanger | 334 | <i>The Computational Analysis of English: A Corpus-Based Approach</i> by R. Garside, G. Leech and G. Sampson (Eds.) |
| Beth K. Eres | 335 | <i>The Information Society: An International Perspective</i> by R. L. Katz |
| Patsy Haley Stann | 336 | <i>Directory of Library and Information Professionals</i> by The American Library Association |

- | | | |
|-------------------------------|-----|--|
| Frederick J. Stielow | 337 | <i>The Basics of Information Technology</i>
by J. Rowley |
| Richard V. Janke | 338 | <i>End User Searching: Services and Providers</i>
by M. Kesselman and S.B. Watstein (Eds.) |
| Louis Vagianos | 339 | <i>The Communications Audit: A Guide for Managers</i> by A. Booth |
| Pauline A. Cochrane | 340 | <i>Thesaurus Construction: A Practical Manual</i>
by J. Aitchinson and A. Gilchrist |
| Edward John Kazlauskas | 340 | <i>Information Systems: A Strategic Approach to Planning and Implementation</i> by R.M. Kesner |
| Patsy Haley Stann | 341 | <i>Encyclopedia of Information Systems and Services</i> by A. Lucas and A. Novallo (Eds.) |
| Jacki Becker | 341 | <i>Information Handling Techniques for the Office: Full Text Rules OA?</i> by S. Hills (Ed.) |
| | 343 | Announcement |
| | I | Software Survey Section |

Vol. 25, No. 4, 1989

- | | | |
|--|-----|--|
| Tefko Saracevic | 345 | Editorial |
| Padmini Srinivasan | 347 | Intelligent Information Retrieval Using Rough Set Approximations |
| Peter Schäuble | 363 | Improving the Effectiveness of Retrieval Systems by Information Structures |
| Jean Tague
Ryan Schultz | 377 | Evaluation of the User Interface in an Information Retrieval System: A Model |
| B.F. Peters
G. Philip
F.J. Smith
D. Crookes | 391 | Online Searching Using Speech as a Man/Machine Interface |
| Carolyn L. Foss | 407 | Tools for Reading and Browsing Hypertext |
| Lisa F. Rau
Paul S. Jacobs
Uri Zernik | 419 | Information Extraction and Text Summarization Using Linguistic Knowledge Acquisition |
| Susan Bonzi
Elizabeth Liddy | 429 | The Use of Anaphoric Resolution for Document Description in Information Retrieval |
| Moshe Telem | 443 | Managing Information Growth and Integration in Small Organizations |

Book Reviews

- | | | |
|----------------------------|-----|--|
| Kevin McGarry | 453 | <i>Future Trends in Information Science and Technology: Proceedings of the Silver Jubilee Conference of the City University Department of Information Science</i> by P.A. Yates Mercer (Ed.) |
| Richard S. Marcus | 454 | <i>Knowledge Engineering: Expert Systems and Information Retrieval</i> by I. Wormell (Ed.) |
| George R. Thoma | 456 | <i>Optical Storage Technology 1988: A State of the Art Review</i> by W. Saffady |
| Evelyn H. Daniel | 457 | <i>Libraries in the '90's: What the Leaders Expect</i> by D.E. Riggs and G.A. Sabine |
| Robert M. Hayes | 457 | <i>Information Science: An Integrated View</i> by A. Debons, E. Horne and S. Cronenweth |
| Elizabeth S. Aversa | 458 | <i>Online Search Services in the Academic Library—Planning, Management and Operation</i> by J.M. Sieburth |
| Trudi Bellardo | 459 | <i>Going International: Librarians' Preparation Guide for a Work Experience/Job Change Abroad</i> by L.E. Williamson |
| Elizabeth S. Aversa | 460 | <i>Thesauri Used in Online Databases: An Analytical Guide</i> by L.M. Chan and R. Pollard |
| Pauline A. Cochrane | 461 | <i>Low Budget Librarianship: Managing Information in Developing Countries</i> by W.G. McD. Partridge |
| C.D. Hurt | 461 | <i>Forms for Library Automated Systems: An Illustrated Guide for Selection, Design & Use</i> by L.M. Kershner |
| Mary Lynn Hale | 462 | <i>Computer-Readable Databases: A Directory and Data Sourcebook</i> by K.Y. Marcaccio (Ed.) and M.E. Williams (Founding Ed.) |
| Bob Usherwood | 463 | <i>Charging for Library Information Services</i> by B. Norton |

I Software Survey Section

Vol. 25, No. 5, 1989

- | | | |
|---------------------|-----|---|
| A. Bookstein | 465 | Set-Oriented Retrieval |
| Jianyun Nie | 477 | An Information Retrieval Model Based on Modal Logic |

- | | | |
|--|-----|---|
| Norbert Fuhr
Hubert Hüther | 493 | Optimum Probability Estimation from Empirical Distributions |
| Hilary D. Burton | 509 | The Livermore Intelligent Gateway: An Integrated Information Processing Environment |
| Hideki Nishimoto
Shoji Ura | 515 | Complex View Support for a Library Database System |
| Ye-Sho Chen | 527 | Analysis of Lotka's Law: The Simon-Yule Approach |
| D.N.J. Mostert
J.H.P. Eloff
S.H. von Solms | 545 | A Methodology for Measuring User Satisfaction |
| Ravinder Nath | 557 | Are Frequent Computer Users More Satisfied? |
| Matthew J. Liberatore
George J. Titus
Michael W. Varano
Paul W. Dixon | 563 | An Experimental Investigation of the Effects of Some Information System Design Variables on Performance, Preference, and Learning |

Brief Communication

- | | | |
|-----------------------------------|-----|--|
| James Christopher Westland | 579 | A Net Benefits Approach to Measuring Retrieval Performance |
|-----------------------------------|-----|--|

Opinion Paper

- | | | |
|--------------------------|-----|--|
| Roberto Refinetti | 583 | Information Processing as a Central Issue in Philosophy of Science |
|--------------------------|-----|--|

