THE UNIVERSITY OF CALGARY

A COMPARISON OF METHODS FOR MARKING STRAIGHT COPY TYPING EXERCISES

by

KENNETH J. LUTERBACH

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF ARTS

DEPARTMENT OF CURRICULUM AND INSTRUCTION

CALGARY, ALBERTA

FEBRUARY, 1988

ISBN 0-315-42421-4

THE UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled, "A Comparison of Methods for Marking Straight Copy Typing Exercises" submitted by Kenneth Luterbach in partial fulfillment of the requirements for the degree of Master of Arts.

Supervisor, Dr. Erv Schieman
Department of Curriculum
and Instruction

Dr. W. Bruce Clark
Department of Curriculum
and Instruction

Prof. Warren Rowley
Department of Music

Date: Feb. 9, 1988

Abstract

A software driven marking algorithm for straight copy typing exercises, called the WoBaT Marker (**Wo**rd **Ba**sed **T**yping Marker), was developed and tested. One test was conducted to show that the WoBaT Marker corrects text generated during straight copy typing exercises more accurately than experienced human markers. A second test was conducted to show that the WoBaT Marker corrects text generated during straight copy typing exercises faster than human markers.

To show that the WoBaT Marker corrects text more accurately than experienced human markers, 35 student typists enrolled in *Typewriting 20* at a local high school generated text during three distinct straight copy typing exercises. The instructions to the typists were varied to promote the production of typing errors. If the typists had made few or no errors, results from the tests would not have been useful to test the error recognition capabilities of the system in question.

The text generated by the student typists was subsequently marked by the WoBaT Marker and six experienced human markers. The experienced human markers, on average, had taught typewriting for 14 years. The least experienced typewriting teacher had taught typewriting for 8 years whereas the most experienced typewriting teacher had taught typewriting for 20 years. To control for the various methods through which the experienced markers learned to score text, six inexperienced human markers also scored the text generated by the student typists. All of the inexperienced markers were trained to score text generated during straight copy typing exercises by the investigator.

To show that the WoBaT Marker corrects text produced during straight copy typing exercises faster than human markers, timing data were

collected while the text was scored. To obtain the timing data the human markers timed themselves with stop watches and the WoBaT Marker was timed by the clocks in the computers used by the student typists.

In addition to the error tally and timing data, questionnaire data were also obtained. The questionnaire data were collected to determine the attitudes of the experienced markers toward the WoBaT Marker.

To analyze the error tally data the investigator compared the error tallies generated by the human markers and the WoBaT Marker to the actual error tallies. The actual error tallies were determined by pooling the errors cited by the experienced markers. Implicit in that approach is the notion that every typing error was discovered by at least one experienced marker. The error tally data were compared through the use of t-tests. The t-tests revealed that a significant difference existed between the actual error tallies and those generated by each of the inexperienced and experienced human marker groups. However, no significant difference was found between the actual error tallies and the WoBaT Marker error tallies. As a result it was concluded that the WoBaT Marker scores text more accurately than human markers.

Analysis of the timing data showed that the WoBaT Marker scored the text approximately 249 times faster than the human markers. Lastly, the questionnaire data revealed that the experienced markers held a generally favourable attitude toward the WoBaT Marker.

Acknowledgements

I wish to thank especially my supervisor, Dr. Erv Schieman, and Dr. W. Bruce Clark.

I also want to thank Business Educators, Mr. R.S. Ash, Mr. Larry Colquhoun, Mr. Tom Cormack, Mrs. Marilyn Dalquist, Miss Christine Eberhardt, Mrs. Judy Gajdos, Miss Helen Koch, Mr. Ed Lucotch, Mr. Jerome McDonald, Mrs. Mary McMaster, Mrs. Wendy Read and Mr. Herb Zimmer.

I wish to acknowledge the University of Calgary for supporting this work with a Thesis Research Grant.

To my Parents

Table of Contents

CHAPTER Page

List of Tables

List of Figures

Chapter 1

INTRODUCTION

Background to the Problem

Over the past three years Business Education departments in Alberta high schools have acquired many microcomputers by taking part in the Senior High School Business Education Equipment Upgrading Project. The upgrading project was instituted by Alberta Education to help schools meet the equipment requirements of the new Senior High School Business Education program and to facilitate the introduction of new technology (Alberta Education, 1985).

The upgrading project was funded by the government of Alberta and local school jurisdictions on a 50-50 cost shared basis. Essentially, then, the government of Alberta matched the funds spent by school jurisdictions on computers and related hardware destined for Business Education departments. The government did set a $365.00 per Business Education student ceiling. Nevertheless this provincial funding incentive was well received. Indeed, in 1986, approximately 60% of all microcomputers in senior high schools in Alberta were located in Business Education departments (Alberta Education, 1986).

To be sure, the mass infusion of microcomputers into Business Education departments in just three years caused several problems, all too familiar to Business Educators. Even though some of the problems have been solved, results from a survey distributed to Business Education Coordinators throughout Alberta in October 1986 (Business Education Council, 1987) identified the following categories as areas of particular concern:

Software Problems
Inservice Needs
Articulation of Courses/Grades
Hardware
Typing/Keyboarding

With respect to the aforementioned topics this study pertains to "Software Problems" and "Typing/Keyboarding."

According to Schmidt (1983, p. 1), "*keyboarding* differs from typing in the basic purpose it serves." Schmidt also stated: "Keyboarding is the act of entering alphanumeric data on a keyboard of information processing equipment for the purpose of obtaining or communicating information." Even though this study involved, as will become evident, data entry into microcomputers, the more familiar term "typewriting" will be used, rather than "keyboarding" or the cumbersome word pair "typewriting/keyboarding."

Statement of the Problem

The microcomputers in Business Education departments are being used, in part, to teach students how to type. Clearly, this is not surprising since the primary input device of computers is still a keyboard. In addition, since microcomputers operate at very high speeds it is possible, even with a minimum of programming experience and effort, to attain accurate information about a typist's speed. Despite these advantages the investigator has not encountered a commercially produced typewriting program with an accurate marking algorithm. (A detailed review of typewriting software appears in the second section of Chapter 2.) The

Marking Algorithm component in typewriting software is composed of the routines that are used to correct text entered by learners during straight copy typing exercises.

Typewriting marking algorithms are of particular interest to the writer for two reasons. First, valid assessment of learner performance is contingent upon an accurate marking algorithm and few such algorithms exist. Since inaccurate assessments of learner performances ultimately lead to the presentation of incorrect feedback, a significant problem exists. After all, feedback is vitally important to learners. With respect to feedback, Coon (1986, p. 213) made the following two points.

1. Feedback is particularly important in human learning.
2. The value of feedback is one of the most useful lessons to be gained from studies of learning.

De Cecco and Crawford (in West, 1983, p. 23) related the importance of feedback more directly when they stated that "feedback is the single most important variable governing the acquisition of skills. . . [and] is perhaps one of the most dependable and thoroughly tested principles in modern psychology." Note, however, that feedback mechanisms will not be the major focus of this thesis.

Software-driven marking algorithms are also of interest because they are typically much faster than human markers. Consequently, if a marking algorithm existed that could correct text produced during straight copy typing exercises as accurately as an experienced human marker, it would reduce teacher marking time.

For this study a software-driven marking algorithm that corrects text produced during straight copy typing exercises was developed. The marking algorithm used in this study differed from earlier marking algorithms in two critical ways. First, it marked text on a *per word* basis whereas earlier typing marking algorithms corrected text on a *per character* basis. When text is marked on a *per character* basis each character entered by a user is compared to an expected character from an original document. On the other hand when text is marked on a *per word* basis each word entered by a user is compared to an expected word from an original document.

