Text generation
Studies in Natural Language Processing

This series publishes monographs, texts, and edited volumes within the interdisciplinary field of computational linguistics. Sponsored by the Association for Computational Linguistics, the series will represent the range of topics of concern to the scholar's working in this increasingly important field, whether their background is in formal linguistics, psycholinguistics, cognitive psychology, or artificial intelligence.
Text generation

Using discourse strategies and focus constraints to generate natural language text

KATHLEEN R. MCKEOWN
Department of Computer Science, Columbia University
Contents

Preface
1. Introduction
   1.1 Problems in generating text
   1.2 A processing model
   1.3 A sketch of related work
   1.4 A text generation theory and method
   1.5 System overview
   1.6 The database application
   1.7 Other issues
   1.8 Guide to remaining chapters
2. Discourse Structure
   2.1 Rhetorical predicates
      2.1.1 Linguistic background
      2.1.2 Ordering communicative techniques
   2.2 Analysis of texts
      2.2.1 Predicate recursiveness
      2.2.2 Summary of text analysis
   2.3 Related research using rhetorical predicates
   2.4 Use of schemata
      2.4.1 Associating technique with purpose
   2.5 Selecting a schema
   2.6 Filling the schema
   2.7 An example
   2.8 Future work
   2.9 Conclusions
3. Focusing in discourse
   3.1 Computational theories and uses of focusing
      3.1.1 Global focus
      3.1.2 Immediate focus
   3.2 Focusing and generation
      3.2.1 Global focus and generation
      3.2.2 Immediate focus and generation
   3.3.1 Current focus versus potential focus list
   3.2.4 Current focus versus focus stack
   3.2.5 Other choices

3.2.6 A focus algorithm for generation 69
3.2.7 Selecting a default focus 69
3.2.8 Overriding the default focus 70
3.2.9 The focus algorithm 71
3.2.10 Use of focus sets 73
3.3 Focus and syntactic structures 75
3.3.1 Linguistic background 75
3.3.2 Passing focus information to the tactical component 77
3.4 Future work 79
3.5 Conclusions 80

4. TEXT system implementation 83
4.1 System components 84
4.2 Knowledge representation 87
4.2.1 Representation overview 87
4.2.2 Portability 90
4.2.3 Summary 91
4.2.4 The entity-relationship model 92
4.2.5 Use of generalization 93
4.2.6 The topic hierarchy 97
4.2.7 Relations 100
4.2.8 Distinguishing descriptive attributes 102
4.2.9 DDAs for database entity generalizations 103
4.2.10 Supporting database attributes 104
4.2.11 Based database attributes 106
4.2.12 DDAs for database entity subsets 109
4.2.13 Constant database attributes 111
4.3 Selection of relevant knowledge 113
4.3.1 Requests for information and definitions 113
4.3.2 Comparisons 114
4.3.3 Determining closeness 114
4.3.4 Relevancy on the basis of conceptual closeness 116
4.3.5 Conclusions 121
4.4 Schema implementation 122
4.4.1 Arc types 122
4.4.2 Arc actions 123
4.4.3 Registers 123
4.4.4 Graphs used 124
4.4.5 Traversing the graph 124
4.4.6 The compare and contrast schema 130
4.5 The tactical component 133
4.5.1 Overview of functional grammar 133
4.5.2 The grammar formalism 134
4.5.3 A functional grammar 138
4.5.4 The unifier 140
4.5.5 The TEXT system unifier 141
4.5.6 Unifying a sample input with a sample grammar 144
4.5.7 Grammar implementation 147
4.5.8 Morphology and linearization 151
4.5.9 Extensions 152
4.5.10 Advantages 152
4.5.11 Disadvantages 153

6. Related generation research 186
6.1 Tactical components - early systems 186
6.2 Tactical components - later works 187
6.3 Generation in database systems 188
6.4 Planning and generation 189
6.5 Knowledge needed for generation 190
6.6 Text generation 190

7. Summary and conclusions 196
7.1 Discourse structure 195
7.2 Relevancy criterion 195
7.3 Discourse coherency 195
7.4 Generality of generation principles 196
7.5 An evaluation of the generated text 197
7.6 Limitations of the implemented system 199
7.7 Future directions 200
7.7.1 Discourse structure 200
7.7.2 Relevancy 203
7.7.3 Coherency 203
7.7.4 User model 204
7.7.5 Conclusion 204

Appendix A. Sample output of the TEXT system 206
Appendix B. Introduction to Working 223
Appendix C. Resources used 225
Appendix D. Predicate Semantics 227
Bibliography 237
Index 244
Preface

There are two major aspects of computer-based text generation: 1) determining the content and textual shape of what is to be said, and 2) transforming that message into natural language. Emphasis in this research has been on a computational solution to the questions of what to say and how to organize it effectively. A generation method was developed and implemented in a system called TEXT that uses principles of discourse structure, discourse coherency, and relevancy criterion. In this book, the theoretical basis of the generation method and the use of the theory within the computer system TEXT are described.

The main theoretical results have been on the effect of discourse structure and focus constraints on the generation process. A computational treatment of rhetorical devices has been developed which is used to guide the generation process. Previous work on focus of attention has been extended for the task of generation to provide constraints on what to say next. The use of these two interacting mechanisms constitutes a departure from earlier generation systems. The approach taken here is that the generation process should not simply trace the knowledge representation to produce text. Instead, communicative strategies people are familiar with are used to effectively convey information. This means that the same information may be described in different ways on different occasions.

The main features of the generation method developed for the TEXT strategic component include 1) selection of relevant information for the answer, 2) the pairing of rhetorical techniques for communication (such as analogy) with discourse purposes (for example, providing definitions) and 3) a focusing mechanism. Rhetorical techniques, which encode aspects of discourse structure, are used to guide the selection of propositions from a relevant knowledge pool. The focusing mechanism aids in the organization of the message by constraining the selection of information to be talked about next to that which ties in with the previous discourse in an appropriate way.

This work on generation has been done within the framework of a natural language interface to a database system. The implemented system generates responses of paragraph length to questions about database structure. Three classes of questions have been considered: questions about information available in the database, requests for definitions, and questions about the differences between database entities.
Text generation

The work described in this book was done at the University of Pennsylvania and would not have been possible without the help of a number of people who deserve special mention. First and foremost, is the influence of my advisor, Aravind K. Joshi, who provided many of the insights and much appreciated guidance throughout all stages of the work. I am also grateful to Bonnie Webber for her many helpful suggestions, pointers to relevant papers, and editorial comments. The implementation of TEXT was greatly assisted by Kathleen McCoy and Steven Boese who designed and implemented portions of the system. Kathy developed a system which automatically generated a portion of the knowledge base and implemented the knowledge base interface. Steve designed and partially implemented the tactical component used in TEXT.

Many people read and commented on various sections of the manuscript. Barbara Groez's comments on the chapter on focusing were particularly valuable. Norman Badler, Peter Buneman, Richard Korf, Michael Lebowitz, Eric Mays, Kevin Matthews, Cecile Paris, and Ellen Prince also contributed in this area. Finally, the detailed and insightful comments of the unnamed reviewers must be mentioned as these were extremely helpful in improving the text.

Support for this work was provided in part by an IBM Research Fellowship, by National Science Foundation grant #MCS81-07290 awarded to the Department of Computer and Information Science of the University of Pennsylvania, by ONR grant N00014-82-K-0256 awarded to the Department of Computer Science of Columbia University, and by ARPA contract N00039-82-C-0427 awarded to the Department of Computer Science of Columbia University.