Book Reviews

- | | | |
|---------------------------|-----|--|
| J.A. Large | 585 | <i>New Horizons for the Information Profession: Meeting the Challenge of Change. Proceedings of the Annual Conference of the Institute of Information Scientists, University of Warwick, 1987</i> by H. Dyer and G. Tseng (Eds.) |
| Samuel T. Waters | 586 | <i>Readings in Distributed Artificial Intelligence</i> by A.H. Bond and L. Gasser (Eds.) |
| Herbert S. White | 586 | <i>Library Services in Theory and Context</i> by M.K. Buckland |
| Lawrence W.S. Auld | 587 | <i>Microcomputer Software for Performing Statistical Analysis: A Handbook Supporting Library Decision Making</i> by P. Hernon and J.V. Richardson (Eds.) |
| Michael S. Nilan | 589 | <i>Information Research: Research Methods in Library and Information Science</i> by N. Tudor-Silovic and I. Mihel (Eds.) |

- | | | |
|--------------------------|-----|---|
| Neal Oliver | 590 | <i>Machine Translation: Linguistic Characteristics of MT Systems and General Methodology of Evaluation</i> by J. Lehrberger and L. Bourbeau |
| Roger R. Flynn | 592 | <i>People and Computers: How to Evaluate Your Company's New Technology</i> by C. Clegg <i>et al.</i> |
| Sheilia S. Intner | 593 | <i>Cataloging Microcomputer Software: A Manual to Accompany AACR 2 Chapter 9, Computer Files</i> by N.B. Olson |
| Robert D. Stueart | 594 | <i>Library School Closings: Four Case Studies</i> by M. Paris |
| Ronald R. Powell | 594 | <i>Basic Statistics for Librarians, 3rd Ed.</i> by I.S. Simpson |
| | 597 | Announcement |
| | I | Software Survey Section |

Vol. 25, No. 6, 1989

- | | | |
|---|-----|--|
| W.B. Croft
T.J. Lucia
J. Cringean
P. Willett | 599 | Retrieving Documents by Plausible Inference: An Experimental Study |
| M. Créhange
O. Foucaut
G. Halin
N. Mouaddib
J.F. Foucaut | 615 | Semantics of User Interface for Image Retrieval: Possibility Theory and Learning Techniques |
| Michael D. Gordon
James P. Fry | 629 | Novel Applications of Information Retrieval to the Storage and Management of Computer Models |
| Fathi Debili
Christian Fluhr
Pierre Radasoa | 647 | About Reformulation in Full-Text IRS |
| Fred J. Damerau
Eric Mays | 659 | An Examination of Undetected Typing Errors |
| Karen E. Lochbaum
Lynn A. Streeter | 665 | Comparing and Combining the Effectiveness of Latent Semantic Indexing and the Ordinary Vector Space Model for Information Retrieval |
| Lorna K. Rees-Potter | 677 | Dynamic Thesaural Systems: A Bibliometric Study of Terminological and Conceptual Change in Sociology and Economics with Application to the Design of Dynamic Thesaural Systems |

Bogdan Trawiński	693	A Methodology for Writing Problem Structured Abstracts
Keith G. Jeffery	703	IDEAS: A System for International Data Exchange and Access for Science
Judy O. Lay		
Jean-Francois Miquel		
Samir Zardan		
Fulvio Naldi		
Ilaria Vannini Parenti		
Ljerka Zmaić	713	Visibility of Peripheral Journals Through the Science Citation Index
Siniša Maričić		
Vladimir Simeon		
Ronald Rousseau	721	Constraint Propagation Applied to Citation Analysis: An Example
Jamshid Beheshti	727	A Cross-Sectional Study of the Use of Library Books by Undergraduate Students
Jamshid Beheshti	737	A Longitudinal Study of the Use of Library Books by Undergraduate Students
		<i>Book Reviews</i>
W.B. Croft	745	<i>Readings in Artificial Intelligence and Databases</i> by J. Mylopoulos and M. Brodie (Eds.)
Deborah Ozga	745	<i>First Stop: The Master Index to Subject Encyclopedias</i> by J. Ryan (Ed.)
Keith M. Cottam	746	<i>Managing Change: A How-To-Do-It Manual for Planning, Implementing, and Evaluating Change in Libraries</i> by S.C. Curzon
Bor-sheng Tsai	747	<i>Using Lotus 1-2-3: A How-To-Do-It Manual for Library Applications</i> by R. Machalow
George S. Bobinski	748	<i>Colleges, Learning and Libraries: The Future</i> by P.J. Pack and F.M. Pack
Michael H. Harris	748	<i>Librarians in Search of Science and Identity: The Elusive Profession</i> by G.E. Bennett
Marianne Broadbent	749	<i>Resource Allocation in Industrial and Commercial Libraries: Optimising New Technology and New Services</i> by L. Wood (Ed.)
	I	List of Contents and Author Index
	XIII	Software Survey Section

AUTHOR INDEX

- Auld, L. W. S., 587
 Auster, E., 221
 Aversa, E., 458, 460

 Bassiouni, M. A., 293
 Bates, M. J., 215
 Becker, J., 341
 Beghtol, C., 333
 Beheshti, J., 727, 737
 Bellardo, T., 459
 Bobinski, G. S., 748
 Bonzi, S., 429
 Bookstein, A., 465
 Borgman, C. L., 237
 Borko, H., 218, 220
 Broadbent, M., 749
 Bruandet, M.-F., 89
 Burton, H., 509
 Butter, K., 219

 Can, F., 275
 Cater, S. C., 15
 Chen, Y.-S., 527
 Childers, T., 216
 Cochrane, P. A., 220, 340, 461
 Cooper, L. K. D., 161
 Cottam, K. M., 746
 Créange, M., 615
 Cringean, J., 599
 Croft, W. B., 599, 745
 Crookes, D., 391

 Damerau, F. J., 659
 Daniel, E. H., 457
 Debili, F., 647
 Dixon, P. W., 563
 Dorelian, P., 205

 Eastman, C. M., 27
 Eloff, J. H. P., 545
 Eres, B. K., 335

 Fluhr, C., 647
 Flynn, R. R., 592
 Foss, C. L., 407
 Foucaut, J. F., 615
 Foucaut, O., 615
 Fox, E., 1
 Fry, J. P., 629
 Fuhr, N., 55, 493

 Gauch, S., 253
 Gordon, M. D., 629
 Guntzer, U., 265

 Hale, M. L., 462
 Halin, G., 615

 Hamers, L., 315
 Harris, M. H., 748
 Hayes, R. M., 457
 Hemeryck, Y., 315
 Herweyers, G., 315
 Humphrey, S. M., 73
 Hurt, C. D., 461
 Hüther, H., 493

