

IN-DEPTH UNDERSTANDING

A Computer Model of Integrated Processing for Narrative Comprehension

Michael George Dyer

**The MIT Press
Cambridge, Massachusetts
London, England**

Copyright © 1983 by Michael G. Dyer and The Massachusetts Institute of Technology

All rights reserved. No part of this book may be reproduced in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the author or publisher.

This book was printed and bound in the United States of America.

Publisher's note: This format is intended to reduce the cost of publishing certain work in book form and to shorten the gap between editorial preparation and the final publication. Detailed editing and composition have been avoided by photographing the text of this book directly from the author's typescript or word-processor output.

This dissertation was presented to the Faculty of the Graduate School of Yale University on May 1982.

An appendix has been added which includes a pedagogical parser, along with exercises.

This work was supported in part by the Advanced Research Projects Agency under contract NOO14-75-C-1111 and in part by the National Science Foundation under contract IST7918463.

Library of Congress Cataloging in Publication Data

Dyer, Michael George.

In-depth understanding.

Bibliography: p.

Includes index.

1. Artificial intelligence—Mathematical models.

2. Comprehension—Data processing. I. Title.

Q335.D93 1983 001.53'5 83-7926

ISBN 0-262-04073-5

Series Foreword

Artificial intelligence is the study of intelligence using the ideas and methods of computation. Unfortunately, a definition of intelligence seems impossible at the moment because intelligence appears to be an amalgam of so many information-processing and information-representation abilities.

Of course psychology, philosophy, linguistics, and related disciplines offer various perspectives and methodologies for studying intelligence. For the most part, however, the theories proposed in these fields are too incomplete and too vaguely stated to be realized in computational terms. Something more is needed, even though valuable ideas, relationships, and constraints can be gleaned from traditional studies of what are, after all, impressive existence proofs that intelligence is in fact possible.

Artificial intelligence offers a new perspective and a new methodology. Its central goal is to make computers intelligent, both to make them more useful and to understand the principles that make intelligence possible. That intelligent computers will be extremely useful is obvious. The more profound point is that artificial intelligence aims to understand intelligence using the ideas and methods of computation, thus offering a radically new and different basis for theory formation. Most of the people doing artificial intelligence believe that these theories will apply to any intelligent information processor, whether biological or solid state.

There are side effects that deserve attention, too. Any program that will successfully model even a small part of intelligence will be inherently massive and complex. Consequently, artificial intelligence continually confronts the limits of computer science technology. The problems encountered have been hard enough and interesting enough to seduce artificial intelligence people into working on them with enthusiasm. It is natural, then, that there has been a steady flow of ideas from artificial intelligence to computer science, and the flow shows no sign of abating.

The purpose of this MIT Press Series in Artificial Intelligence is to provide people in many areas, both professionals and students, with timely, detailed information about what is happening on the frontiers in research centers all over the world.

Patrick Henry Winston
Michael Brady

Preface

It takes a great deal of knowledge and processing to understand narrative text. BORIS attempts to understand just a few very complicated stories as deeply as possible in contrast to the approach of skimming a great number of stories. As a result, the BORIS project has had to deal with little explored areas concerning knowledge interactions, expectation failures, multiple perspectives, and multiple levels of abstraction, including thematic patterns. Novel representational problems have also been addressed. For instance, unlike previous story understanders, BORIS contains a representational system of AFFECT primitives for dealing with the affective reactions of narrative characters.

Understanding narratives "in depth" means being able to do more than simply extract the facts of a narrative and infer causal connections between them. An in-depth understander must be able to recognize what was memorable about a narrative, what episodes were of significance, and what the point of the narrative was -- that is, why the narrative was worth telling in the first place. Finally, if a narrative is significant in some way, then the process of in-depth comprehension should result in indexing the narrative in memory so that it will come to mind in appropriate future situations.

At this point in time, people are the only information processors able to understand and appreciate the significance of complex narratives. Consequently, the process model embodied in BORIS is based on numerous subject protocols over tasks involving title selection, text summarization, and question answering. For instance, experimental evidence suggests that subjects process text in a highly integrated manner. Syntactic, semantic, and episodic processes arise as each word or phrase is encountered. Therefore, BORIS takes the same theoretical approach.

The thesis presented here represents an initial attempt to specify and model those knowledge constructs, inference strategies, and memory search processes which are prerequisite for in-depth understanding.

