RESEARCH IN
WORD PROCESSING
NEWSLETTER

Dr. Bradford A. Morgan and Dr. James M. Schwartz, Editors
South Dakota School of Mines and Technology Rapid City, SD 57701 Phone: (605) 394-2481

Vol. 2 No. 6 September 1984

Evaluating Student Papers With a Word Processor

Abstract

A professor with just a standalone word processor can evaluate stacks of student papers more efficiently and thoroughly by using the power of the computer to insert and format boilerplate comments onto a student-specific, paper-specific, but professional-looking evaluation report. In addition, a writer's progress from paper to paper can be better monitored by scrolling successive comment sheets.

Introduction

Just as word processing is revolutionizing the way students write, interactive text analysis will dramatically change the role of faculty in academic writing programs in the decades ahead. Students will interactively brainstorm, compose, and revise on a screen. Then text-analysis programs will automatically analyze various stylistic variables (most of which are quantifiable) and suggest alternatives when a problem is discovered.

For each stylistic feature, the student will be able to measure his response against a norm. For example, a computer can easily locate misspelled words for writers, telling them to look words up in the dictionary. Can't such analysis be done before a paper is turned in? Can't the professor's time be used more professionally than circling spelling errors?

But most academic writing programs are just beginning to awaken to the promise of student word processing and text analysis—and haven't yet figured out how to finance expensive hardware and software systems. Even so, most professors can personally afford a standalone word processor. This paper describes how just such a standalone can be used to assist faculty with a major activity: evaluating stacks of student papers. First, let's look at the familiar manual method:

Manual Evaluation of Papers

Before the word processor, most professors graded papers manually. The familiar method of "marking up" a paper to deliver evaluative feedback is familiar to most teachers of writing: phrase-type comments are jotted in the margins, words are circled, arrows are drawn, and the like. Even double-spaced text puts awkward physical constraints upon the comment itself. Only the briefest, most general "label" is possible in the majority of cases. But is that enough for the less prepared student who might not know how to continue?

And what do students see when they look at their returned papers? Most often they see ugly, black, often illegible scribblings everywhere. The more "dedicated" the evaluator, the greater the confusion. For the student, the initial impression may seem like a personal attack on ideas painstakingly, even enthusiastically, developed. Would an artist tolerate a similar marking up of a canvas? Jumping to the conclusion that *all* of the comments are probably negative, some students fail to read them at all. (In contrast, the proverbial good student will overcome any barrier to intercept the professor's advice.)

2--Research in Word Processing Newsletter

Even before the word processor, many researchers advocated that comments be made primarily on a separate cover sheet, a professional-looking report of the evaluator's assessment. But faced with sheer volume—stacks of papers—teachers with pencils or typewriters could only generate limited commentary.

Using Text-Coded Symbols in Papers

Using boilerplate comments generated by a word processor is not far removed from another popular paper-evaluating mechanism. Many publishers of grammar texts provide professors with handy refer-to-text symbols to be placed in the margins of papers, sending the student to appropriate text sections for extra study. Such coding has the advantage of abbreviated effort for the evaluator—two or three characters, perhaps—but a full theory-example treatment for the student.

Despite its merits, such an approach can only achieve so much. After all, the student is fully aware of the "quantity" of the professor's contribution—just the coded symbol. Compare this to a boilerplate insertion that has been personally articulated by the evaluator himself: the sheer volume of response suggests a stronger professional interest in the student's work. Instead of three words, what about three sentences or more? Where to look in the grammar text for help becomes just *one* of many elements of the boilerplate comment; other advice, tailored to the professor's own way of putting things, can be easily built into the larger comment (including tutorials, if the text is not enough).

Then, too, refer-to-text coding sometimes fails to specifically pinpoint particular problems for students. Having located the right section in the text, some students find too extensive a treatment—and aren't sure which part of the discussion fits the professor's intentions. Perhaps ony ten percent of the text section appropriately corresponds to the problem flagged in the paper—but which part some students may ask? While it is certainly a useful exercise to have students read all of the comments, there are also advantages to a more specific match between problem and solution.