Sample original and user-entered text for a straight copy typing exercise is shown in Figure 1. If the user-entered text in Figure 1 is marked on a *per character* basis, 16 errors would be logged because each of the user-entered characters at and beyond the "s" in "tres" is out of place. On the other hand, 1 error would be logged if the user-entered text were marked on a *per word* basis. Using *per word* marking, it would have been discovered that "tres " is not "trees " and therefore just one error should be logged.

According to the rules for marking straight copy typing exercises, which can be found in nearly all typewriting textbooks (see, for example, Ubelacker, Guest & McConaghy, 1983), only one error should be logged per word at most. (More rules for marking straight copy typing exercises are described in the first section of Chapter 2.) Although the example above was contrived to depict the striking difference between the two marking methods it should be clear that *character based* marking algorithms nearly always produce grossly inflated error tallies when used to correct straight copy typing exercises.

The marking algorithm in the program developed for this study will be called the "WoBaT Marker" since it is a **Wo**rd **Ba**sed **T**yping **Marker**. The WoBaT Marker also differed from earlier marking algorithms in that it compensated for multiple word omissions and/or repetitions. Typewriting marking algorithms that cannot compensate for omitted and repeated words also produce inflated error tallies. (Examples depicting how those marking algorithms inflate error tallies are provided in the second section of

Figure 1. Sample original and user-entered text for a straight copy typing exercise.

Original Text:	In autumn, the leaves on many trees change colour.
User-entered Text:	In autumn, the leaves on many tres change colour.

Chapter 2.)

This thesis will attempt to obtain objective evidence so that typing teachers may rely on its results. This is in keeping with the following statement directed to teachers by Leonard West (1983, p. 13): "Professional behavior requires reliance on reputable objective evidence, not on mere subjective impressions, personal hunches or collections of opinions."

In contrast, the approach to this thesis will be quite unlike that of Faborn Etier who, after consulting with colleagues, deemed that the electronic typewriting teaching aid would reduce typewriting instruction time by one half. In fact, Etier (1971, p. 24) stated: "By using the electronic typewriting teaching aid, it is believed by those closely associated with its use in teaching typing that as much can be accomplished in one semester as in two semesters using the conventional teaching method."

To begin this quest for reputable objective evidence the researcher advanced the following two hypotheses:

1. The WoBaT Marker corrects text produced during straight copy typing exercises more accurately than experienced human markers.

2. The WoBaT Marker corrects text produced during straight copy typing exercises in less time than human markers.

To test the first hypothesis, human markers and the WoBaT Marker corrected paragraphs generated by student typists. The marking processes were timed in order to test the second hypothesis.

Importance of the Study

This study was conducted to show that the WoBaT Marker is superior to human markers with respect to speed and accuracy. If this result could be demonstrated, it would likely lead to a significant reduction in the time typing teachers currently spend marking text produced during straight copy typing exercises. Also, if the marking algorithm were used in classrooms, students would receive feedback about their typing performance immediately after copying a textual document. In this way students could learn to type more quickly than students presently enrolled in typing classes.

This study is of particular importance to Business Educators in Alberta since it addresses two of five areas of specific concern to them, as noted previously. Indirectly, this study is also of concern to educators in general because it addresses computer software in an educational setting. In fact, responses to a survey, gathered on behalf of Alberta Education to determine the extent of educational computing in this province, demonstrated that "The single problem which 55.8% of the respondents identified to be of high importance was procuring and maintaining software" (Alberta Education, 1986).

With respect to further programming endeavours it was hoped that insights gained during this study would lead to more efficient typewriting software. Given an accurate marking algorithm, it is conceivable that typewriting software of the future could analyze the types of errors made by a user. Then the software could display suitable standard, or even custom-made, exercises that the user could practice typing. Irrespective of the extent to which future typewriting programs may be able to diagnose learning difficulties and prescribe learning materials, at this point more efficient software could be developed by increasing the speed and accuracy of the marking algorithm used in this study.

Scope and Limitations

This study is pertinent to all local high school typing teachers. Even though the teacher/markers involved in this study were not selected randomly, they possessed marking skills similar to those of other local typing teachers. Further, although only grade eleven typists were used to generate text from straight copy typing exercises, the errors they made can be considered typical of all typists. In fact, West (1983, p. 129 & 153) reported that error analysis studies have consistently shown that typing errors are randomly distributed. On average, then, no particular group of typists is prone to making more or certain types of errors.

Explanation of Terms

In this study the explanations for the typewriting terms in Table 1 and computing terms in Table 2 are relevant.

Table 1

<u>Typewriting Terms</u>

Term	Explanation
Word	In a typewriting framework the term "word" must be understood in terms of composition. When marking text* produced during straight copy typing exercises a word consists of the letters of a word, as generally understood, and the punctuation and spacing that follow them. Alternatively, when calculating gross typing speed, every five characters constitutes a word.
Syllable Intensity	Syllable intensity is the average number of syllables per word. It is an index of difficulty for straight copy typing material. Nearly always a typist's speed will increase as syllable intensity decreases.
Stroke Intensity	Stroke intensity is the average number of keyboard strikes per word. This is another index of difficulty for straight copy typing material. Just like syllable intensity, an inverse relationship exists between a typist's speed and stroke intensity.

* For the remainder of this document, the type of text of concern, when references are made to the marking of text, will be text produced during straight copy typing exercises.

Table 2

Computing Terms*

Term	Explanation
Text Editor	Software, operating with a human user in an interactive mode, that facilitates the entry, revision, and deletion of textual information into a computer system. A text editor provides a variety of facilities for copying, moving, inserting, deleting and searching for text and other useful features.
Source Code	The series of statements of a programming language, such as Pascal or Fortran, that defines a program. The source code must be translated into machine code so that a computer can execute it.
Object Code	The binary code into which source code is translated. Object code can be executed by a computer. Machine level language and machine code are synonymous with object code.
Compiler	Software that takes a computer program written in a programming language (i.e. source code) and translates it into machine level language (i.e. object code).
Linker	Software used after the translation of programs from source code to object code. A linker joins interconnected programs and subroutines into a format that can be loaded directly into the computer for processing.
Debugger	A software tool that allows the internal behaviour of a program to be investigated to aid in the process of finding errors or malfunctioning parts within it.

* The explanations for all of the computing terms were derived from definitions in "The Prentice-Hall Standard Glossary of Computer Terminology (Edmunds, 1985). Part of the definition for "linker" was also adapted from the "Dictionary of Computing" (Illingworth, 1984).

Summary

A need for an accurate marking algorithm for straight copy typing exercises was identified. To address this need a study was conducted to show that a new marking algorithm, developed for this study and called the WoBaT Marker, corrects text more accurately and faster than human markers. It was also noted that an accurate straight copy typing marking algorithm would save typing teachers a significant amount of time. Further, it was shown that such a program would reduce the amount of time students need to learn how to type. Lastly, definitions pertinent to this study were detailed.

Chapter 2

REVIEW OF THE LITERATURE

This chapter is divided into three sections. The first section addresses the rules for marking straight copy typing exercises. The second section reviews several typing programs. The review contains information about the accuracy and speed of the marking algorithms used in the typing programs. Lastly, the utility of Turbo Pascal*, the computer programming language used to define the WoBaT Marker, is examined.

Marking Rules

Four sources that contain marking rules were examined. Two of the sources used were the textbooks "Mastering Keyboarding Skills" (Ubelacker, Guest & McConaghy, 1983) and "Typing 300" (Rowe, Lloyd & Winger, 1972). Both of those textbooks are currently used in the local school districts. Two research based rule sources, a master's thesis by Helen Koch (1987) and a doctoral dissertation by Iris Johnson (1981), were also consulted. All four sources stated and/or exemplified the following two rules.