 Intner, S. S., 593

 Jacobs, P. S., 419
 Janke, R. V., 338
 Janssen, M., 315
 Jeffery, K. G., 703
 Jüttner, G., 265

 Kazlauskas, E. J., 340
 Keters, H., 315
 Knowles, L. Z. T., 221
 Kraft, D. H., 15

 Large, J. A., 585
 Lay, J. O., 703
 Liberatore, M. J., 563
 Liddy, E., 429
 Lochbaum, K. E., 665
 Lucia, T. J., 599

 Marcus, R. S., 454
 Maričić, S., 713
 Maybury, M. T., 137
 Mays, E., 659
 McGarry, K., 453
 Miquel, J.-F., 703
 Mostert, D. N. J., 545
 Mouaddib, N., 615
 Mukherjee, A., 293

 Naldi, F., 703
 Nath, R., 557
 Nie, J., 477
 Nilan, M. S., 589
 Nishimoto, H., 525

 Oliver, N., 590
 Oskins, M., 187
 Ozga, D., 745
 Ozkarahan, E. A., 275

 Packer, C., 307
 Parenti, I. V., 703
 Parkes, A. P., 171
 Peters, B. F., 391
 Philip, G., 391
 Powell, R. R., 594

 Radasoja, P., 647
 Raita, T., 151
 Ranganathan, N., 293
 Rau, L. F., 419
 Rees-Potter, L. K., 677
 Refinetti, R., 583
 Rousseau, R., 315, 721
 Rovelstad, M. V., 222

 Sanger, N., 334
 Saracevic, T., 345
 Sarre, F., 265
 Schauble, P., 363
 Schultz, R., 377
 Seegmüller, G., 265
 Sharma, R., 223
 Simeon, V., 713
 Smith, F. J., 391
 Smith, J. B., 253
 Srinivasan, P., 347
 Stann, P. H., 336, 341
 Stielow, F. J., 337
 Streeter, L. A., 665
 Stueart, R. D., 594
 Švarc, A., 319
 Švarc, J., 319

 Tague, J., 377
 Telem, M., 443
 Teskey, F. N., 7
 Teuhola, J., 151
 Tharp, A. L., 161
 Thoma, G. R., 456
 Titus, G. J., 563
 Trawiński, B., 693
 Trenner, L., 119
 Tsai, B.-S., 747

 Ura, S., 525
 Usherwood, B., 463

 Vagianos, L., 339
 Vanhoutte, A., 315
 Varano, M. W., 563
 Vizine-Goetz, D., 187
 von Solms, S. H., 545

 Waters, S. T., 586
 Weibel, S., 187
 Westland, J. C., 579
 White, H. S., 586
 Willett, P., 599
 Wong, S. K. M., 39

 Yao, Y. Y., 39

 Zardan, S., 703
 Zernik, U., 419
 Zmaić, L., 713

INFORMATION EXTRACTION AND TEXT SUMMARIZATION USING LINGUISTIC KNOWLEDGE ACQUISITION*

LISA F. RAU, PAUL S. JACOBS and URI ZERNIK
Artificial Intelligence Program, GE Research and Development Center,
Schenectady, NY 12301, U.S.A.

(Received 8 June 1988; accepted in final form 21 October 1988)

Abstract—Storing and accessing texts in a conceptual format has a number of advantages over traditional document retrieval methods. A conceptual format facilitates natural language access to text information. It can support imprecise and inexact queries, conceptual information summarization, and, ultimately, document translation.

The lack of extensive linguistic coverage is the major barrier to extracting useful information from large bodies of text. Current natural language processing (NLP) systems do not have rich enough lexicons to cover all the important words and phrases in extended texts. Two methods of overcoming this limitation are (1) to apply a text processing strategy that is tolerant of unknown words and gaps in linguistics knowledge, and (2) to acquire lexical information automatically from the texts.

These two methods have been implemented in a prototype intelligent information retrieval system called SCISOR (System for Conceptual Information Summarization, Organization and Retrieval). This article describes the text processing, language acquisition, and summarization components of SCISOR.

1. INTRODUCTION

Information obtained from natural language texts, using current technology, is extracted by human beings. Although relevant texts may be found through a process of document retrieval, after those relevant documents are retrieved, a person isolates and digests portions of documents that are of interest. This paradigm is appropriate when obtaining the "source" document is important. However, when conceptual information is desired *from* the source document, artificial intelligence (AI) techniques can aid in automating the information extraction process.

Within the last decade, there have been some limited successes in the development of AI systems that partially parse and understand short texts in constrained domains [1-4]. Based on these successes, a prototype intelligent information system, System for Conceptual Information Summarization, Organization and Retrieval (SCISOR), is currently under development at GE's Research and Development Center. SCISOR is designed to read short articles from an online source of information and extract information from the articles. This information is stored in a conceptual form, and users can then ask questions in natural language. Answers to questions are formulated and passed to a natural language generator, which produces an English response.

SCISOR currently operates in the domain of stories about corporate mergers and acquisitions, although it is built with domain-independence in mind.

Figure 1 illustrates the architecture of the SCISOR system. Each of the boxes in Fig. 1 represents a major component of SCISOR. First, newspaper stories, or questions about the stories, dealing with corporate takeovers are interpreted using the combination of two parsing systems, one labeled "TRUMP" and the other labeled "TRUMPET." Input questions are parsed with the same understanding mechanism as is used for the input stories. The questions are stored along with the stories for future user modeling and to enable the system to answer a user's question when the answer becomes known, if it is not known to

*A shorter version of this article was presented at the Conference "User-oriented Content-based Text and Image Handling" (RIA088) at MIT, Cambridge, March 1988.