Using Computer Power to Evaluate Papers

Those familiar with processing stacks of student papers will agree that the bulk of their individual evaluative responses actually have relevance to not one, but many different students. Take spelling, for example. If several students in a class need to be told to approach spelling more systematically, why should the same comment be rephrased several different times? Why not just say it once? Haven't we always tended to give the *same* advice to different students if they share the *same* problem? Isn't such consistency almost necessary to ensure objective standards?

A particular response needs to be articulated just once, then saved. It can be recalled again and again—just like information on a rubber stamp—and delivered as needed to various students who deserve the same comment. Think of the time wasted manually penning or typing the same comments year after year. Why not let the computer do what it can do best—save time and conserve labor? The repetition in paper grading is tremendous; the feedback is fragmentary and often grudgingly undertaken.

With the word processor, an extended comment can be called up with a keystroke or two, inserted into an evaluative commentary, and then printed automatically. Such a practice ultimately makes more professional use of teachers' time, allowing them to concentrate on evaluative concerns beyond the reach of the computer.

The Evaluation Sheet

A professional-looking evaluation sheet can be arranged and printed in minutes. The evaluation sheet can be variously fashioned, but a consistent format is desirable for successive sets of papers (just as it is for a memo). The following approach might serve as an example, each step accomplished with a keystroke or two by using the "insert" mode found on most word processors:

- Develop a masthead with the name of the school, course, date, or whatever.
- Call in comments one at a time, allowing the computer to adjust the new text to the margins but stopping two spaces after the final period to receive the insertion of yet another comment. Manually inserted items can also be included whenever necessary, denoting the student's first name or some other personal response fitting only that paper or student.

- Use boilerplate insertions to direct students to relevant paragraphs or pages in their papers. After a boilerplate comment on the comma splice problem, for example, you might want to add the following comment. "See your fifth paragraph for a good example." The comma-splice-problem comment is called up by keying-in "CS"; the refer-to-paper locator is summoned by keying in "P5" in this case. P1 stands for the first paragraph, P2 for the second, and so on. Similar codes can direct students to pages rather than paragraphs.

Catalog of Comments

What do you want to say to a student? Time has to be invested creating a catalog of comments which reflects your range of critical feedback: what's good, what's bad, and what needs extension and consistency. In fact, *any* comment that can be made orally can be committed to the catalog under a mnemonic file label. You merely need to type the response onto the screen, then save it.

The range of comments is seemingly unlimited: where to check in the grammar or rhetoric text for corrective advice, how to strengthen a rhetorical strategy, a short-cut way of conceptualizing a problem—anything worth saying. Entries can be as long or as short as necessary. Earlier versions of a comment can be gradually reinforced later (revising is simple on the word processor).

In addition, the same comment can be phrased in different ways, adjusting for tone and other variables worth including. If you don't want to make the same comment (about spelling, for example) to two students who are friends and might compare evaluation sheets, then *two* different comments will help.

Personalizing Comments

Special care should be taken to make comments seem fresh and original to the student. When articulating comments for the catalog, avoid anything that sounds like it came from a computer—or from a textbook. Any teacher of writing should have little difficulty adjusting tone through the use of personal pronouns, contractions, and otherwise, informal ways of putting things. Then, too, the student's full name and paper title should appear in the masthead, with the first name scattered here and there on the evaluation sheet. The boilerplate insertions can also be stopped now and then to key-in a sentence or two that is unmistakably connected with that particular paper and student.

To avoid a "form" response, it is also advisable to never give the same student the same comment on successive papers. If a similar comment must be given (because a problem wasn't successfully corrected or acted upon), then simply boilerplate "follow-up" comments—just as you would otherwise.

Program to Monitor Previous Comments for a Particular Student

Monitoring a writer's progress from paper to paper is essential. Under the manual method of evaluating papers, how efficiently was this accomplished? Brute memory was probably the most common method, but how effective was this when a professor had a hundred or more students, each of whom was submitting ten papers over the course of a semester? With a word processor, successive comment sheets can be scrolled in their entirety before evaluating a new paper. Better yet, a program can be easily written to have the computer convert the boilerplate block back to its original coded label. The following procedure serves as an example of how to accomplish this:

1. Reverse code-to-text process to save valuable disk space. (Manually keyed-in comments can be attached).

   A program can be easily written to make the computer "remember" the evaluator's actual keystrokes—"sp" for example—but not the much-longer inserted text.

