THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 6, July–August 1983
Printed in U.S.A.

Human Factors and Behavioral Science:

The *UNIX*™ Writer's Workbench Software: Philosophy

By L. T. FRASE*

(Manuscript received December 17, 1981)

Technology has dramatically increased the number and complexity of written documents. The Bell System spends over $100,000,000 annually on technical documents, with much effort devoted to review and revision. The *UNIX*™ Writer's Workbench programs assist documentation by automating copy editing and proofreading tasks. These programs deliver detailed measures and comments about text readability, punctuation, word use, abstractness, and other features. Authors use the system to evaluate draft documents, and many feel the programs improve their writing skills. The programs have been used for quality control and text research, and in writing courses. This paper discusses principles used in system development, introduces the two following papers, and suggests possible uses for automated language analysis aids.

I. INTRODUCTION

Massive growth in the number of documents and in the sophistication needed to understand technical language has created challenges for many organizations, outside and within the Bell System. The Naval Air Systems Command, for instance, supplies technical manuals

* Bell Laboratories.

for 135 aircraft. There are over 25,000 manuals, totaling 3,000,000 pages. In 1950, the manuals for one aircraft contained fewer than 2,000 pages; today, the manuals for one aircraft contain nearly 300,000 pages.[1] In the Bell System, there are over 100 general categories of documents. Just one of these categories, the Bell System Practices, contains 400 broad document categories, with 35,000 active titles. Current issues are routinely distributed to 45,000 locations. But number of pages is not the only concern. In some states, if documents are relevant to the physical or financial well-being of individuals, laws dictate how difficult the language of a text may be. Recent "plain language" legislation, passed by six states, allows courts to rewrite texts that are unclear.[2]

The telecommunications industry, along with others, is faced with two substantial problems: (1) how to manage a large number of documents, and (2) how to make them understandable. Review and evaluation of technical documents is costly, time-consuming, subjective, and often tedious. Mechanized aids for reviewing documents, although not relieving humans of responsibility for document quality, can simplify and speed the documentation process. Efforts elsewhere in Bell Laboratories address the problem of managing a large number of documents.[3,4] The *UNIX** Writer's Workbench software was designed to help improve quality.

Until recently, document technology has consisted of two parallel, but separate, lines of work. The first consists of experimental research on what makes text easy or difficult to understand. This work, reflected in psychological and linguistic research, includes studies of the effects of text format, wording, and graphic displays.[5] The second line of work includes more intuitive or prescriptive bases for writing and text design. This involves the expertise of text and graphics designers,[6] such as technical editors and layout specialists, as well as rhetorical traditions that specify writing standards.[7] Both knowledge bases, one derived from experimental research and the other from nonexperimental sources, provide useful guidelines for effective text design.

Unfortunately, text researchers and designers have not always worked closely together. Psychological theories of what makes a document difficult to understand often seem oversimplified to designers, and designers' intuitions and prescriptions often seem vague to researchers. Designers have been concerned with creating good documents, while researchers have been trying to understand problems of text comprehension.

The use of computers for language analysis is beginning to stimulate

---

* Trademark of Bell Laboratories.

integration of technical knowledge from research and design areas. McMahon, Cherry, and Morris,[8] for instance, have shown how language analysis, such as vocabulary analysis or assigning parts of speech to words, can be done easily and accurately by computer. Humans find those tasks difficult and tedious. Models of decision processes used by experts have been incorporated into programs that make design recommendations, such as the most suitable column width to use for a particular text.[9] Knowledge-based expert systems,[10] developed by researchers in artificial intelligence, provide additional ways of expressing, in computer-readable language, expertise derived from different disciplines. Development of the Writer's Workbench system has required, and continues to encourage, this contact between different disciplines. The programs recognize the validity of principles derived from research and the validity of principles derived from intuition and practical experience. Hence, they help bring research and application closer together.

The Writer's Workbench system provides a simple set of commands that deliver many assessments needed in documentation work. These include editorial comments on punctuation, word use, spelling, and text abstractness, along with analyses of grammatical parts of speech and calculations of overall text readability (expressed as reading grade levels). The programs combine work done by the Computing Science Research Center and the Human Performance Engineering Department of Bell Laboratories.