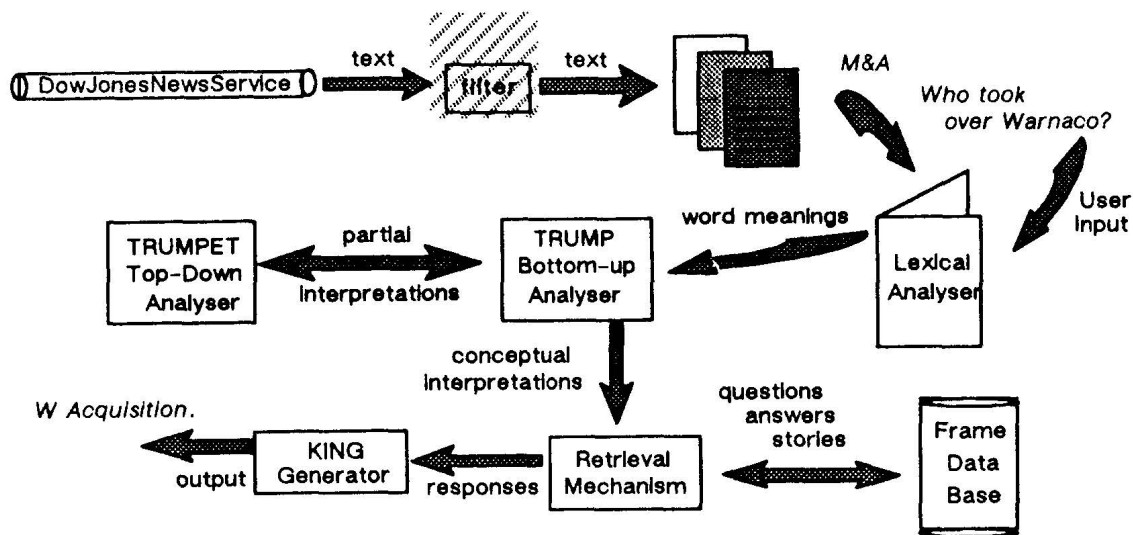


Fig. 1. SCISOR system architecture.

the system at the time the question is posed. Answers to input questions are retrieved with a two-pass retrieval mechanism. After answers have been retrieved, they are passed to the KING (Knowledge INTensive Generator) [5] natural language generator for expression.

Information extracted from the input stories, as well as the meaning of input questions, are represented in the KODIAK [6] knowledge representation language. KODIAK has been augmented with some scriptal knowledge [7] of typical sequences of events in the domain. These domain-dependent scripts must be encoded by hand for each application area. The following is a typical input to the SCISOR system:

Group Offers to Sweeten Warnaco Bid

April 8—An investor group said yesterday that it is prepared to raise its cash bid for Warnaco, Inc. from \$40 a share to at least \$42.50, or \$433.5 million, if it can reach a merger agreement with the apparel maker.

The California-based group, called W Acquisition Corp., already had sweetened its hostile tender offer to \$40 a share from the \$36 a share offered when the group launched its bid in mid-March.

SCISOR is a complex system with many components. The sections that follow consider three capabilities of the SCISOR system in more detail: the text processing mechanisms, the summarization component, and the strategy for automatic acquisition of unknown words. A previous article [8] discusses the retrieval mechanism and question-answering capabilities of SCISOR.

2. TEXT PROCESSING

Natural language analysis of free text is an extremely difficult and challenging problem. However, in constrained domains with sufficient knowledge about the domain, various techniques have been successfully applied to the problem. Two such techniques are to perform a "top-down," expectation-driven conceptual analysis as in [1,9] or analyze the domain in terms of a sublanguage [3,10], and use that sublanguage to guide understanding. The approach to text processing in SCISOR is to use a variety of knowledge sources to improve both the quantity and quality of information extracted from texts.

The two primary techniques integrated in SCISOR are top-down conceptual analysis and bottom-up linguistic analysis. This combination of bottom-up (full parsing) and top-down (partial parsing) processing allows a text processing system to make use of all sources of information it may have in the understanding of texts in a domain. It can perform

in-depth analysis when more complete grammatical and lexical knowledge is present, or perform more superficial analysis when faced with unknown words and constructs. This flexibility is necessary to achieve robust text processing. Bottom-up methods tend to produce more accurate parses and semantic interpretations, account for subtleties in linguistic expression, and detect inconsistencies and lexical gaps. Top-down methods are more tolerant of unknown words or grammatical lapses, but are also more apt to derive erroneous interpretations, fail to detect inconsistencies between what is said and how it is interpreted, and often cannot produce any results when the text presents unusual or unexpected information. Integration of these two approaches can improve the depth and accuracy of the understanding process.

The bottom-up processing is performed by TRUMP (TRansportable Understanding Mechanism Package). TRUMP is a suite of natural language processing tools. The system currently includes a parser, a semantic interpreter, and a set of lexical acquisition tools, although its theoretical foundation lies in work done on knowledge representation for natural language generation [5]. Some of the background of TRUMP is described in [11].

The TRUMP parser combines words into phrases and sentences, checks syntactic constraints, and instantiates linguistic relations. The system's acquisition strategy is to facilitate the addition of domain-specific vocabulary while preserving a core of linguistic knowledge across domains. Consider a phrase such as "the investor group seeking control of Warnaco." TRUMP may be capable of parsing the phrase without much knowledge about corporate takeovers, but with such knowledge it will produce both a more precise interpretation and a better choice in the case of ambiguity. "The investor group" describes a group of investors, not a group that also invests; "seeking control" refers to the takeover objective, not really a "search"; "of Warnaco" means that Warnaco is to be controlled, and not that Warnaco is currently in control of something else. These decisions are obvious, yet they all actually require a great deal of knowledge on the part of the system. This knowledge is sometimes used to help parsing, but mostly it is used to produce more specific interpretations from general concepts.

TRUMP is designed to allow a variety of knowledge sources to contribute to the interpretation of the input. TRUMP does not really care where additional information is coming from, but tries to fit any more specific interpretations with what it has already produced. Thus, the specific interpretation of "investor group" may be derived from a lexical entry for that compound nominal, while the interpretation of "seeking control" depends on domain knowledge about corporate takeovers.

Top-down processing in SCISOR is performed by a partial parser called TRUMPET (TRUMP Expectation Tool). TRUMPET incorporates a more top-down, or expectation-driven mode of operation. In this mode, various sources of information are combined to aid in the extraction of expected pieces of information from text. These pieces of information are integrated to derive as complete an interpretation as possible.

TRUMPET uses the following sources of information to perform its top-down interpretation of the input.