Acknowledgments

First, I would like to thank my thesis advisor, Professor Wendy Lehnert, who served as overseer of the BORIS project and without whom this thesis would have been impossible. She supported me through rough times and that is when it counts. Wendy also helped me organize the chaos of multiple problems (that is BORIS) into a coherent thesis.

I also want to thank the other readers on my committee, Professor Bob Abelson and Professor Roger Schank, who both made valuable comments and suggestions on the final draft of this thesis. Professor Abelson has been a source of constant support throughout my years at Yale. He has also directed a number of experiments on thematic structures in BORIS. Professor Schank has been the architect of an environment at the Yale Artificial Intelligence Lab which is conducive to excellence in research. BORIS is built on numerous theoretical constructs developed by Professor Schank and his previous students.

Very special thanks go to Margot Flowers, who made many valuable contentful, editorial, and stylistic suggestions to this dissertation. Margot was the first to read many of the chapters which follow.

Over the years a great number of individuals at the Yale A.I. lab have contributed to the BORIS project. Without their efforts BORIS would still be sitting on the launching pad. I'm pleased here to be able to acknowledge each in turn:

Tom Wolf was central in the design and implementation of the most recent version of BORIS, and made important contributions to BORIS parsing integration and question answering. Pete Johnson was central in earlier versions of BORIS and made important contributions to BORIS memory and processing structures. No one could ask for better co-programmers and co-researchers than Tom and Pete.

Mark Burstein also made major contributions to earlier versions of BORIS, and was involved in the design of MOP overlays and event

assimilation. Marty Korsin, CJ Yang, Lewis Johnson, and Mark Tanner worked a great deal on various BORIS parsers. Margot Flowers provided a number of LISP utilities for BORIS, including output and file access functions.

Steve Harley, Mark Tanner, and Judith Lempert worked on the English generator for BORIS. This generator is based upon code from Rod McGuire's generator. Hence Rod served as the ultimate advisor on all issues regarding English generation.

A number of researchers in psychology have also contributed to the BORIS project. Colleen Seifert (under the guidance of Professors McKoon, Ratcliffe, and Abelson) designed and ran a number of significant, novel experiments on BORIS memory structures, specifically TAUs. Scott Robertson conducted experiments on memory modifications during question answering. Brian Reiser worked on narrative summarization and Plot Units in BORIS. Steve Minton designed and ran a number of experiments on interpersonal structures in BORIS, while Jill Fain experimented with scenario maps. I also want to thank Ira Roseman, who has conducted a number of significant experiments on emotions, and who was kind enough to read and offer useful comments concerning my chapter on the role of AFFECT in BORIS. Andrew Ortony (at the Center for the Study of Reading) was also kind enough to read and offer suggestions on a final draft of the AFFECT chapter.

Others were particularly helpful in my first years: Dr. Chris Riesbeck taught me how to parse. Professors Drew McDermott and Dave Barstow expanded my A.I. horizons outside of natural language processing. Discussions with Professor Mallory Selfridge and with Larry Birnbaum on parsing were very worthwhile and served as a source of much inspiration.

Among those who have helped make my stay at the Yale A.I. Lab both interesting and memorable are: Valerie Abbott, Bill Bain, Prof. John Black, Prof. Jaime Carbonell, Greg Collins, Ernie Davis, Natalie Dehn, Prof. Jerry DeJong, Ann Drinan, Konrad Dyer, Glen Edelson, Bill Ferguson, Patti Oronzo, Linda Fusco, Jim Galambos, Tom Galloway, Dr. Anatole Gershman, Dr. Ray Gibbs, Prof. Rick Granger, Abraham Gutman, Chris Hammond, Dr. Shoshanna Hardt, Ed Hovy, Dr. Dave Johnson, Prof. Janet Kolodner, Prof. Mike Lebowitz, John Leddo, Stan Letovsky, Jeff Lowney, Steve Lytinen, Beth Marshburn, Judy Martel, Kay Parady, Tony Passera, Rich

Plevin, John Ruttenburg, Steve Salzburg, Udi Shapiro, Dr. Steve Shwartz, Steven Slade, Prof. Elliot Soloway, and Michael Wilk.

One reason the Yale A.I. Lab is a great place to work is that it receives tremendous systems support from the Yale Tools group. This group is lead by John O'Donnell, who is the best facilities director anyone could ask for. Other systems "wizards" I've relied on, who have created numerous utilities, and who I would like to thank, are: Steve Wood, who created the nifty Z screen editor; John Ellis, who created MUF (Multiple User Forks) and SM (the Session Manager); Nat Mishkin, who expanded the capabilities of Yale TLISP and built the windowing package, and Bob Nix, who contributed to Z and other tools. The type fonts for this thesis result from Bob's work on a scribe-driver for the Yale laser printer.