1. Only one error is logged for one word, no matter how many errors it may contain.

* Turbo Pascal is a product of Borland International, Inc. Scotts Valley, CA.

2. An error has been made in a word if any of its parts (i.e. letters, punctuation or spacing) differs from the corresponding parts of the expected word.

There is disagreement among the sources with respect to omission and repetition of text. Both Koch and Johnson stated that words omitted or repeated in succession should be counted as a single error. Alternatively, Rowe, Lloyd and Winger noted that in such an instance each word omitted or repeated should be counted as an error. Ubelacker, Guest and McConaghy ignored how multiple word omissions and repetitions should be counted.

The two methods for marking text omissions and repetitions just cited can be treated as legitimate so long as the method selected is used consistently. Straight copy timings in school settings serve two purposes: diagnosis and testing. In terms of diagnosis it is only important to note when words are omitted or repeated; it is not important to debate how many errors should be charged for such an infraction.

With respect to testing, West (1983, p. 364) states that there are two non-diagnostic functions served by straight copy testing in schools. First, straight copy testing serves "to check on the rapid gains (in gross speed) expected during the first few months of training." Since this testing function is related to gross speed, the differing marking methods of concern are not relevant to it because these marking methods are used to determine how accurately a student performs.

Second, according to West, a function of straight copy testing in schools "is to prepare for and predict employment test performance." The first part of this two part function is served when students take part in straight copy tests. It has nothing to do with marking methods. The second part, on the other hand, is related to marking methods. However, since predicting how accurately students may perform on employment typing tests requires only that a marking method be used consistently, both marking methods are legitimate and may be used effectively.

Common practice in the local school districts dictates that successive multiple word omissions and repetitions be treated as a single error. As such, the WoBaT Marker corrects text in that fashion.

Lastly, it is important to note that implicit in the manner in which errors were depicted in all four sources is the notion that accuracy should be gauged by an error count. This is in keeping with West's (1983, p. 377) statement: "[With respect to the appropriate scoring procedures for straight copy typing exercises,] absolute accuracy is properly measured by number of errors."

Typewriting Software Review

To determine the extent to which marking algorithms in computer typewriting programs are valuable, the investigator collected forty-three typewriting software reviews from thirteen distinct sources. Many of the reviews (Alberta Education, 1984; Alberta Education, 1985-1986; Alberta Education, 1986; Cameron, 1985; Dickinson, 1986; Hlynka-Laskiewicz, 1985; Lambrecht & Pullis, 1983) contained a description of the marking algorithm used to correct text generated during straight copy typing exercises.

Unfortunately, those descriptions did not include any statements about the utility of the marking algorithms. In fact, only one of the forty-three reviews contained information specific to the practicality of the marking algorithm used to correct text! Shapiro (1986, p. 34), commenting on the program "Mastertype" by D.C.Heath, stated: "The drills unfairly penalize students; a word that's off by one letter is marked off for every succeeding letter." As only this one statement specifically about the utility of marking algorithms was found, twenty-one typewriting programs were examined by the investigator to gain greater insight into the ways in which developers have implemented marking routines in the context of straight copy typing exercises. The investigator obtained the software from the Canadian Center for Learning Systems (CCLS), the Calgary Board of Education, the Educational Psychology Department at the University of Calgary and local software retailers.

Of the twenty-one programs reviewed, two were not suitable for the investigator's research purposes because they did not attempt to correct text in a straight copy typing exercise framework. (A list of the names and publication information of the nineteen programs reviewed appears in Appendix A.) The programs reviewed ran on Apple II series computers or IBM PC's, except for one program which ran on an Apple Macintosh, and were released between 1980 and 1987. As such they constitute a substantial sampling of current microcomputer educational typewriting software. The two most popular series of microcomputers for school use in North America, the Apple II series and the IBM PC series, were only introduced in 1977 (Cohen, 1987) and August 1981 (Getts, 1987), respectively.

When reviewing the typewriting programs the investigator first determined how accurately the marking algorithms corrected text. Since program code was not available, the investigator was forced to draw inferences about the marking algorithms by examining feedback about errors made by the typist. Of course the error feedback was based on assessments made by the marking algorithms in the various programs. Fortunately, drawing inferences based on feedback was usually straightforward because characters or words in error were highlighted in some manner in all but the two programs, "Typing Made Easy" and "Microtype: The Wonderful World of Paws." The highlighting of errors was usually done by showing characters or words in reverse video or by underlining them.

After determining the general approach of a marking algorithm the investigator classified it as either character based or word based. If a marking algorithm were deemed to be word based, two subsequent tests were performed. In the first test, words were intentionally omitted to determine if the marking algorithm could accommodate such an infraction. Then, in the second test, words were repeated. Again, this was done to assess the accuracy of the marking algorithm. Such tests were not necessary for character based marking algorithms since they could not compensate for the absence or addition of even a single character.

Regardless of a marking algorithm's classification, the investigator determined the flexibility of the marking algorithms with respect to two issues. First, the investigator determined if a marking algorithm could distinguish between upper and lower case characters. Then the investigator determined if custom-made documents could be used in a straight copy typing exercise setting which would submit copied text to the marking algorithm.

Those flexibility issues are vital because they separate programs which can use meaningful practice material from those that cannot. Meaningful practice material is text that a beginning typist would expect to type after gaining proficiency at the keyboard. For most people, then, practice materials should consist of sentences in paragraphs. The need for meaningful practice materials as described here is supported by several writers (for example, BeMent, Hirsch & Johnson, 1985-1986; Lambrecht & Pullis, 1983; McLean, 1984; Robinson, Erickson, Crawford, Beaumont & Ownby, 1979; Knapp, 1984; Wasylenki, 1985; West, 1983). Also, if a marking algorithm can correct text based on a custom-made document, it must be able to parse, or extract, words from sentences in the original document. Such algorithms are more robust than programs that merely compare user-entered words to words already in a program. Therefore, programs that contain the aforementioned parsing feature are superior to ones lacking it.

Figure 2 summarizes the results of the analysis of the typewriting software. The MECC "Typing" routine is the only reviewed routine that did

Figure 2. A summary of the typewriting software.*

Name	Character Based Marking Algorithm	Word Based Marking Algorithm	Upper and Lower Case Characters Permitted	Custom-made Documents Permitted
Minnesota Educational Computing Consortium Typing	X			X
Gregg Keyboarding for Information Processing		X		
Computer Keyboarding	X			
Micro-computer Keyboarding		X	X	X
Typing Made Easy	X			
Keyboarding Plus (2nd. Edition)	X		X	
Mastertype	X		X	
Typing for New Typists	X		X	
Typing Intrigue	X		X	

* An "X" indicates the presence of the feature listed at the top of the column.

Figure 2 (continued).