2. Alphabetize coded "comments" for each paper to allow for progress monitoring of each successive paper.

   Alphabetizing comments allows for quick scanning. Paper-to-paper development can be assessed at a glance.

3. A "collected" set of comments from all papers prevents previously given comments from being added again. A prompt comes on when you attempt to give an already-given comment to the same student. A similar program can be written to insert the student's name at the end of selected paragraphs as well.

Mnemonic Catalog Should Allow Fast Access

Gathering comments from the catalog can be more effective when natural mnemonics become so personalized that further indexing is minimized. A simple system should be developed using the word processor's ability to sort letters, numbers, and symbols—that same alphanumeric logic which orders page labels and other routine filings. Ideally, comment labels should be brief to minimize keystrokes and make sense mnemonically (such as *sp* for spelling) to avoid unnecessary time referencing lists. The best systems will allow easy retrieval and merging of evaluative comments.

Once the primary subject category has been mnemonically identified—*av* for adverb, for example—numerals can systematically identify the specific approach of the comment as follows:

av1	good use
av2	used incorrectly
av2r	still incorrect
av3	used inconsistently
av3r	still inconsistent
av4	missing; use next time
av4r	still missing
av5	special consideration
av6	other comment

It is clear that the first two letters—*av*—can be changed infinitely to accommodate other categories of evaluative concerns. As an option, the letter *r* might be used as follow up reinforcement, acknowledging that the student has not acted upon the earlier-given advice.

SOUTH DAKOTA SCHOOL OF MINES AND TECHNOLOGY
English 101: Composition

JOHN DOE

Title: "Video Games: The Pros and Cons"

This paper on computer games contains interesting insights and valuable information, and the flow of expression is clear and readable. Video games seem to be an engaging subject to write about; they certainly seem to be propelling the computer revolution forward, John.

You establish a good sense of common ground at the beginning, winning the reader over without letting him throw up a defensive barrier. This is a good example of inductive reasoning at work.

But you need to work for economy by removing deadwood and other redundant matter. Weigh the sheer information content of a statement and then trim away unnecessary or assumed information, i.e., information which has already been established or is being repeated. More subordination will help in some cases, and repetition can also be eliminated by switching to synonyms. See the example I've pointed to on your third page.

I notice that you don't tend to use the colon, a useful workshop tool for any writer. Next time, try it out once or twice; you'll like it. Just keep in mind that a full IC comes before it. The word or phrase following it becomes highlighted and emphasized. See p. 177 in *Harbrace* for some examples, John.

In addition, study p. 132 in *Harbrace*. Before the main clause (subject-verb) begins, you need to add a comma if there are introductory phrases or clauses. Don't worry about one prepositional phrase, but two or more should be followed by a comma. This helps the reader to process your idea groupings more efficiently. Take a look at your third paragraph for an example.

Good, you pencilled-in some of your own corrections. Continue to do so! Proofreading and troubleshooting communicate conscientiousness and will pay off in the long run.

John, all in all, this paper demonstrates a good ability to weigh pros and cons, thus letting the reader know that you're not a one-sided thinker who, therefore, might simply be unaware of the real strengths of the other point of view. You do this, in part, by successfully scanning your reader's "counter" arguments for possible objections.

B A "B" this time, but you will need to study the comments I've made—and master the bulk of them—to earn the same or higher grade next time. You can do it!

Figure 1: Example of Evaluation Sheet Using Boilerplate Comments

Boilerplate: jd

SOUTH DAKOTA SCHOOL OF MINES AND TECHNOLOGY
English 101: Composition

JOHN DOE

Title:

Boilerplate: o1

This paper on * contains interesting insights and valuable information, and the flow of expression is clear and readable. [The * causes the word processor to stop for an additional phrase insertion.]

Boilerplate: m1

Video games seem to be an engaging subject to write about; they certainly seem to be propelling the computer revolution forward.

Boilerplate: arg1

You establish a good sense of common ground at the beginning, winning the reader over without letting him throw up a defensive barrier. This is a good example of inductive reasoning at work.