In addition, I want to thank a number of individuals outside of Yale, who have, at one time or another in the past, had a great influence on me. They are: Prof. Neil Jones, Prof. Henry Lundsgaarde, Prof. Steven Muchnick, Prof. Hans Penner, Prof. Jerome Rothstein, Prof. Henry Selby, and Prof. Doug Tuggle.

Finally, this work was supported in part by the Advanced Research Projects Agency under contract NOO14-75-C-1111 and in part by the National Science Foundation under contract IST7918463.

Table of Contents

Preface	-xiii-
Acknowledgments	-xv-
1 The Meaning of In-Depth Understanding	1
1.1 BORIS -- A Computer Program	1
1.2 What BORIS Is Up Against	5
1.3 Knowledge and Memory for Comprehension	9
1.4 Natural Language Processing: Some Background	11
1.4.1 Early Semantics-Based Approaches	11
1.4.2 Script-Based Understanding	13
1.4.3 Goal-Based Understanding	14
1.4.4 Text Skimmers	15
1.5 In-Depth Understanding	16
1.6 Methodology, Scope, and Aims	18
1.7 A Guide to the Reader	23
 PART I. RECOGNIZING NARRATIVE THEMES	 25
 2 Thematic Abstraction Units	 27
2.1 Introduction	27
2.2 Thematic Abstraction Units	29
2.3 Bad Planning is Widespread	32
2.4 TAU Implementation	35
2.4.1 TAU-DIRE-STRAITS	35
2.4.2 TAU-CLOSE-CALL	37
2.4.3 TAU-REG-MISTAKE	39
2.4.4 TAU-RED-HANDED	41
2.4.5 TAU-HIDDEN-BLESSING	43
2.4.6 Searching TAUs During QA	44
2.5 Recognizing TAUs: An Overview	45

2.5.1 Tracking the Reasons Plans Fail	48
2.5.2 Capturing the Point of a Story	51
2.6 TAU Experiments	52
2.7 Comparison With Related Work On High Level Structures	54
2.7.1 TAUs and TOPs	55
2.7.2 TAUs and Meta-Planning	57
2.7.3 TAUs and Plot-Units	58
2.8 Very Large Thematic Patterns	60
2.9 Conclusions	62
3 Recognizing TAUs	65
3.1 Introduction	65
3.2 Activating The Recognition Process	68
3.2.1 Goal Failures Activate TAUs	68
3.2.2 Expectation Failures Activate TAUs	70
3.2.3 Planning Choices Activate TAUs	72
3.3 A TAU Categorization Based on Planning Metrics	75
3.3.1 ENABLEMENT	78
3.3.2 COST	83
3.3.3 EFFICACY	84
3.3.4 RISK	85
3.3.5 COORDINATION	88
3.3.6 AVAILABILITY	89
3.3.7 LEGITIMACY	92
3.3.8 AFFECT	92
3.3.9 SKILL	94
3.3.10 VULNERABILITY	95
3.3.11 LIABILITY	97
3.4 TAU Interrelationships	97
3.5 Planning Metric Tradeoffs	99
3.6 The Point of Planning Metrics	100
3.7 Conclusions	101
4 The Role of AFFECT in Narratives	103
4.1 Introduction	103
4.2 Representing Emotions	105
4.3 Affective Connections	110

4.4 AFFECTs in Processing	112
4.4.1 Cognition and Arousal	112
4.4.2 AFFECT Intensity and Modality	116
4.4.3 AFFECTs and Norm Deviations	117
4.5 Affective Empathy and Interpersonal Themes	118
4.6 ACEs	120
4.7 AFFECTs and Episodic Memory	124
4.7.1 AFFECTs Help Built TAUs	126
4.7.2 Answering AFFECT Questions	127
4.8 Comparison With Other Work	130
4.8.1 Roseman's Model	131
4.8.2 Scope of BORIS AFFECT Model	135
4.9 Conclusions	136