Name	Character Based Marking Algorithm	Word Based Marking Algorithm	Upper and Lower Case Characters Permitted	Custom-made Documents Permitted
Microtype: The Wonderful World of Paws	X		X	
Touch Type	X		X	X
Touch Typing for Beginners	X		X	
Stickybear Typing	X		X	
Superkey	X		X	
Type to Learn	X		X	X
Type!	X		X	X
Rainbow Keyboarding	X			
Success with Typing	X		X	X
Typing Tutor IV	X		X	X

Note: None of the typing programs listed, consistently compensated for omitted or repeated words. (For more details refer to the comments, beginning on P. 19, about the two word based typing programs.)

not function as a stand-alone program. It is actually one part of a set of business education utilities. It was included because it allows a user to enter one line of text, containing thirty-nine characters at most, and then lets the user practice typing the entered line. Also, the program marked the line typed in the practice session

Two programs did not highlight specific errors. In the program, "Typing Made Easy," exercise and test performance feedback pertaining to accuracy is presented to the user in an unusual manner. It is presented as a percentage. Seemingly, the percentage reflects the ratio of characters typed correctly to total characters typed. At one point during the review the investigator repeated a line of text. Upon completing the exercise the program displayed an accuracy rating of 17%. It appeared that a program marking on a per character basis would have produced an accuracy percentage approximately equal to the one cited by the program. On the other hand, a program marking on a per word percentage basis (words typed correctly to total words typed) would likely have computed a value approximately equal to 98%. As such it was concluded that this program contains a per character basis marking algorithm.

The other program that did not indicate any specific character or word errors was "Microtype: The Wonderful World of Paws." Instead, it purported to force the user to repeat a line if the program determined that the user entered more than half of the words in the line incorrectly. However, the investigator was forced to repeat a line of seven words even though the investigator made only three errors, as shown in Figure 3.

The errant program logic that forced the investigator to retype the line could have originated in one or both of two places. It is possible that the routine that computes error tallies erroneously computed an inflated error tally in this case. It is also conceivable that the one conditional expression that determined if more than half of the words typed are incorrect is itself incorrect. The investigator is willing to give the programmer the benefit of some doubt and suggest that the error lies in one, rather than in both, of the

Figure 3. A sample straight copy typing exercise from "Microtype: The Wonderful World of Paws".

O:	land work lend worn wood nice towns
U:	land work*len wron widd* nice towns

Notes: The words in italics are in error.

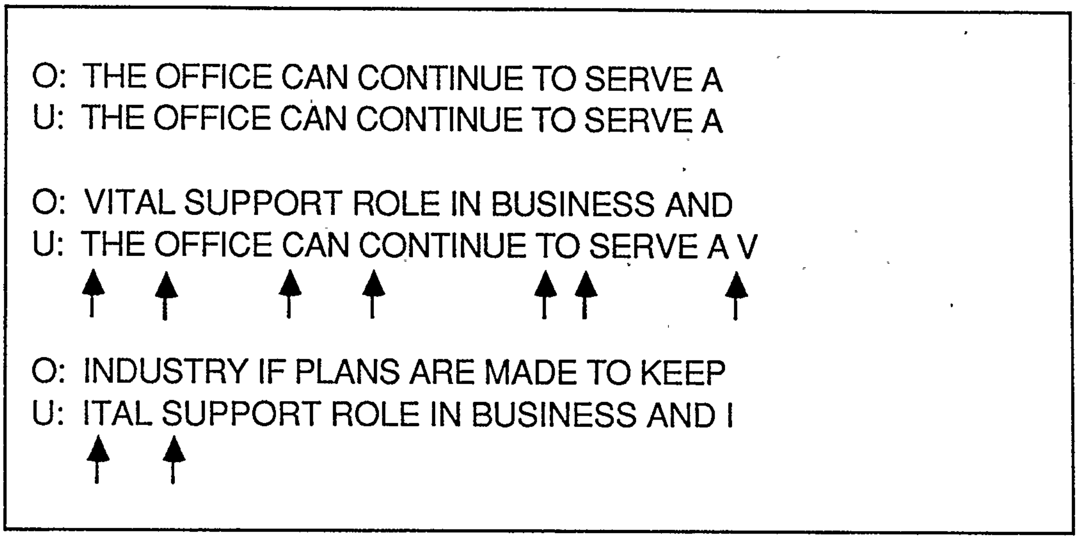
Original text is opposite the letter "O." User-entered text is opposite the letter "U."

An "O" for "original text" and a "U" for "user-entered text" will be used in all subsequent examples.

two potential erroneous places. It is more likely that the set of statements comprising the error tally routine is incorrect rather than a fragment of the conditional statement. As such, it is suspected that the error tally routine is erroneous. Since the error tally routine seemingly inflates error tallies it is believed that this program contains a per character basis marking algorithm.

Of the nineteen programs reviewed only the two programs, "Gregg Keyboarding for Information Processing" and "Microcomputer Keyboarding," contained marking algorithms that corrected text on a per word basis. To examine one practice exercise in "Gregg Keyboarding for Information Processing" the investigator typed "SHRPLY" for the characters "SHARPELY." The program properly logged just one error and corrected all subsequent text appropriately. However, the marking algorithm in "Gregg Keyboarding for Information Processing" cannot compensate for word omission nor insertion errors. The example in Figure 4 shows how the

Figure 4. An excerpt from a straight copy typing exercise in "Gregg Keyboarding for Information Processing".



Note: The program points to errors with arrows as shown.

program mismarked text that contained a multiple word repetition error.

The marking algorithm always terminates the marking process if more than eight errors are detected. Since more than eight errors were detected during the practice run shown in Figure 4 the marking process was terminated and the following message was displayed: "MORE THAN EIGHT ERRORS EXIST IN PARAGRAPH". The following message was also displayed: "SOME ERRORS MAY NOT SCORE PROPERLY. IF THIS HAPPENS, SIMPLY REDO THE DRILL." By including that message in the program it is apparent that the authors of the program knew about the shortcomings of the marking algorithm yet chose not to correct them.

In addition to the inaccurate marking algorithm, this program hinders one's using it for straight copy typing exercises in a number of ways. First, it does not perform the "word-wrap" function of word processors correctly. As can be seen in Figure 4 the letter "V" should have been brought down to the next user line along with the letters "ITAL" to form the word "VITAL." Secondly, the program forces the user to enter upper case letters at all times. Lastly, "Gregg Keyboarding for Information Processing" does not allow the user to create custom-made documents. This is especially limiting in this program because only three, six-line paragraphs are available for extended practice.

In all the programs reviewed, the marking algorithm in "Microcomputer Keyboarding" was judged most appropriate. It approached the marking process on a per word basis and to a limited extent it compensated for word omissions and repetitions. For example, Figure 5 shows how it appropriately logged just one error for the repetition of the word "of ".

Figure 6, alternatively, shows how the program mismarked the text when the three words, "and find a ", were omitted. The program incorrectly cited the words "better " and "way" as errors. Also, it logged one error at the end of the second line of user-entered text. Seemingly, this was done because the marking algorithm expected at least one more word.

Figure 5. An excerpt from a straight copy typing exercise in "Microcomputer Keyboarding".

O:	better way of fitting all the pieces together
U:	better way *of* of fitting all the pieces together

Figure 6. An excerpt from a straight copy typing exercise in "Microcomputer Keyboarding" depicting a multiple word omission error.

O:	get your typing act together, study how you type and find a
U:	get your typing act together, study how you type *better way*
O:	better way of fitting all the pieces together.
U:	of fitting all the pieces together. _______

Note: The line after the word "together. " was inserted by the program to denote an error.

While the marking algorithm in "Microcomputer Keyboarding" is considerably more accurate than the other ones reviewed, it does not correctly mark all single words. For example, the program erroneously logged two errors when the investigator typed "puzzle " for "puzzle, ". Only one error should have been logged in that case. Similarly, when the investigator typed "typng " for "typing " two errors were logged even though only one error was committed.

Other program shortcomings that adversely affect the marking algorithm are evident. The marking algorithm is not engaged until the user has entered as many characters on the last line as there are characters in the last expected line. Whenever an omission error (including a single character omission) has been made, the user must pad the remainder of the line with a spurious character which is subsequently flagged as incorrect. In addition, word omission errors can result in another type of error. In Figure 7 it is apparent that a word omission error in the first line left space at

Figure 7. An excerpt from a straight copy typing exercise in "Microcomputer Keyboarding" depicting an extraneous error introduced by an omission error.