Boilerplate: dw2

But you need to work for economy by removing deadwood and other redundant matter. Weigh the sheer information content of a statement and then trim away unnecessary or assumed information, i.e., information which has already been established or is being repeated. More subordination will help in some cases, and repetition can also be eliminated by switching to synonyms.

Boilerplate: ex3

See the example I've pointed to on your third page.

Boilerplate: cl4

I notice that you don't tend to use the colon, a useful workshop tool for any writer. Next time, try it out once or twice; you'll like it. Just keep in mind that a full IC comes before it. The word or phrase following it becomes highlighted and emphasized. See p. 177 in *Harbrace* for some examples.

Boilerplate: cm2

In addition, study p. 132 in *Harbrace*. Before the main clause (subject-verb) begins, you need to add a comma if there are introductory phrases or clauses. Don't worry about one prepositional phrase, but two or more should be followed by a comma. This helps the reader to process your idea groupings more efficiently.

Boilerplate: p3

Take a look at your third paragraph for an example.

Boilerplate: ty1

Good, you pencilled-in some of your own corrections. Continue to do so! Proofreading and troubleshooting communicate conscientiousness and will pay off in the long run.

Boilerplate: arg5

All in all, this paper demonstrates a good ability to weigh pros and cons, thus letting the reader know that you're not a one-sided thinker who, therefore, might simply be unaware of the real strengths of the other point of view. You do this, in part, by successfully scanning your reader's "counter" arguments for possible objections.

Boilerplate: g2

BA "B" this time, but you will need to study the comments I've made—and master the bulk of them—to earn the same or higher grade next time. You can do it!

Figure 2: Keystrokes Needed To Insert Boilerplate in Sample Evaluation Sheet

Conclusion

Correctly applied, boilerplate-driven evaluations of student papers can allow a teacher to engage more completely and specifically with individual student papers. Increased efficiency of the evaluator's time and effort may well reverse the demoralized outlook of those in our disciplines who face the sometimes soul-deadening prospect of processing yet another stack of student papers. Faced with sheer volume, we will never afford the time to offer an "ideal" set of comments to students—the kind that might be possible if we had just one student, for example. We are not Marcus Welbys whose practice is limited to one or two patients, allowing us to minister to even the most trivial of their needs. We can, however, harness the power of the word processor to serve our traditional goals in evaluating papers—letting the computer do what it can do best and freeing our time for more professional service.

[Ed. Note: Portions of the above were presented at Villanova University's conference, "The Written Word and the Word Processor," February 25, 1984.]

Call For Papers

"Theory and Reality: the Ideal Writing Center" is the theme of the Southeastern Writing Center Association's 1985 national conference to be held in Atlanta next April 18-20, 1985. Double-spaced manuscripts on the use of computers in writing centers must be submitted by December 1st. Contact Dr. Dabney Hart, English Department, Georgia State University, Atlanta, GA 30303.

Bibliography Update

Arca, Julie Ann. "Boilerplate Paragraphs." *Practical WordStar Uses.* Berkeley: Sybex, 1983, pp. 85-110.

Becker, Joseph D. "Multilingual Word Processing." *Scientific American.* 251:1 (July 1984), pp. 96-107.

Bolter, J. David. "Electronic Language." *Turing's Man: Western Culture in the Computer Age.* Chapel Hill: Univ. of North Carolina Press, 1984, pp. 124-150.

Chandler, Daniel, ed. *Exploring English with Microcomputers.* London: Council for Educational Technology, 1983.

Conrad, Lora P. and Louis W. Bender. *Word Processing and Microcomputers in Small Two-Year Colleges: A National Study.* Tallahassee: Florida State University (Institute for Higher Education), 1984.

Cullen, Charles T. "The Word Processor and Scholarly Editions." *A Grin on the Interface: Word Processing for the Academic Humanist.* ed. Alan T. McKenzie. New York: The Modern Language Association of America, 1984, pp. 39-48.

Dyer, Michael George. "In-depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension." Unpublished Dissertation. Yale University, 1982.

Freyburger, Janice Anne. "A Computer System for Understanding English Text in a Kindergarten Story." Unpublished Thesis. Western Michigan University, 1982.