1. Role-filler expectations: Constraints on what can fill a conceptual role add to the confidence of assignments. These are implemented as declarative properties of the roles.
2. Event expectations: Event expectations created from previous stories are used to predict values in new events. These are implemented as demons that are activated upon the instantiation of a typically continuing event.
3. World knowledge expectations: World knowledge expectations can disambiguate multiple interpretations through domain-specific heuristics. These heuristics are implemented as special-purpose procedures.

The program can operate with any combination of these information sources. When one or more sources are lacking, the information extracted from the texts is more superficial, or has a small degree of reliability. This flexibility in depth of processing is an interesting feature in its own right, in addition to having the practical advantage of allowing the sys-

tem to “skim” efficiently when a text overlaps with material already processed or not pertaining to the topic at hand.

As an example, consider the following sample text:

Warnaco received another merger offer, valued at \$36 a share, or \$360 million.

Role-filler expectations allow SCISOR to make reliable interpretations of the dollar figures in spite of incomplete lexical knowledge. Event expectations lead to the deeper interpretation that this offer is an increase over a previous offer. World knowledge allows the system to predict whether the offer was an increase or a competing offer, depending on what other information is available. A unique feature of SCISOR is that partial linguistic knowledge contributes to all of these interpretations and to the understanding of “received” in this context. This is noteworthy because general knowledge about “receive” in this case interacts with domain knowledge in understanding the role of Warnaco in the offer.

This top-down, or expectation-driven, approach, offers the benefit of being able to “skim” texts for particular pieces of information, passing gracefully over unknown words or constructs and ignoring some of the complexities of the language. A typical, although early, effort at skimming news stories was implemented in FRUMP [1], which accurately extracted certain conceptual information from texts in preselected topic areas. FRUMP proved that the expectation-driven strategy was useful for scanning texts in constrained domains. This strategy includes the banking telex readers TESS [4] and ATRANS [12].

Clearly, a robust parser and semantic interpreter could obtain predefined features from the texts without the use of expectations. This would make predefining items of interest unnecessary. Robust in-depth analysis of texts, however, is beyond the near-term capabilities of natural language technology; thus, SCISOR is designed with the understanding that there will always be gaps in the system’s knowledge base that must be dealt with gracefully. SCISOR’s text processing techniques are more fully described in [13].

After information has been extracted using the text processing system just described, it is instantiated and stored using processes not described here. This is how graphs such as in Figs. 2 and 3 are constructed. The rest of this section covers the acquisition component of SCISOR’s text processing mechanism, and section 3 explains how the system uses retrieved information for summarization.

2.1 Acquisition in SCISOR

Although the vast majority of lexical and conceptual knowledge in the SCISOR system is still hand-encoded, the system’s lexicon is the result of the combination of online

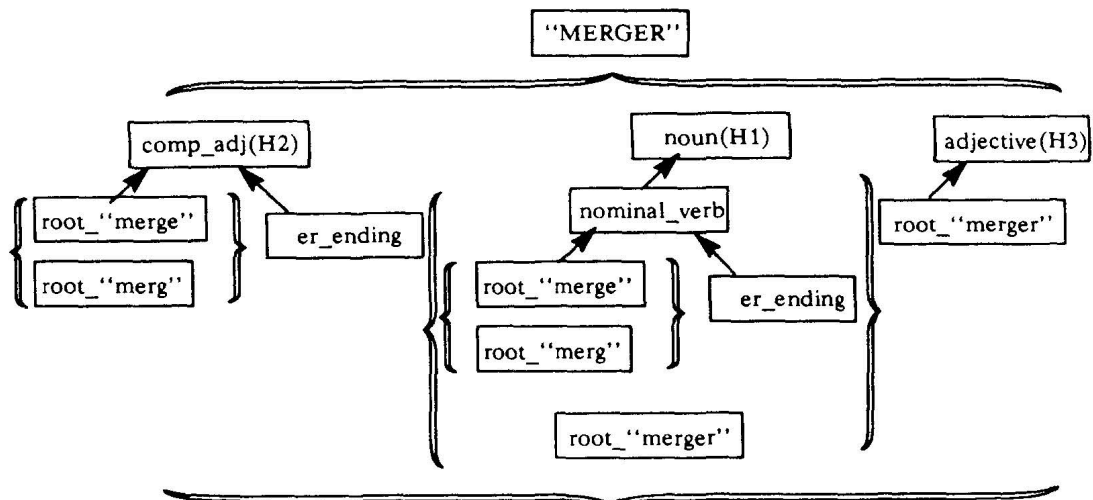


Fig. 2. Word analysis chart of *merger*.

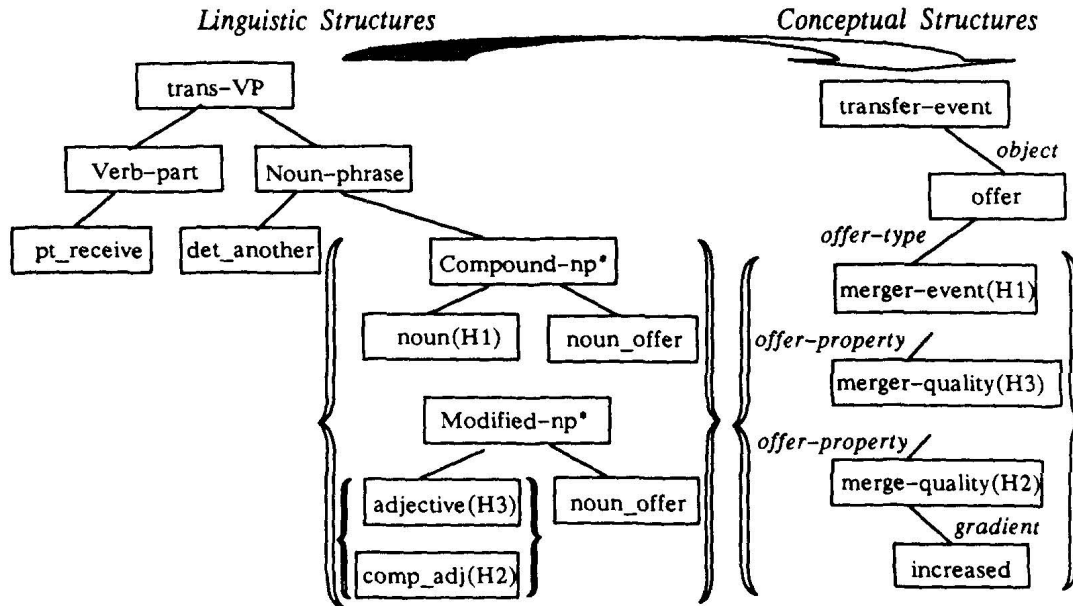


Fig. 3. Hypothesis chart for received another merger offer.

sources, interactive tools described in [14], and automatic acquisition techniques [15]. The core (domain-independent) lexicon of the system contains about 10,000 entries. Some hand-coded terminology supplements this, especially to handle compound nominals and other specialized constructs.