Using routines already developed for *UNIX* software, such as spelling and parts of speech analysis programs, developers were able to build programs that expanded and went beyond existing facilities. For portability, the wide distribution of *UNIX* operating systems in Bell Laboratories was an advantage. An early version of the programs was tried on several *UNIX* systems for four months before the first general release within Bell Laboratories. During those months, program efficiency, outputs, and options were modified in response to user feedback. Although current versions of the programs incorporate further enhancements and pruning of esoteric programs, the system architecture has remained the same.

Text to be analyzed must be stored in the computer. Some authors prefer to compose text on line; others have it entered by clerical staff. Text should contain standard formatting commands, for instance, commands that define headings and other special elements. To analyze a text, the user logs in to the system and issues a command name followed by the name of the file that contains the text. The program then executes, prints analyses, and perhaps suggests changes to improve the text. Currently, the programs are appropriate for proofreading and stylistic analyses of expository and descriptive prose, but not

for procedural documents consisting entirely of lists or abbreviated step-by-step directions.

The Writer's Workbench system is limited by the text features that the programs can recognize, and also by our understanding of the relation of text features to reading difficulty. For instance, one program recognizes when the first word in a sentence is not capitalized; it is more difficult to develop a program that can recognize when important content is missing. Currently, the programs do not use linguistic parsing algorithms; they cannot "understand" the subject and object of a sentence, for example, as humans would. Nevertheless, the richness and variety of feedback delivered by the programs are different enough from commonly encountered word and text processing systems that we find it useful to refer to these expanded capabilities as "language processing" functions. Hence, we have used this terminology here and in the following two papers, perhaps as much to remind ourselves of what remains to be done as to characterize the current programs.

II. DESIGN PRINCIPLES

Six specific principles, reviewed below, guided Writer's Workbench program development. The principles provide a context for the detailed program descriptions in the paper that follows this.[11] To summarize, the principles assume that language analysis programs should be rational, diverse, evaluative, modifiable, specific, and informative.

2.1 Expert knowledge base

The Writer's Workbench programs are *rational* in the sense that they are based on research or expert consensus that justifies the relevance of program information for text comprehension or use. Some program outputs are based on psychological research, showing, for instance, that passive sentences are more difficult to understand than active sentences.[11] Other programs are based on advice from writing and style experts.[7,12] For instance, standard punctuation usage is evaluated by some programs. Knowledge about the features of good writing is scattered among disciplines; hence, no single discipline is entirely adequate for formulating language analysis programs. Currently, we have drawn from rhetorical and psychological literature to provide text assessments that people commonly use in document evaluation. To do more, contributions from artificial intelligence and related fields are required. In addition, since not all features of document design can be clearly supported by research or expert opinion,[13] programs contain messages directing users to information sources that explain certain evaluations.

2.2 Program diversity

Written communication is so complex that single measures, such as an overall index of text readability, can draw an author's attention away from other important text characteristics. Therefore, we wanted to provide *diverse* measures so users would think about a constellation of features. For instance, one program delivers over 40 measures of text characteristics, some measuring word characteristics, others measuring sentence characteristics.

We assume that authors have diverse needs, which cannot be foreseen entirely. Hence, the Writer's Workbench system contains many programs and options that can be used, or not, as an author desires. The diversity of programs works as a structured system to let users explore characteristics of words (such as syllables), sentences (such as length in words), or paragraphs (such as beginning and ending sentences).

2.3 Relative judgments

A text is good or bad depending on whether it is adequate for a particular task or audience. *Evaluative* judgments about a document usually relate to some standard of excellence. The Writer's Workbench system makes such comparisons explicit. Measures from one document are compared to measures obtained from other documents. For instance, evaluative comments about the readability of a text can be based on statistics derived from training or technical memoranda judged to be of high quality. The author is warned when the text being analyzed departs significantly from the standard documents. Ability to compare documents to various standards emulates an important component of human text judgments, and it is a distinctive feature of the Writer's Workbench system.