O:	typing must fit into its specific place. So if you want to
U:	typing must fit into its specific place. So if you want *get*
O:	get your
U:	your

the end of the line for the first word of the second line of expected words. It is possible that the user-entered word "get" was marked wrong because it was associated with the end of line character. Whatever the case, it should not have been logged as incorrect.

Lastly, it must be noted that "Microcomputer Keyboarding" has no facility through which a user may utilize custom-made documents. Again this is especially limiting, for this program has but one four-line paragraph for extended accuracy practice!

When reviewing the typewriting programs the investigator also considered the speed of the marking algorithms if they were word based. Ideally, when comparing performance properties such as speed or memory requirements of algorithms run on different computers, a code-dependent rather than a machine-dependent notation should be used (see Standish, 1980). However, because program code was not available the use of such a notation was not possible. As such, before a comparison of algorithm speeds is made, one must at least take into account the different processing speeds of the systems used to execute the programs. The IBM PC used to run "Microcomputer Keyboarding" can execute nearly five million instructions per second while the Apple IIe used to run "Gregg Keyboarding for Information Processing" executes just barely over one million instructions per second.

The marking algorithm in "Gregg Keyboarding for Information Processing" was approximately twice as fast as the one in "Microcomputer Keyboarding." Obviously, the marking algorithm in "Microcomputer Keyboarding" executes many more instructions than the one in "Gregg Keyboarding for Information Processing" to achieve more accurate error tallies. As such it is understandable that the marking algorithm in "Microcomputer Keyboarding" is slower than the one in "Gregg Keyboarding for Information Processing."

The Utility of Turbo Pascal

Turbo Pascal is a computer programming environment in which fast executable programs may be developed. It has been shown (Bridger, 1986) that programs created in the Turbo Pascal programming environment are as fast or faster than programs created in other Pascal environments. This speed advantage is obviously of benefit to people using programs developed in the Turbo Pascal environment.

The Turbo Pascal programming environment consists of a text editor, a compiler and a transparent linker; it does not contain a debugger. The text editor allows the programmer to enter Pascal source code. The source code is then translated to machine code by the compiler. If even one error exists in the source code the compiler will flag the error. Then the programmer, by simply pressing a key, may go to the point in the source code that contains the error. Next, typically, the programmer attempts to correct it. It should be noted that such intricate coordination between compiler and editor is rare in a programming environment. Further, it is very beneficial because it saves time. In any industry, time saving devices are valuable. However in a programming environment, where only fifty lines of useable code are generated by a programmer per day (Unger, Lomow & Birtwistle, 1984), they are most welcome.

When the source code is error free and the compiler has translated all of it to machine code the programmer may execute the program. This is only possible because the machine code is automatically linked to memory by the linker. As this is the case the programmer is again spared some time. Next, programmers usually test the program by executing it. If errors are discovered, and they usually are, some source code must be rewritten. When source code is rewritten it must also be recompiled and the new machine code must be relinked. Since programmers often need to rewrite, recompile and relink code it is beneficial when programming environments are integrated and contain a fast compiler.

It has already been noted that the Turbo Pascal environment is integrated in that source code errors detected by the compiler are flagged and the programmer is directed immediately to the location of the error. It should also be noted that, at times, the programmer is directed to the point in the source code that is the cause of an error that occurred during program execution. This indicates another level of program environment integration. It is somewhat unfortunate that a debugger is not part of the Turbo Pascal environment since it would help programmers find the execution errors the environment could not locate.

With respect to compiling time, it has been reported that the Turbo Pascal compiler is easily five times as fast as other similar compilers (Bridger, 1986). In terms of program development, then, the Turbo Pascal environment offers tools through which programmers may create software expeditiously. For that reason it was selected for the present project.

Summary

In the first of the three sections in this chapter the rules for marking text produced during straight copy typing exercises were related. It was shown that typewriting rules, for the most part, can be applied in a consistent manner. However, since multiple word omissions and repetitions may be marked in one of two ways, some inconsistency in error tallies would result if markers did not use the same scoring method.

In section two the results of a typewriting software review conducted by the investigator were noted. It was shown that seventeen of nineteen typewriting programs viewed by the investigator contained marking algorithms that corrected text on a per character basis. The two programs, "Gregg Keyboarding for Information Processing" and "Microcomputer Keyboarding," contained word based marking algorithms. However, the marking algorithms within those programs failed to correctly tally errors when multiple word omission and/or repetition errors were made.

In the last section, Turbo Pascal was shown to be useful because it compiles program code quicker than other similar compilers. Also, executable code produced by Turbo Pascal runs as fast or faster than code generated by other Pascal compilers.

Chapter 3

METHODOLOGY

In this chapter the materials used and procedures followed for testing the two hypotheses advanced in Chapter I are detailed. Since the WoBaT Marker is only a marking routine, it was necessary to incorporate it into a program in order to test it. The copy typing program into which the WoBaT Marker was incorporated was developed by the researcher and is called SCOTT - **S**traight **CO**py **T**yping **T**ester. A brief description of the history and functional aspects of SCOTT are noted first. Then the development of the WoBaT Marker is described. Next, details pertaining to the development of the straight copy typing exercises are described. Following that are details concerning the human markers. Subsequently the events surrounding the development and administration of the attitude scale used in this study are documented. Then the Pilot Study is described and modifications to the original materials of this study are noted. Lastly, details pertaining to the main study are cited.

The Development and Function of SCOTT

SCOTT was developed over the 29-month period prior to the date on which data for this study were collected. It was tested with high school student typists, high school typewriting teachers, undergraduate instructional design students and expert instructional designers following formative evaluation procedures advocated by Weston (1986) and Dick and Carey (1985). The formative evaluations were conducted in individual, small group and large group settings.

To begin, as shown in Figure 8, SCOTT displayed an introductory screen of text that related only the purpose of the study. Then SCOTT displayed a screen of text that stated the role the typists would take in the study. That is, the typists were to take part in straight copy typing exercises.

To allow the typists to take part in straight copy typing exercises SCOTT displayed the text to be copied on the top half of each typist's monitor. Then, as the typists copied the text, SCOTT displayed the entered characters on the lower part of the screen. The manner in which the first exercise was displayed is evident in Figure 9. Figure 10 also shows how the first exercise was displayed; in addition though, it contains a sampling of user-entered text. It should be noted that only six lines of original text can be displayed on the top half of the monitor because all text is double spaced as per Figures 9 and 10. As such, lines of original text are updated while the typist copies text if the original document contains more than six lines. For example, if an original document contains seven lines of text (see Figure 11), after the typist completes the fourth line of text the first line of original text would be deleted and replaced by the seventh line of original text (see Figures 12, 13 and 14). Also, when more than six lines of text have been typed the user-entered text on the lower half of the monitor is scrolled upward (see Figures 15 and 16). Upward scrolling of text is typical of word processors and does not affect typists.

With respect to the manner in which entered characters were displayed, two other features should be noted. First, the lines of text were formed using an automatic word wrap routine much like ones often found in text editors. Secondly, the backspace key enabled students to delete text to the beginning of a line if desired. Again, this feature or one much like it is typically found in text editors. By incorporating simple screen displays as described by Mehlmann (1981) and standard editing techniques for user input as noted by Martin (1973) in SCOTT, the typists adapted to its user

Figure 8. The introductory screen in SCOTT.

A Keyboarding Program

This experimental program was written to test the effectiveness of its keyboarding marking algorithm.