Gabler, Hans Walter. "Computer-Aided Critical Edition of Ulysses." *Association for Literary and Linguistic Computing Bulletin.* 8:3 (1981), pp. 232-248.

Gigley, Helen Mueller. "Neurolinguistically Constrained Simulation of Sentence Comprehension: Integrating Artificial Intelligence and Brain Theory." Unpublished Dissertation. University of Massachusetts, 1982.

Green, John O. "Computers, Kids and Writing: An Interview with Donald Graves." *Classroom Computer Learning.* 4:8 (March 1984), pp. 20-23, 28.

Grover, Mark Donald. "A Synthetic Approach to the Representation and Processing of Temporal Phenomena of English." Unpublished Dissertation. Northwestern University, 1982.

Haselkorn, Mark P. and Jim S. Borck. "The Word Processor and the Scholarly Journal." *A Grin on the Interface: Word Processing for the Academic Humanist.* ed. Alan T. McKenzie. New York: The Modern Language Association of America, 1984, pp. 49-58.

Heilbrun, Carolyn. "Some Words for the Trepid." *A Grin on the Interface: Word Processing for the Academic Humanist.* ed. Alan T. McKenzie. New York: The Modern Language Association of America, 1984, pp. 1-9.

Hill, Jane Anne Collins. "A Computational Model of Language Acquisition in the Two-year-old." Unpublished Dissertation. University of Massachusetts, 1982.

Kiefer, Kathleen E. and Charles R. Smith. "Textual Analysis with Computers: Tests of Bell Laboratories' Computer Software." *Research in the Teaching of English.* 17:3 (October 1983), pp. 201-214.

Leahy, Ellen K. "A Writing Teacher's Shopping and Reading List for Software." *English Journal.* 73 (January 1984), pp. 62-65.

Mace, Scott. "Novelists Inspire Games: Science-Fiction Authors Lend Their Talents to 'Interactive Fiction' ". *InfoWorld.* 6:25 (June 18, 1984), pp. 42-45.

McKenzie, Alan T. "The Word Processor and the Work Habits of the Academic Humanist." *A Grin on the Interface: Word Processing for the Academic Humanist.* ed. Alan T. McKenzie. New York: The Modern Language Association of America, 1984, pp. 10-24.

-------. "The Word Processor and the Typing Chores of the Academic Department." *A Grin on the Interface: Word Processing for the Academic Humanist.* ed. Alan T. McKenzie. New York: The Modern Language Association of America, 1984, pp. 25-38.

Miller, Lance A. et al. "Text-Critiquing with the EPISTLE System: An Author's Aid to Better Syntax." *Proceedings.* National Computer Conference, May 4-7, 1981. (Arlington, VA: AFIPS Press, 1981), pp. 649-655.

Modern Language Association. "Multilingual Word-Processing Equipment." *MLA Newsletter.* 16:1 (Spring 1984), p. 15.

Modern Language Association. "Update on Multilingual Word-Processing." *MLA Newsletter.* 16:2 (Summer 1984), pp. 13-14.

Nancarrow, Paula Reed, Donald Ross, and Lillian Bridwell. Eds. *Word Processors and the Writing Process: An Annotated Bibliography.* Westport, CT: Greenwood Press, 1984.

Ohanian, Susan. "IBM's 'Writing To Read' Program: Hot New Item or Same Old Stew?" *Classroom Computer Learning.* 4:8 (March 1984), pp. 30-33.

Oliver, Lawrence J. "Monster, Not Mentor: A Comment on 'Monsters and Mentors.' " *College English.* 46 (April 1984), pp. 410-416 (with response by Helen J. Schwartz).

Ritchie, Graeme D. *Computational Grammar: An Artificial Intelligence Approach to Linguistic Description.* New York: Barnes and Noble, 1980.

Rogers, Raymond A. "Pinyin for the PC: Designing a Chinese Word Processor Is No Easy Task." *PC World.* 2:6 (June 1984), pp. 285-289.

Schwartz, Helen J. "Teaching Writing with Computer Aids." *College English.* 46 (March 1984), p. 239-245.

Scotchmoor, Judith. "Order Out of Chaos: Making a Mess Out of Stories Can Actually Help Your Students Learn Word Processing and Sentence-Sequencing Skills." *Classroom Computer Learning.* 4:8 (March 1984], p. 69.