2.4 User modifications

Since it is not possible to anticipate the documentation needs of all users, programs should be *modifiable*. Three aspects of the programs can be adjusted by the user—standards, input, and output. For instance, one program calculates statistics for variables derived from texts supplied by an author. These statistics can then be used as standards to evaluate features of the author's subsequent texts, such as percentage of passive sentences. Other programs allow adding or deleting words or phrases that are detected by spelling and word usage dictionaries. In addition, input can be modified to include or exclude lists from analyses, and output format and length can be changed by appending simple descriptors to commands.

2.5 Text location

For authors to review exactly where language may have gone awry in a text, *specific* text locales must be identified. For instance, printing an average measure of sentence length is less useful for an author than printing sentences that exceed a specified length. Early work on identifying troublesome text locales for a reader was done by Koether and Coke.[14] Command options in the Writer's Workbench system provide summary statistics or review sentences that violate certain standards, such as length or word use. In the latter case, line numbers show where offending text can be found. This level of specificity is necessary if programs are to speed text revision.

2.6 Information resources

The Writer's Workbench system is intended to be *informative* rather than to make rigid decisions. Text statistics should be interpreted cautiously, and we suspect they will be used most effectively when an author can get advice related to those statistics. The Writer's Workbench system provides guidance, using information files, about word use, punctuation, and other features. Brief explanatory comments are also given in programs, with information about how to obtain more information.

III. RATIONALE AND DESIGN

"*UNIX*™ Writer's Workbench Software: Rationale and Design,"[11] describes program architecture, and discusses at length the literature behind programs and their options. Current literature on writing is rich in descriptions of text patterns that characterize faulty expression,[15] and contains descriptions of actions that a writer can take to correct those faults. These patterns and actions, which are the bases for program measures and recommended revisions, are described in the first part of the next paper. Next, the individual programs and their options are described. The programs are divided into those that proofread, those that comment on style, and those that comment on organization. Finally, features that help people use the programs—human factors issues—are surveyed.

IV. RESULTS OF A FIELD STUDY

The Writer's Workbench system has been used in Bell Laboratories for over three years; currently it runs on over 100 *UNIX* systems. Bell Laboratories users have responded favorably to the programs, and many believe the programs improve writing skills.[16]

Of course, there are many environments in which language analysis programs might be used. "*UNIX*™ Writer's Workbench Software:

Results of a Field Study,"[17] describes data collected from program trials conducted outside Bell Laboratories. The two outside locations included one human factors organization and one technical writing organization. The groups were engaged in different writing tasks and the staffs had different computer experience.

According to the field study, authors liked the detailed computer suggestions about how to revise their texts, and they were better able to detect errors with program help than without it.

V. CONCLUSIONS

The Writer's Workbench programs do not do everything humans do when evaluating text. They cannot determine whether important content is missing, nor do they assess format features or the meaning of texts, as humans would. Nevertheless, the programs speed editing, and they help detect features that authors might otherwise miss.

Language analysis programs have many uses. Components of the Writer's Workbench system have been used for research; for instance, to derive general quantitative descriptions of text style and to relate these styles to text difficulty.[18] They have also been used, by various groups at Bell Laboratories, to help develop technical documents for new software systems and for writing technical memoranda. Programs could be used to obtain management information, for example, to monitor changes in documents across time, and to determine the match between skills of readers and the texts they must read.[19] Finally, the programs can be used for instruction to provide detailed feedback to students about the characteristics of their writing, or by an instructor to evaluate the effects of instruction on student writing.

The Writer's Workbench programs have recently been used as an adjunct to a technical writing course sponsored by the Bell Laboratories Education Center. Such programs could be especially useful to authors whose native language is other than English. Use of language analysis programs in research, development, management, and instruction will no doubt grow as these programs increase in diversity and sophistication.