The long-term acquisition strategy of SCISOR is to augment the core lexicon using machine learning techniques. In this approach, an addition to the lexicon starts as a set of assumptions derived from the analysis of each word in a textual context. These assumptions, along with justifications and dependencies, are interpreted and refined by a learning program that ultimately updates the system's lexicon. The program uses TRUMP to tolerate lexical gaps and a learning module called RINA [16] to acquire new words. This method uses the initial structure of the lexicon and generalization of multiple occurrences to create applicable new lexical entries.

2.1.1 The core lexicon A robust text processing system is of limited usefulness without a large vocabulary of words and their syntactic and semantic properties.

Currently, there are a number of publicly available sources of online dictionaries and thesauruses. These sources require extensive modification to be accessible to a real language analyzer [17]. Other efforts are directed at producing large, public domain lexicons suitable for language analysis, but these will take years to become a reality.

SCISOR incorporates a 10,000-root lexical analyzer, specifically tailored for use in language understanding. Gaps in the lexicon, typically because of technical words and proper nouns, can be overcome by the periodic manual addition of the unknowns to the lexicon. This brute-force process will be replaced as automatic techniques become refined. The discussion that follows describes how lexical gaps are handled in the system and how repeated occurrences of new words are used to update the SCISOR lexicon.

2.1.2. Coping with lexical gaps. The input to TRUMP is natural language text; the output from the system is a data structure, or chart, representing the space of possible syntactic and semantic interpretations. When the system encounters unknown words, TRUMP uses assumptions about the unknown words to construct the chart. If, for example, TRUMP reads the sentence **Warnaco received another merger offer**, and does not know the word **merger**, the following knowledge sources contribute to these assumptions:

1. **Morphology.** **Merger** has several morphological breakdowns, each conveying some conceptual content. **Merger** can be either a simple root (like **ledger**); a comparative adjective with root **Merg-** or **Merge-** (like **larger**); or a noun (like **shopper**,

buyer, and **seller**), where **merger** describes the actor of some hypothetical action—the yet unknown **merge/merg** action in this case.

2. *Syntax*. The word may be an adjective or noun serving as a modifier, and may be part of the compound nominal **merger offer**.
3. *Semantics/conceptual*. **Merger offer** may describe a type of offer or something being offered. It may also describe an offer given by someone who **merges**.
4. *Semantics/lexical*. **Received . . . merger offer** suggests that a **merger offer** was the object of an abstract transfer event in which Warnaco was the recipient.
5. *Context*. The specifier **another** presupposes the existence of a **merger offer** in the context. Thus, either there was an offer before and this offer is **merger** than that one, there was another **merger offer** before, or there was an offer and this is one of a group of offers that are **merger** than the first.

2.1.3 *Analyzing unknown words*. For an unknown word, TRUMP produces an ordered list of candidate interpretations, including morphological and syntactic characteristics. These are then used in completing the syntactic/semantic chart in much the same manner as in the case of known words.

For **merger**, the result of the initial analysis is a set of candidates, as shown in Fig. 2. Braces in the figure group together sets of mutually exclusive assumptions. The main assumptions are labeled H1, H2, or H3 in the diagram. Thus, **merger** is either a comparative adjective meaning more *merge* (H2), a noun (H1), or a single adjective (H3) in this linguistic context. If the word derives from a root plus an **-er** ending, the root can be either **merge-** or **merg-**.

2.1.4 *Producing a hypothesis chart* The previous section described the analysis of a single unknown word in a linguistic context. The next step is to build from this analysis an interpretation of an entire sentence. TRUMP constructs a semantic representation of its input by mapping linguistic structures into conceptual structures and combining the new structures. The hypothesis chart includes the possible conceptual interpretations along with the assumptions on which they depend. For example, the hypothesis chart for part of sentence (1) is shown in Fig. 3.

The Warnaco sentence results in the hypothesis chart shown in Fig. 3. The linguistic assumptions H1–H3 are identical to those in Fig. 2; thus, the hypothesis chart includes the word analysis. Conceptual assumptions, such as that *merge* is a quality and that **merger offer** is thus more *merge*, are dependent on corresponding linguistic assumptions (labeled H3 in this case). These assumptions in turn depend on the morphological analysis shown in the previous diagram. When a conceptual assumption proves false, the corresponding linguistic assumptions can be corrected because they are explicitly related within the chart. Similarly, support for a conceptual assumption also supports a corresponding linguistic assumption. The main advantage of this approach is that it allows multiple sources of linguistic knowledge to contribute to the language acquisition process. This process is described in the next section.

2.1.5 *Learning new lexical entries*. From the assumptions introduced by TRUMP, RINA generates full-fledged lexical hypotheses associating structural forms and their meanings. Initially RINA receives a pair: a *chart* representing the text and a *context*, presenting the objects referred to in prior discourse. RINA creates an association between the new words and a conceptual template, and ultimately generalizes this association to form a new lexical entry.

For example, from the phrase *another merger offer* and the context of a corporate takeover offer, RINA can hypothesize the following association:

Template 1

pattern: merger offer
concept: X offer Y to acquire company Y

From this initial template and the analysis of the unknown that TRUMP has provided, RINA makes three assumptions about the new word:

1. **Merger** is a noun, root **merger**.
2. In the compound **merger offer**, the noun **merger** is a modifier, modifying the noun **offer**.
3. **Merger offer** is an instance of a general lexical entry transaction offer, which denotes *an offer to perform a transaction*.