Press any key to continue

Figure 9. The text entry screen in SCOTT for exercise 1.

All computers are made up of many units. One such unit is called the central processing unit. It is used to control all of the other parts in a computer. As such it is fitting that the word "central" is in its name.

Figure 10. The text entry screen in SCOTT for exercise 1 with a sampling of user-entered text.

All computers are made up of many units. One such unit is called the central processing unit. It is used to control all of the other parts in a computer. As such it is fitting that the word "central" is in its name.

All computers are made up of many unirs. On such unit iscalled the central processing unit. It is used to control all of the other parts in a computer. As such it is fitting that the word "central" is in ist name.

Figure 11. A sample straight copy typing exercise with seven lines of text.

```
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
is 5.
```

Note: The font used to display the text was changed to show precisely the spacing of the characters. This new font will be used in the next 5 figures for the same purpose.

Figure 12. The manner in which SCOTT would originally display the text in Figure 11.

```
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
```

Figure 13. The status of the text entry screen in SCOTT one key press before the first line of original text would be updated.

```
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
-----------------------------------------
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two char
```

Figure 14. The status of the text entry screen in SCOTT one key press after the first line of original text would have been updated.

```
is  5.
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
--------------------------------------------
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
chara
```

<u>Figure 15.</u> The status of the text entry screen in SCOTT one key press before the user-entered text would be scrolled upward.

```
is  5.


text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
--------------------------------------------
This sample straight copy typing
exercise contains seven lines of
text.  The longest lines in this
document contain thirty-two
characters.  The size, in
characters, of the shortest line
```

<u>Figure 16.</u> The status of the text entry screen in SCOTT one key press after the user-entered text would have been scrolled upward.

```
is  5.



document  contain  thirty-two
characters.   The  size,  in
characters,  of  the  shortest  line
--------------------------------------------------
exercise  contains  seven  lines  of
text.   The  longest  lines  in  this
document  contain  thrity-two
characters.   The  size,  in
characters,  of  the  shortest  line
i
```

interface readily.

To signal the completion of an exercise the typists pressed the F10 keyboard function key. SCOTT then saved the text produced during the straight copy typing exercise on diskette. After each typist completed all five of the straight copy typing exercises SCOTT displayed a message thanking the typist for participating in the study. It is important to note that the text generated by the typists during the first two straight copy typing exercises was not recorded on diskette. In this way the first two exercises were used only to familiarize the typists with the program, SCOTT. There existed no relationship between the content of the practice paragraphs used in the first two straight copy typing exercises and the three paragraphs used in the exercises in which data were collected.

The Development of the WoBaT Marker

In its earliest stages the WoBaT Marker was word based. However that characteristic alone was not entirely satisfactory to the student typists nor the typewriting teachers who evaluated it. It was evident that the algorithm would have to accommodate word omission and repetition errors to be functional. Consequently, changes were made to the WoBaT Marker until those shortcomings were rectified.

After the WoBaT Marker scored the text generated by the typists in this study, it saved a single whole number that denoted the number of errors it discovered. This permitted subsequent comparisons of the WoBaT Marker error tallies and the human marker error tallies.

A subroutine was included in SCOTT to time the WoBaT Marker. Just prior to initiating the WoBaT Marker, SCOTT executed an instruction that read the time of day clock in the computer. Then SCOTT, immediately after the WoBaT Marker finished scoring text, read the time of day clock again.

The difference in the two times of day was computed and stored on diskette for each paragraph. Since the time of day clock is updated every 55 milliseconds (approximately 18.2 times per second) it is accurate to within that time period.

The text generated by each typist was marked in parallel because each typist was at a stand-alone microcomputer. Therefore it would be unfair to sum all of the times taken to mark each paragraph and report that sum as the total marking time. Rather, the sum of the longest time taken to mark the thirty-five copies of Paragraph 1 and the equivalent times for paragraphs two and three constitutes the total marking time because all typists must have had their text marked within the time specified by that sum. As such that sum depicts the worst case scenario.

The Straight Copy Typing Material

Three paragraphs about computing were written by the researcher and together they comprised the straight copy typing material used in the study for data collection purposes. (The three paragraphs appear in Appendix B.) The words in the paragraphs were modified until the difference in syllable intensities of the paragraphs was at least 0.1. The syllable intensities of the paragraphs are 1.35 for Paragraph 1, 1.55 for Paragraph 2 and 1.69 for Paragraph 3. As syllable intensity increases typewriting difficulty increases (see Chapter 2, *Explanation of Terms*).

The syllable intensities represent "easy," "average" and "difficult" copy material for the typists in this study since they were nearing the end of *Typewriting 20.* According to the Typewriting 10-20-30 curriculum guide for Alberta (Alberta Education, 1985) *Typewriting 20* students should be copying materials with syllable intensities in the range 1.3 to 1.5. Also it should be noted that Paragraph 2 contains more words than Paragraph 1 and Paragraph 3 contains more words than Paragraph 2.

Statistics describing the length and difficulty of the paragraphs are summarized in Table 3. Note that the differences in stroke intensity also show that the paragraphs become increasingly more difficult to copy. However, the difference in stroke intensity between Paragraph 2 and Paragraph 3 is not statistically significant (West, 1983). The copy difficulty of the paragraphs, the length of the paragraphs and the instructions to the student typists were varied in the manner described to promote the production of a variable number of errors on a per paragraph basis.

Table 3

Descriptive Statistics for the Straight Copy Typing Material

Paragraph Number	Length (in words)	Syllable Intensity	Stroke Intensity
1	43	1.35	5.00
2	49	1.55	6.10
3	80	1.69	6.18

The Human Markers

All of the experienced human markers participated on a voluntary basis. The experienced markers had at least five years of typewriting teaching experience. The requirement for typewriting teaching experience was critical because typewriting teachers engage in marking text produced during straight copy typing exercises whereas others involved in typewriting may not necessarily mark text. The experienced markers, on average, had been teaching typewriting for 14 years. The least experienced typing teacher had 8 years of experience while the most experienced typing teacher had taught typewriting for 20 years. The number of years that each of the experienced markers had taught typewriting is detailed in Table 4.

In preparation for the scoring of the text by the experienced markers, the researcher produced paper copies of the 105 paragraphs generated by the typists. Each of the thirty-five paper copy pages contained three paragraphs produced by one of the typists. The thirty-five pages contained the text generated by all thirty-five typists. Each of those thirty-five paper copy pages was photocopied six times. In this way each of the experienced markers received a package consisting of thirty-five pages that contained the text generated by all of the typists. In addition, each experienced marker received one page that contained a copy of the original three paragraphs.

The experienced scorers marked the paragraphs by circling words entered in error, if any, and tallying the errors on a per paragraph basis. All of the experienced markers scored the text within one week of receiving their packages as instructed. When the experienced markers scored the text they timed themselves with stop watches. In addition to the marked paragraphs, the experienced markers submitted to the researcher the time it took them to score the paragraphs.

When the error identification and tally data generated by the

Table 4

Years of Typewriting Teaching Experience for each of the Experienced Markers

Years of Typewriting Teaching Experience
8
11
14
15
17
20

$\overline{X} = 14$

$s = 3.89$

experienced markers were collected, the investigator determined the actual errors made by the typists. This was done by pooling the results of all the experienced markers in the following manner.

1. The errors cited by the experienced marker who identified the most errors constituted the base of the actual errors.

2. Errors found by the other experienced markers that were not already in the initial base of errors were added to the error base.

Implicit in this approach is the notion that every typing error was discovered by at least one of the human markers. After completing that process the investigator had 105 actual error tallies, one for each of the 105 generated paragraphs. The investigator also had error tallies, on that same per paragraph basis, from each of the six experienced markers.