PART II. PROCESS INTEGRATION AND MEMORY INTERACTIONS 141

5 Integrated Processing with a Unified Parser	143
5.1 Introduction	143
5.2 Background	143
5.3 Integrating Parsing With Memory Search	144
5.4 Parser Unification	148
5.4.1 Event Explanation	151
5.4.2 Parsing in Different Modes	154
5.5 Memory Modification During QA	157
5.6 Memory Reconstruction in BORIS	161
5.7 Conclusions	162
6 The Process of Comprehension	165
6.1 Introduction	165
6.2 Demons and Processing Control	165
6.3 Knowledge Interaction Demons	170
6.3.1 Knowledge Dependencies	170
6.3.2 Origin of the Knowledge Dependency Graph	176
6.4 Task Oriented Demons	178

6.4.1 Role Binding and Role Checking	178
6.4.2 Memory Instantiations	178
6.4.3 Word Sense Disambiguation	180
6.4.4 Reference	182
6.4.5 The Lexicon: An Example	183
6.4.6 Theoretical Perspective	188
6.4.7 Reinterpretation Processes and MOP Application	190
6.4.8 Recognizing Deviations	192
6.5 Summary	195

PART III. REPRESENTING AND ORGANIZING KNOWLEDGE 197

7 Intentionality and MOPs	199
7.1 Introduction	199
7.2 I-links and Episodic Memory	199
7.3 I-link Interpretation	205
7.4 Packaging Knowledge with I-links	207
7.5 Using I-links to Search Memory and Answer Questions	208
7.5.1 Choosing an Appropriate Retrieval Heuristic	209
7.5.2 Selecting an Appropriate Answer	212
7.6 Intentionality: Related Work	215
7.7 Conclusions	216
8 Memory Overlays with MOPs	217
8.1 Introduction	217
8.2 Scripts in Review	217
8.3 Multiple Perspectives with MOPs	219
8.4 Processing Deviations	223
8.5 Service Triangles	226
8.6 MOP Implementation in BORIS	228
8.6.1 Contracts	229
8.6.2 Legal Disputes	236
8.6.3 The Role of Lawyer	238
8.6.4 Legal Reasoning	241

8.7 Conclusions	254
9 A Spatial/Temporal Organization for Narratives	255
9.1 Introduction	255
9.2 Scenario Participant Map	256
9.3 Main Scenarios	259
9.4 Transition Scenarios	263
9.4.1 Recognizing Transitions	264
9.4.2 Accessing Transition Scenarios	265
9.5 Mental Scenarios	267
9.5.1 Recognizing Mental Scenarios	268
9.5.2 Accessing Mental Scenarios	269
9.6 Inferring Knowledge States	270
9.7 Inferring Temporal Relationships	273
9.8 Conclusions	276
10 The Interpersonal Dimension	277
10.1 Principles of Interpersonal Interaction	277
10.2 IP-FAVOR	279
10.2.1 Returning Favors	285
10.2.2 Violating Favors	286
10.3 IP-COMM	288
10.4 IP-SOC and Interpersonal Stages	291
10.5 Conclusions	294
PART IV. SYNTHESIS -- Putting It All Together	297
11 Narrative Comprehension: A Detailed Example	299
11.1 Introduction	299
11.2 Caveats	300
11.3 Annotated Narrative Trace	301
11.3.1 PARAGRAPH-1	302
11.3.2 PARAGRAPH-2	315
11.3.3 PARAGRAPH-3	318
11.3.4 PARAGRAPH-4	325
11.3.5 PARAGRAPH-5	333

11.3.6 PARAGRAPH-6	336
11.4 Question-Answering Traces	338
11.4.1 Event Specification Questions	339
11.4.2 Affective Reaction Questions	341
11.4.3 Causal Antecedent Questions	343
11.4.4 Presupposition Violations	346
11.4.5 Empathy Related Questions	348
11.5 English Generation	349
11.6 BORIS Implementation Specs	351
12 Future Work and Conclusions	353
12.1 Review	353
12.2 Was Building BORIS Worth It?	354
12.3 Future Research on TAUs	356
12.4 Future Work on Processing Integration	362
12.5 Final Retrospection	365
I. APPENDIX: Narratives and Sample Q/A	367
I.1 Narrative: KIDNAP-0	367
I.2 Question-Answering: KIDNAP-0	367
I.3 Narrative: DIVORCE-1	369
I.4 Question-Answering: DIVORCE-1	370
I.5 Narrative: DIVORCE-2	372
I.6 Question-Answering: DIVORCE-2	373
I.7 Major DIVORCE Knowledge Structures	376
II. APPENDIX: CD, Goals, Plans, Scripts	379
II.1 CD: Primitive ACTs	379
II.2 Goals and Plans	382
II.3 Scripts	383
III. APPENDIX: McDYPAR -- A Demon Parser	385
III.1 Introduction	385
III.2 An Overview of McDYPAR	385
III.2.1 An Informal Lexicon	386
III.2.2 A Parse Result in Working Memory	388
III.2.3 Demon Syntax and Semantics	389
III.2.4 Demon Trees	395