VI. ACKNOWLEDGMENTS

Programs created by L. L. Cherry contributed much to the development of the Writer's Workbench system. She also helped with program enhancements. W. Vesterman, of Rutgers University, stimulated early work on stylistic and diction programs. S. A. Keenan made important contributions to program development and documentation, as well as to Writer's Workbench system trials. J. L. Collymore, M. L. Fox, and M. F. Poller also contributed to development, and exchanges with W. F. Fox and J. J. Dever were helpful.

REFERENCES

1. W. G. Muller, "Useability Research in the Navy," in T. G. Sticht and D. Zapf (Eds.), *Reading and Readability Research in the Armed Services*, Alexandria, VA.: Human Resources Research Organization, 1976.
2. "Plain Language Laws," in "Simply Stated," Newsletter of the Document Design Center, Washington, D.C.: American Institutes for Research, No. 18(July, 1981), p. 1.
3. M. J. Rochkind, "The Source Code Control System," IEEE Trans. Software Eng., *se-1*, No. 4(December 1975), pp. 364–70.
4. M. H. Bianchi, R. J. Glushko, and J. R. Mashey, "A Software/Documentation Development Environment Built from the UNIX Toolkit," Int. Fed. Inform. Processing, Working Conf. Automated Tools Inform. Syst. Design and Development (January 1982), New Orleans, LA.
5. J. Hartley, "Eighty Ways of Improving Instructional Text," IEEE Trans. Professional Commun., *PC-24*, No. 1(March 1981), pp. 17–27.
6. E. S. Ferguson, "The Mind's Eye: Nonverbal Thought in Technology," Science, *197*, No. 4306(August 1977), pp. 827–36.
7. R. A. Lanham, *Revising Prose*, New York: Charles Scribner's Sons, 1979.
8. L. E. McMahon, L. L. Cherry and R. Morris, "UNIX Time-Sharing System: Statistical Text Processing," B.S.T.J., *57*, No. 6(July–August 1978), pp. 2137–54.
9. S. A. Keenan, "Computer Projections of the Cognitive Effects of Text Changes," Amer. Educ. Res. Assn., Boston, MA, April 7–11, 1980.
10. R. O. Duda and J. G. Gaschnig, "Knowledge-Based Expert Systems Come of Age," Byte, *6*, No. 9(September 1981), pp. 238–81.
11. N. H. Macdonald, "*UNIX*™ Writer's Workbench Software: Rationale and Design," B.S.T.J., this issue.
12. W. Strunk, Jr. and E. B. White, *The Elements of Style*, New York: Macmillan, 1972.
13. L. T. Frase, "Ethics of Imperfect Measures," IEEE Trans. Professional Commun., *PC-24*, No. 1(March 1981), pp. 48–50.
14. M. E. Koether and E. U. Coke, "A Scheme for Text Analysis Using FORTRAN," Amer. Educ. Res. Assn., New Orleans, LA, February 25–March 1, 1973.
15. L. Flower, *Problem-Solving Strategies for Writing*, New York: Harcourt Brace Jovanovich, Inc., 1981.
16. L. T. Frase, N. H. Macdonald, P. S. Gingrich, S. A. Keenan and J. L. Collymore, "Computer Aids for Text Assessment and Writing Instruction," N.S.P.I. Journal, (November 1981), pp. 21–4.
17. P. S. Gingrich, "*UNIX*™ Writer's Workbench Software: Results of a Field Study," B.S.T.J., this issue.
18. E. U. Coke, unpublished work.
19. E. U. Coke and M. E. Koether, "A Study of the Match Between Technical Document Difficulty and the Reading Skills of Technical Personnel," B.S.T.J., this issue.

AUTHOR

Lawrence T. Frase, A.B. (Philosophy), 1960, University of Miami; M.S., 1963, Ph.D. (Educational Psychology), 1965, University of Illinois; University of Massachusetts (Psychology Department), 1965–1967; Bell Laboratories, 1967–1976; National Institute of Education, 1967–1968; Bell Laboratories, 1978—. Mr. Frase has published research on human reasoning, reading, writing, and document design. While on leave from Bell Laboratories, he was head of the Learning Division at the National Institute of Education in Washington, D.C. Currently he is working on applications of computers to human information processing. He is a fellow of the American Psychological Association.