To control for variation in marking approaches among the experienced markers additional error tally data were also collected from inexperienced human markers. The inexperienced markers, all of whom were graduate students, had never marked text produced during straight copy typing exercises in any capacity other than as high school typewriting students. All of the markers had at least ten years teaching experience in an area other than business education. All volunteered to participate.

The inexperienced markers were trained to criterion in two groups of three using instructional materials which appear in Appendix C. To train the inexperienced markers to score text the investigator:

1. related introductory information about the nature of scoring straight copy typing exercises.

2. defined the term "word" in the context of marking text.

3. stated the rules for marking text.

4. discussed examples depicting each marking rule.

5. guided the learners through their initial attempts at marking text.

6. provided the learners with the opportunity to score text independently.

7. displayed correct assessments of the paragraphs that the learners scored independently.

After the learners had been trained to criterion they were handed a package containing the thirty-five pages of three paragraph sets generated by the student typists. In preparation for that the investigator photocopied the paragraph sets another six times. The inexperienced markers scored the text by circling words entered in error and tallying the errors on a per paragraph basis. The inexperienced markers scored the text immediately after receiving the paragraph sets. When the inexperienced markers had finished scoring the text the investigator had another six sets of error tallies on a per paragraph basis.

To summarize, since the error tallies were related on a per paragraph basis the investigator had amassed 105 error tallies from each of 6 experienced and inexperienced human markers, 105 error tallies from the WoBaT Marker and 105 error tallies that reflected the actual number of errors made by the typists. With those error tally data in hand the investigator had completed the error tally data collection process.

The WoBaT Marker Questionnaire

A six item Likert-type questionnaire was developed for this study. The questionnaire is in Figure 17. Note that each of the six items in the questionnaire pertains to the WoBaT Marker. Modifications to those items were made on the basis of critical comments advanced by graduate students and the investigator's advisors. The questionnaires were handed

Figure 17. The WoBaT Marker questionnaire (Reduced to 75% of Original Size)

A Questionnaire About

The **Word Based Typing (WoBaT) Marker**

Place a check mark between "strongly disagree" and "strongly agree" for each of the six scales below.

1.	A program that contained the WoBaT Marker would be cost effective in a classroom setting if it cost about $40.00 per copy.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree
2.	I would not save time if I used a program that contained the WoBaT Marker.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree
3.	Teachers should mark straight copy typing exercises, not a computer program that contained the WoBaT Marker.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree
4.	The WoBaT Marker does not score straight copy typing exercises accurately enough for classroom use.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree
5.	I would not incorporate a program that contained the WoBaT Marker into my typing classes.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree
6.	I would be willing to spend fifteen hours learning how to use a program that contained only the WoBaT Marker.	strongly disagree	_ 1	_ 2	_ 3	_ 4	_ 5	strongly agree

If you have any general comments about the WoBaT Marker, please state them below.

____

____

____

____

How many years have you been teaching typewriting? ___

to the experienced markers by the investigator after a brief presentation. During the presentation the investigator only related preliminary results of the study. The questionnaires were collected by the investigator immediately after the respondents had completed them.

The Pilot Study

For convenience, twenty five grade ten students enrolled in *Typewriting 10* at a local high school, distinct from the one in which the main study was conducted, were involved in the pilot study. The typists in the pilot study participated voluntarily. After each typist had completed the straight copy typing exercises displayed by SCOTT, each typist was invited to comment on the functionality of SCOTT and the WoBaT Marker. Further, the typists were encouraged to relate potential modifications that would likely improve those routines.

Based on the typists' suggestions and observations by the investigator one instructional screen in SCOTT was changed. The instruction that detailed how the typists were to signal that they had completed a document was modified. Also the WoBaT Marker was changed to accommodate white space characters entered at the end of a paragraph.

With respect to the straight copy typing material, results from the pilot study showed that the typists made, on average, 3.3 mistakes copying Paragraph 1, 5.3 mistakes copying Paragraph 2 and 12.1 mistakes copying Paragraph 3. Since those results were in keeping with what was expected no changes to the instructions to the typists nor to the paragraphs were made.

The Main Study

Students enrolled in *Typewriting 20* were chosen for this study because they are familiar with straight copy typing exercises. After meeting the requirements of the Ethics Review Committee (see Appendix D) the investigator asked the students in two *Typewriting 20* classes in a local high school to participate voluntarily. The 35 participating students were, for the most part, in grade eleven.

Each student typist was given a diskette upon which resided the program SCOTT. Each student typist started SCOTT at one of the stand alone microcomputers in the lab. There were more microcomputers in the lab than there were students in each of the classes so the student typists in each of the classes worked simultaneously.

Since this study compares marking sources for straight copy typing exercises it was vital that the student typists generate errors. Before the student typists began working with SCOTT the investigator ensured that they understood that point. Before the typists copied each of the paragraphs SCOTT displayed an instruction as per Figures 18, 19 and 20. The typists then copied the text displayed by SCOTT and the text they generated was scored by the WoBaT Marker.

When the investigator had collected all of the diskettes the investigator had collected 35 copies of each of 3 distinct paragraphs. Also the investigator had collected, by taking in the diskettes, results depicting how well the typists copied the paragraphs as determined by the WoBaT Marker.

Figure 18. The instruction screen in SCOTT that was displayed before the typists copied the first paragraph that was used for data collection purposes.

Instructions

As you copy the next document try for ACCURACY
by typing a little SLOWER than you normally do.

Press any key when ready to begin the next document

Figure 19. The instruction screen in SCOTT that was displayed before the typists copied the second paragraph that was used for data collection purposes.

Instructions

As you copy the next document pay due attention to both SPEED and ACCURACY by typing at your REGULAR pace.

Press any key when ready to begin the next document

Figure 20. The instruction screen in SCOTT that was displayed before the typists copied the last paragraph that was used for data collection purposes.

Instructions

As you copy the next document try for SPEED
by typing a little FASTER than you normally do.

Press any key when ready to begin the next document

Summary

In this chapter details pertaining to the testing of the two hypotheses advanced in chapter one were described. First, to show that the WoBaT Marker corrects text more accurately than experienced human markers, each of 35 student typists generated text during three distinct straight copy typing exercises. The text generated by the student typists was subsequently scored by the WoBaT Marker and six experienced human markers. Also six inexperienced human markers, all of whom were trained to mark text by the investigator, scored the text generated by the student typists. This was done to control for the various methods through which the experienced markers learned to score text.

Secondly, to show that the WoBaT Marker corrects text faster than human markers, timing data were collected while the text was scored. Also in this chapter it was noted that a six item Likert-type questionnaire about the WoBaT Marker was distributed to the experienced markers.

Chapter 4

RESULTS

Analyses of the data gathered are presented in this chapter. With respect to the error tally data, they are first analyzed in a preliminary manner through descriptive statistics. Then they are analyzed through the use of a parametric statistic. The timing data gathered in this study are presented through descriptive statistics. Lastly, results obtained from the items on the WoBaT Marker questionnaire are described.

The Error Tally Data

One hundred five error tallies from thirteen sources were obtained. These thirteen sources included the WoBaT Marker, the six experienced markers and the six inexperienced markers. In addition, errors cited by the six experienced markers were pooled to generate another 105 error tallies. These errors are treated as the actual error tallies.

For the purposes of analysis the three error tallies assigned to each typist, one for each paragraph, were summed. In this way the number of error tallies was reduced from 105 to 35. In addition, since differences that may exist among the individuals in the experienced and inexperienced human marker groups were not sought, means for those groups were computed. The analysis of the error tally data, then, will be based on the 35 error tally means computed for each of the experienced and inexperienced human marker groups, the 35 error tally sums determined by the WoBaT Marker and the 35 actual error tally sums.