Selfe, Cynthia L. "Wordsworth II: New Wave CAI for College Composition Teachers." [ED 225 151] November, 1982. 15pp.

Smith, Charles R. and Kathleen E. Kiefer. "Using the Writer's Workbench Programs at Colorado State University." *International Conference on Computers and the Humanities.* eds. Sarah K. Burton and Douglas D. Short. Rockville, Maryland: Computer Science Press, 1983, pp. 672-684.

Smith, Donald Edwin. "Focuser: A Strategic Interaction Paradigm for Language Acquisition." Unpublished Dissertation. Rutgers University, 1982.

Smith, John B. "Computer Criticism." *Style.* 12 (Fall 1978), pp. 326-356.

Swartz, Theodore F. "Finding Funding for Your Computer Project." *Classroom Computer Learning.* 4:8 (March 1984), pp. 36-41.

Wanderman, Richard. "Bridge to Clarity: The Computer As Compensatory Writing Tool." [dyslexia] *Softalk for the IBM Personal Computer.* 3 (June 1984), pp. 25-30.

White, Ron. "Writing to Read: 25,000 Kids Help IBM Test Unconventional Learning Theory." *Softalk for the IBM PC*. 3 (June 1984), pp. 76-82.

Young, Robert Lee. "Explorations in Story Understanding Using Contextually Accessed Semantic Networks." Unpublished Dissertation. The University of Texas at Austin, 1982.

SOFTWARE REVIEW - *Textra*

The newsletter does evaluations of word-processing software to help you discover programs that might fulfill your and your students' writing or research needs. This month, we evaluate *Textra*, an inexpensive yet easy to use and powerful word-processing program from Ann Arbor Software.

When reviewing a word-processing package, we are not endorsing any product. Rather, we are describing the software's strengths and weaknesses and examining how these features (or lack of them) might affect students and teachers in academic writing situations.

PROGRAM:	*Textra*
PUBLISHER:	Ann Arbor Software
ADDRESS:	5407 N. Main, Ann Arbor, MI 48104
LIST PRICE:	$95.00
WILL RUN ON:	IBM-PC, XT, and compatibles
MEMORY (RAM) NEEDED:	128k
DISK DRIVES NEEDED:	one
SPELLING DICTIONARY:	not included with program
CORRECTION METHOD:	n/a
ON-DISK TUTORIAL:	yes (excellent)
QUALITY OF MANUAL:	good

COMPOSITION

FEATURES	**YES-NO**	**COMMENT**
Help screens (video tutorials you can call on during editing)	yes	you may ask for help at anytime during an editing session by pressing the "Escape" key and then the function key you want to learn more about
Automatic headers (titles), footers, and page numbers	yes	the program allows for flush left, right, and centered information as well as "floating" (i.e., for titles that vary on odd and even pages]
Full-screen cursor control (ability to move to any spot on the screen to edit)	yes	
Automatic word wrap (no "Return" required at the end of each typed line)	yes	
Adjustable left and right margin settings (e.g., for indenting extended single-spaced quotations)	yes	some word-processing programs force you to compose documents in a predetermined format; *Textra* does not
Single and double spacing	yes	
Automatic text adjusting after making editorial changes (insert/delete)	yes	you have the option either to reformat manually after editorial changes (with the F4 key) or to choose automatic reformatting, which reformats text as you type (but which also slows down the editing process)

View your text on the screen as it will appear on paper after printing	yes	while you can see how it will look, you cannot make any changes to your text when in this mode
Search for and/or replace words	yes	you can both search for and replace a word or a phrase up to 40 characters in length
Move text from one location to another in a document	yes	you can both copy (or duplicate) and move (or cut/paste) blocks of words

LITERATURE

Superscripting (...the end.''[12])	no	for research papers requiring references

CREATIVE WRITING

Ability to space lines in less than full increments (quarter- and half-line spacing)	no	important when the appearance of a document complements content (i.e., in concrete poetry)
Proportional spacing	yes	when your text is fully justified, *Textra* automatically inserts spaces between words (other programs use "microjustification," or true proportional spacing, inserting spaces of 1/120" between individual letters)
Right-justified text (text lines up on the right margin)	yes	