Through these assumptions, RINA interprets Template 1 above as a productive composition of its constituents:

Template 2

pattern: <:modifier merger :head-noun offer>
concept: offer to perform a merger-transaction

RINA employs structure matching with the productive pattern of Template 2, to break down Template 1 into its constituents. Accordingly, merger is associated with *company acquisition*, leading to the following lexical hypothesis:

Template 3:

Pattern: merger
concept: company-acquisition

The templates above play two important roles. Template 1 helps to construct a meaning for the original sentence. Template 3 represents a long-term hypothesis to be used in future examples. Later examples may refute or refine this hypothesis, as illustrated by the following sentences:

1. The merger was completed last week.
2. Bendix finally merged with United Technologies.
3. Midcon will be merged into a subsidiary of Occidental.

Sentence (1) provides a *confirmation* of the hypothesis above (and rules out other spurious hypotheses that are not described here). Sentences (2) and (3) initiate a new assumption: the morphological relation between **merger** the noun, and **merge/merg** the verb, is used for transferring semantic properties of the already known noun to the yet unknown verb. Accordingly, to **merge** means *perform company acquisition*. RWA requires specific combinations of examples to refine the meaning further, but this mechanism allows for the graceful handling of new words in a text processing context.

It is critical that the mechanism for dealing with lexical gaps be integrated with the method for acquiring new lexical entries. Even if RINA never progressed beyond Template 1 above, SCISOR would benefit from having this template as a placeholder for later occurrences of the word *merger*. The following are several advantages of this method:

- The system can use assumptions from coping with a lexical gap to respond to requests. For example, if the phrase *Hyperactive Technologies* is used in a text, and SCISOR assumes it is a name without knowing what it describes, the name itself becomes part of the structure that is used to index the text. Placing the unknowns into a partial interpretation of the text is thus much more effective than ignoring the unknowns completely.
- Assumptions about the behavior of a new word can avoid excessive processing the next time the word appears, and help in forming a hypothesis about the next occurrence. For example, if SCISOR reads *the merger was completed* before it encounters *another merger offer*, it can avoid considering the possibility that merger is a comparative at all.
- The contextual clues in each occurrence of a word contribute to the final lexical structure of the word; thus, even if this structure is incomplete or abstract, it reflects

the texts themselves. An assumption, for example, that *crude* is a type of oil, is certainly not a good representation of the meaning of the word *crude*, but it is very effective in responding to inquires about financial texts.

This section has described SCISOR's approach to the acquisition and application of lexical knowledge. The next section will present how SCISOR uses its conceptual representation of events to generate summaries of events that take place over time.

3. TEXT SUMMARIZATION

In application areas dealing with events that take place over time, the ability to summarize information is especially important. In accessing information about corporate mergers, for example, a user is often interested in a concise summary. A large set of documents, each describing a minor event with a rehash of all events up to that point, would overload that user. Summarization also turns some of the limitations of top-down approaches into an advantage: since the system is unable to understand many of the details of the stories, it tends to rely on common, "expected" structures as the basis of a summary. The material that is understood is often most relevant to a summary.

Summaries do not replace source texts. In many cases, a document or section of a document is an appropriate response to a user query. In other cases, however, a summary is much more relevant than the documents.

For the purposes of summarization, the distinction between narrative texts and expository texts is important. In narrative texts where a story is being told, there is a plot. Events central to the plot are described in the same way as superfluous events. Thus, the task of summarization becomes one of determining what the plot is and gathering together descriptions of events central to that plot. Some examples of approaches to narrative text summarization may be found in [18–20].

Expository texts do not have "plots" per se. As a result of this, the events described in an expository text may all be relevant. For example, it can be assumed that the facts present in a newspaper story are all somehow relevant to that story. The task in expository text summarization then becomes one of determining the appropriate *level of detail* for the summary.

With expository texts, the conciseness of a summary can be roughly approximated by a linear function of the distance of the concepts to be expressed from the containing category that represents what is to be summarized. In narrative texts, a program must have a more complicated method of determining the relative importance of the events described in the story. Although the "linear function of the distance" heuristic does not always produce an optimal summary, it is a starting point for more complex heuristics.

In SCISOR, summaries are currently being generated in a conceptual form, and are to be converted into English output when the KING generator is connected. At the time of this publication the generator can produce simple answers but not complete summaries. The program uses several heuristics to guide the selection of nodes that are passed to the generator. One simple heuristic is to limit the number of concepts included in the summary. Another heuristic is to cluster concepts that are members of the same category. For example, in Fig. 4, three components of the representation of the corporate takeover scenario are members of the LEGAL-COMPLICATION category. This would cause the sentence "... involving some potential conflict of interest matters . . ." to be generated in place of the details of those legal complications.

Figure 4 also illustrates a portion of a corporate takeover scenario. The encircled concepts are the concepts that are to be expressed in the summary. This figure corresponds to one summary that might be given below.

Summary: Rumors that ACME was to be taken over started May 13, 1985. The stock rose \$5 a share to \$65. On May 16, the ACE company announced that it had made an offer to ACME valued at \$66 a share. The offer was a cash offer to purchase all 2 million outstanding shares of ACME. After a month of deliberation involving some potential conflict of interest matters, the ACE-ACME deal was completed last Friday.

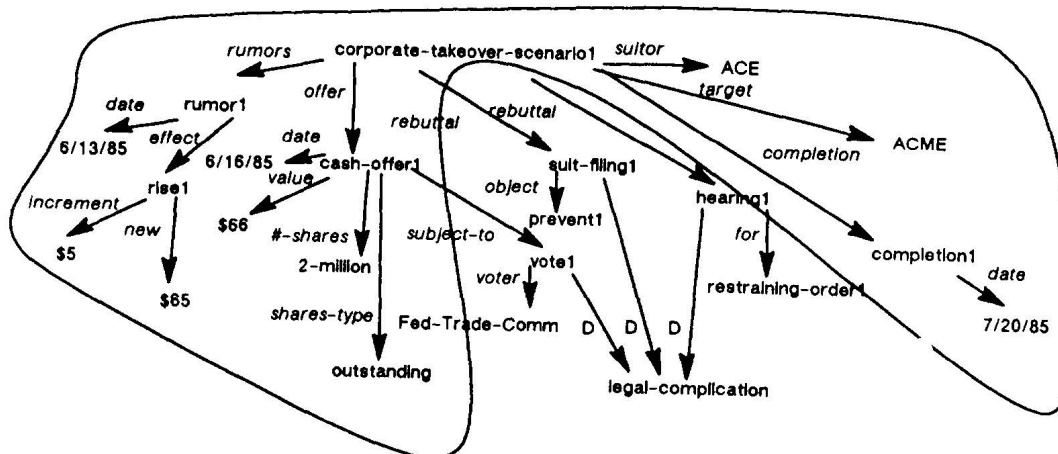


Fig. 4. Story to be summarized.