III.3 Background	397
III.4 A Detailed Trace, With Associated Lexicon and Demons	401
III.4.1 Execution Trace	401
III.4.2 Sample McDYPAR Lexicon	407
III.4.3 Associated Demon Definitions	409
III.5 McDYPAR As Implemented In TLISP	411
III.5.1 Word Tasks	414
III.5.2 Code To Handle Declarative Notation For Word Senses	415
III.5.3 Demon Management	417
III.5.4 Demon Interpreter Functions	418
III.5.5 Functions to Spawn Demons	420
III.5.6 Functions to Define Words and Demons	421
III.5.7 Standard Utility Functions Used By Demons	422
III.5.8 Search Utilities	425
III.5.9 Tracing Functions	426
III.6 Exercises	
References	437
Index	449

CHAPTER 1

The Meaning of In-Depth Understanding

1.1 BORIS -- A Computer Program

The theory of narrative comprehension presented here is implemented within BORIS, a computer program. BORIS is able to answer questions about a narrative in order to demonstrate that it has read and understood it. Thus BORIS also incorporates a theory of memory search and retrieval for question answering.

What follows is an actual understanding and question answering session with the BORIS program running on a narrative concerning a divorce (called DIVORCE-1). Input to the program is in lower case; output in uppercase. BORIS receives the English shown here as verbatim input and produces verbatim English below as output:

I/O Behavior on DIVORCE-1

*(BORIS 'DIVORCE-1)

PROCESSING PARAGRAPH 1:

Richard hadn't heard from his college roommate Paul for years. Richard had borrowed money from Paul which was never paid back. but now he had no idea where to find his old friend. When a letter finally arrived from San Francisco, Richard was anxious to find out how Paul was.

READY TO ACCEPT Q/A:

What happened to Richard at home?
RICHARD GOT A LETTER FROM PAUL.

Who is Paul?
RICHARD 'S FRIEND.

2 *In-Depth Understanding*

Did Richard want to see Paul?
YES, RICHARD WANTED TO KNOW HOW PAUL WAS.

Had Paul helped Richard?
YES, PAUL LENT MONEY TO RICHARD.

Why didn't Richard pay Paul back?
RICHARD DID NOT KNOW WHERE PAUL WAS.

How did Richard feel when the letter appeared?
RICHARD FELT GLAD BECAUSE PAUL AND HE WERE FRIENDS.

PROCESSING PARAGRAPH 2:

Unfortunately, the news was not good. Paul's wife Sarah wanted a divorce. She also wanted the car, the house, the children, and alimony. Paul wanted the divorce, but he didn't want to see Sarah walk off with everything he had. His salary from the state school system was very small. Not knowing who to turn to, he was hoping for a favor from the only lawyer he knew. Paul gave his home phone number in case Richard felt he could help.

READY TO ACCEPT Q/A:

What was the letter about?
PAUL AND SARAH WERE GETTING A DIVORCE.

Did Paul write Sarah?
NO, IT WASN'T SARAH, IT WAS RICHARD WHO GOT THE LETTER.

Why was Paul upset about the divorce?
PAUL AND SARAH WERE FIGHTING OVER THE FAMILY POSSESSIONS.

What did Paul do for a living?
PAUL WAS A TEACHER.

What did Sarah want?
SARAH WANTED TO HAVE THE CAR AND THE HOUSE AND
THE KIDS AND THE ALIMONY.

Why did Paul write to Richard?
PAUL WANTED RICHARD TO BE HIS LAWYER.

PROCESSING PARAGRAPH 3:

Richard eagerly picked up the phone and dialed.
After a brief conversation, Paul agreed to have
lunch with him the next day. He sounded extremely
relieved and grateful.

READY TO ACCEPT Q/A:

Why did Richard call Paul?
RICHARD WANTED TO ARRANGE A MEETING WITH PAUL.

Did Richard contact Paul?
YES. RICHARD CALLED PAUL.

How did Richard contact Paul?
BY PHONE.

How did Paul feel when Richard called?
PAUL WAS HAPPY BECAUSE RICHARD AGREED TO BE PAUL'S
LAWYER.