TECHNICAL WRITING

Subscripting (H_2O)	no	
Graphics	no	enables you to create flowcharts, graphs, schematics, and other illustrations (to *Textra's* credit, very few word-processing programs currently on the market allow for mixing graphics and text)

PROFESSIONAL

Create your own "HELP" screens	no	with this feature, you can design your own on-disk grammar, rhetoric, and usage tutorials that your students may retrieve while composing or editing their papers
Text merging (also referred to as "boilerplating")	no	with this feature, you can create, store, and recall frequently used citations (as in the case of a computerized composition evaluaton program—see related article in this issue)
Background printing (printing one document while simultaneously editing another)	no	while *Textra* does not come with this feature, it is compatible with the many print spooler programs and hardware units now on the market

OTHER FEATURES

There are only a few minor irritations with *Textra*. In addition to the lack of subscript and superscript support, it does not have the ability to list a directory of your textfiles. Essentially, this means that you have to remember the exact title of your file when *Textra* prompts you for it or else it's no go! The good folks at Ann Arbor Software have

assured us that directory support will be forthcoming, as will other editing and formatting enhancements. Although *Textra* does not come equipped with its own spelling dictionary, programs such as *The Word Plus* [Oasis Systems, 2765 Reynard Way, San Diego, CA 92103], *The Random House Proofreader* [dist. by Digital Marketing Software, 2363 Boulevard Circle, Walnut Creek, CA 94595], and other spelling checker or text analysis programs which read plain ASCII (American Standard Code for Information Interchange) textfiles should work quite well.

OVERALL EVALUATION

Don't let *Textra's* diminutive price fool you. Students and teachers from high school to graduate school will find the software an easy to learn, easy to use package with features that rival more expensive word-processing programs. If you have any problems understanding the instructions after you purchase *Textra,* the folks at Ann Arbor Software are there with answers quicker than you can say "nanosecond." With a variety of screen-display formats (including color) from which to choose, *Textra* works hard to keep your eyes interested in your words (creative content, however, still must be "colored" by your own imagination). With the addition of directory assistance, superscript and subscript support, merge capabilities, and (possibly) a spelling dictionary of at least 50,000 words, *Textra* would be comparable with word-processing programs at four times the price. At under $100, *Textra* does shine but be sure that either you don't need—or can wait for—the formatting and disk handling features not currently in the program's repertoire before purchasing it.

[ED. NOTE: The categories we include in our software reviews reflect course offerings found in academic settings. If you feel we should add other categories that address common writing initiatives, or if you would like to see more program features included under existing categories, let us know.]

Manuscript Submissions Welcome

The newsletter welcomes article submissions from our readers which pertain to word-processing applications in academic writing. Manuscripts should be OCR readable (Courier, Letter Gothic, or similar letter-quality typefaces) or may be submitted on disk using WordStar or standard ASCII code in IBM-PC DOS (5¼" diskette; 1.1, 2.0, or 2.1) or CPT 8500 (8" disk) formats (direct uploading of articles via modem will be available soon). All manuscripts should include a short autobiographical sketch. The Editors reserve the right to edit articles, if necessary. If you want your manuscript returned, please enclose a stamped, self-addressed envelope with your submission. Address all correspondence to the Editors, *Research in Word Processing Newsletter,* Liberal Arts Department, South Dakota School of Mines and Technology, Rapid City, SD 57701-3995.

Research in Word Processing Newsletter. Volume 2, Number 6. Copyright © 1984 by the South Dakota School of Mines and Technology. All rights reserved. ISSN: 0748-5484. *Research in Word Processing Newsletter* is published 9 times a year (September through May) by the South Dakota School of Mines and Technology, Rapid City, South Dakota 57701-3995; telephone (605) 394-2481. Postage paid at Rapid City, South Dakota. **SUBSCRIPTIONS:** $12 per year (US); $18 per year (Canada); $24 per year (foreign). Address all subscription inquiries to the Editors, *Research in Word Processing Newsletter,* South Dakota School of Mines and Technology, Rapid City, SD 57701-3995. Please allow four to six weeks for processing.