The natural language generator of SCISOR has not yet been connected so that output such as this is currently “canned.”

SCISOR also uses the “linear function of distance” heuristic to produce appropriate summaries. For example, a very long article about a takeover may have many extraneous, historical details and surrounding context that might not be appropriate for a summary of that takeover. Although these details are included in the representation of the article, they are not used in summarization because of their conceptual distance from the corporate takeover script.

To guide the summary of an event, the user must refer to the event at an appropriate conceptual level. For example, to elicit the summary above, the user might request “Please give me a summary of the events in the ACE-ACME takeover.” The program outputs the contents of any conceptual category that is inquired about by the user. If that category is a high-level one, a summary will be produced. If it is very specific, something very specific such as an answer to a specific question will be produced. If the user asks for a “short” summary or an “extended” summary, SCISOR uses heuristics to compute a relative measure of length, based on the longest paths existing in the particular representations.

4. IMPLEMENTATION AND STATUS

SCISOR is an implemented system written in Common Lisp. It is capable of processing stories at the rate of a few seconds per sentence. It currently has available to it a core lexicon of over 10,000 words.

In recent tests, SCISOR has constructed partial interpretations of a few days’ worth of stories taken directly from an online news service, and answered simple questions about them.

In the immediate future, we plan to begin collecting large amounts of conceptually analyzed material and extensively testing the system. In the longer term, we hope to continue to increase the size of the lexicon and system grammar and improve the capabilities of the bottom-up TRUMP parser. The most important extension, however, will be to test SCISOR in real-life information gathering tasks.

As the system has been engineered to be portable across domains, we will eventually choose another application and tailor the system for this new domain. This should provide valuable insight into issues of system extensibility and portability and the prospects for potential “generic” text processing packages.

5. SUMMARY

SCISOR represents a unique combination of technologies aimed at the robust and reliable extraction of information from naturally occurring texts in constrained domains. The

system processes texts using a substantial grammar and lexicon as well as expectations about the conceptual content of the texts. Machine learning techniques augment the lexicon with domain-specific terms. A set of heuristics for summarization helps to provide cooperative responses to user queries where retrieval of the original texts is unhelpful. This integration of extensive text processing capabilities with automated language acquisition and a summarization algorithm forms a practical strategy for providing conceptual information retrieval in limited domains.

REFERENCES

1. DeJong, G. Skimming stories in real time: An experiment in integrated understanding. Research Report 158. New Haven, Connecticut: Department of Computer Science, Yale University; 1979.
2. Lebowitz, M. Generalization from natural language text. *Cognitive Science*, 7(1): 1-40; 1983.
3. Kittredge, R.; Lehrberger, J. *Sublanguages: Studies of language in restricted domains*. New York: Walter DeGruyter; 1982.
4. Young, S.; Hayes, P. Automatic classification and summarization of banking telexes. In: *The second conference on artificial intelligence applications*. IEEE Press; 1985: 402-208.
5. Jacobs, P.S. Knowledge-intensive natural language generation. *Artificial Intelligence*, 33(3): 325-378; 1987.
6. Wilensky, R. Knowledge representation—a critique and a proposal. In: Kolodner, J., Riesbeck, C., editors. *Experience, memory, and reasoning*. Hillsdale, New Jersey: Lawrence Erlbaum Associates; 1986: 15-28.
7. Schank, R.C.; Abelson, R.P. *Scripts, plans, goals, and understanding*. Halsted, New Jersey: Lawrence Erlbaum; 1977.
8. Rau, L.F. Knowledge organization and access in a conceptual information system. *Information Processing and Management, Special Issue on Artificial Intelligence for Information Retrieval*, 23(4): 269-283; 1987.
9. Cullingford, R.E. *Natural language processing: A knowledge-engineering approach*. Totowa, New Jersey: Rowman and Littlefield; 1986.
10. Grishman, R.; Hirschman, L. *PROTEUS and PUNDIT: Research in text understanding*. PROTEUS Project Memorandum 1. New York: NYU; 1986.
11. Jacobs, P.S. Language analysis in not-so-limited domains. In *Proceedings of the fall joint computer conference*; 1986; Dallas, Texas.
12. Lytinen, S.; Gershman, A. ATRANS: Automatic processing of money transfer messages. In *Proceedings of the fifth national conference on artificial intelligence*; 1986; Philadelphia.
13. Rau, L.F.; Jacobs, P.S. Integrating top-down and bottom-up strategies in a text processing system. In *Proceedings of second conference on applied natural language processing*. Association for Computational Linguistics; Morristown, New Jersey: 1988: 129-135.
14. Besemer, D. FLUSH: Beyond the phrasal lexicon. Technical Report 086CRD181. General Electric Corporate Research and Development; 1986.
15. Jacobs, P.S.; Zernik, U. Acquiring lexical knowledge from text. In *Proceedings of the seventh national conference on artificial intelligence*. Palo Alto, CA: Morgan Kaufmann Inc.; 1988.
16. Zernik, U. Language acquisition: Learning phrases in a hierarchy. In *Proceedings of the tenth international joint conference on artificial intelligence*; 1987; Milan, Italy.
17. Boguraev, B.; Briscoe, T.; Carroll, J.; Carter, D.; Grover, C. The derivation of a grammatically indexed lexicon from the longman dictionary of contemporary English. In *Proceedings of the 25th meeting of the Association for Computational Linguistics*; 1987; Palo Alto, California.
18. Lehnert, W.G. Plot units: A narrative summarization strategy. In: Lehnert, W.; Ringle, M., editors. *Strategies for natural language processing*. Hillsdale, New Jersey: Lawrence Erlbaum Associates; 1982: 375-412.
19. Klein, S. Meta-compiling text grammars as a model for human behavior. In *Proceedings of TINLAP-1*; 1975; Cambridge, Massachusetts.
20. Dyer, M.G. *In-depth understanding*. Cambridge, Massachusetts: MIT Press; 1983.