The error tally sums derived by all marking sources and the actual error tally sums for each typist appear in Appendix E. Descriptive statistics summarizing the actual error tally data and the error tallies of the WoBaT Marker and the human marker groups appear in Table 5. Based on that summary it is evident that the marking sources detected differing numbers of errors. Further, since the WoBaT Marker mean score is closer to the actual mean score than either of the two group mean scores it is evident that the WoBaT Marker scored the text more accurately than the human markers. As the subsequent analyses rely heavily on the total number of actual errors it is worthwhile to note that the typists made 262 errors altogether.

Preliminary Analysis of the Error Tally Data

The percentage of agreement between the actual scores and those of the experienced markers, the inexperienced markers and the WoBaT Marker are detailed in Figure 21. The experienced markers found, on average, 85% or 222 of the 262 errors while the inexperienced markers found, on average, 81% or 213 of the 262 errors. The percentage of errors found by the experienced markers ranged from 92 to 77. For the inexperienced markers the percentage of errors found varied from 84 to 73.

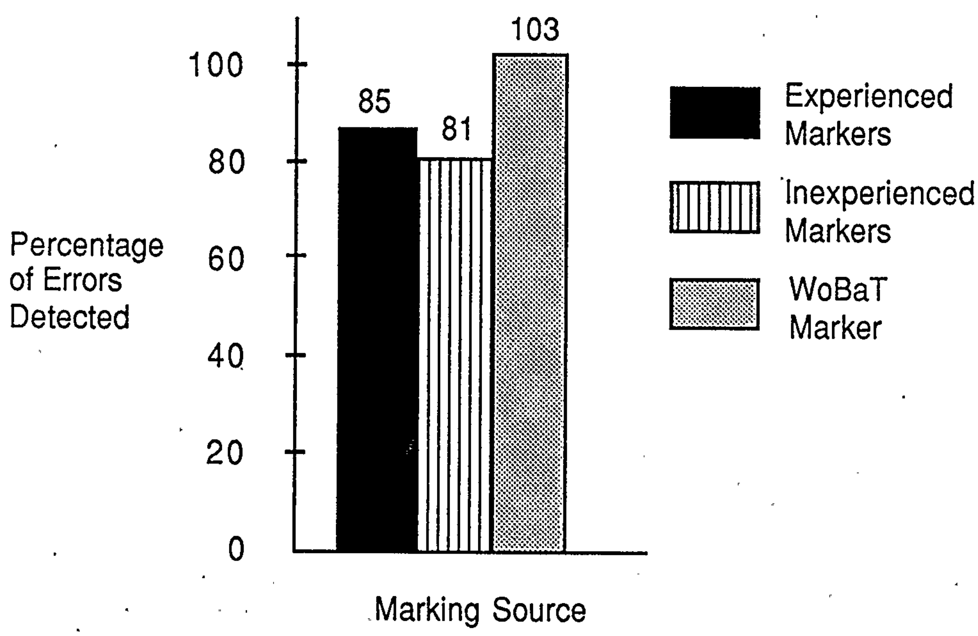
The WoBaT Marker logged 270 errors even though the typists made 262 errors. It located all but one of the typists' errors and cited nine words as errors even though they were typed correctly. As evident in Figure 21 the percentage assigned to the WoBaT Marker is 103 since 270 errors were logged and 262 were made. However the WoBaT Marker made 10 errors as described (see Chapter 5 for the analysis of the errors made by the WoBaT Marker). As such it may reasonably be reported that the WoBaT Marker operated at 96 percent effectiveness [i.e. (262 - 10) ÷ 262 * 100].

Table 5

Summary of the Actual Error Tally Data and the Error Tally Data for the WoBaT Marker, the Experienced Markers and the Inexperienced Markers

	Actual	WoBaT Marker	Experienced Markers	Inexperienced Makers
Mean	7.49	7.71	6.35	6.08
Standard Deviation	8.18	8.82	7.38	7.31

Figure 21. Percentage of errors detected for each marking source.



Detailed Analysis of the Error Tally Data

To determine if the differences in error tally sums were statistically significant, t-tests were performed. The computer program, Statistical Package for the Social Sciences (Nie, Hull, Jenkins, Steinbrenner & Bent, 1975) was used to compute the t-tests. Since the text generated by each typist was scored by each of the thirteen marking sources, t-scores for correlated data were computed. The significance level for all t-tests was set at the five percent level ($p = 0.05$). All of the t-tests were 2-tailed and the degrees of freedom was 34.

Before each t-score was computed an F-test was conducted to ensure that the variances of the groups were homogeneous. In all cases the F-values were not statistically significant. Therefore in each t-test, the underlying assumption of homogeneity of variance was met. Also for each t-test, the underlying assumption of normal distribution was assumed because the sample size in this study is 35. Freund (1979, p. 231) noted that a sample size of thirty is usually regarded as sufficiently large to apply the central limit theorem which justifies the use of normal curve methods. Further, even if one or more of the group distributions did not approximate the normal curve, the t-test is robust to violations of its assumptions (Freund, 1979).

First, a t-test comparing the experienced and inexperienced marker groups was conducted to determine if the human marker groups could be pooled. In this case a statistically significant t-score would indicate that the human marker groups should be treated separately, otherwise they could be pooled. Since the t-test revealed a statistically significant t-score, namely 2.23, the human marker groups will be considered as distinct groups. The t-score calculated for the two human marker groups also appears in Figure 18 with all of the other t-scores computed for this study.

Next, t-tests were conducted to assess the difference between the

Figure 22. Matrix of all t-scores

	E.M.	I.M.	W.M.	Actual
E.M.	-	2.23*	-5.02**	-6.68**
I.M.	-	-	-5.54**	-6.70**
W.M.	-	-	-	1.60
Actual	-	-	-	-

E.M. - Experienced Markers
I.M. - Inexperienced Markers
W.M. - WoBaT Marker

* t-score is significant at the 0.05 level
** t-score is significant at the 0.001 level

WoBaT Marker and the human marker groups. The t-score depicting the difference between the WoBaT Marker and the inexperienced marker group (i.e. -5.54) and the t-score depicting the difference between the WoBaT Marker and the experienced marker group (i.e. -5.02) are significant at the 0.001 level. Therefore it is maintained that the WoBaT Marker scores text differently than experienced and inexperienced human markers.

To determine whether human markers or the WoBaT Marker was superior, t-tests indicating the difference between the actual scores and the scores produced by the human marker groups were conducted. In addition, a t-test denoting the difference between the actual scores and the WoBaT Marker was completed. The t-scores indicating the difference between the actual scores and human marker scores, namely -6.70 and -6.68, are significant at the 0.001 level. Therefore the null hypotheses of no difference were rejected. Lastly, the t-score depicting the difference between the actual scores and the WoBaT Marker scores, namely 1.60, is not statistically significant. As such the null hypothesis was retained and it is held that there is no difference between the actual error tallies and those cited by the WoBaT Marker. Given the results of those t-tests it is evident that the WoBaT Marker corrects text more accurately than inexperienced and experienced human markers.

Timing Data of the Human Markers

As noted in the previous chapter the human markers timed themselves as they marked the text and, when finished, each of them recorded their marking time in minutes and seconds. Those times reflect total marking time per marker. The total marking times for all of the human markers appear in Table 6. The human markers as one group scored the text, on average, in 42.36 minutes. The standard deviation for the entire

Table 6

Marking Times of All Human Markers by Group

Experienced Markers	Inexperienced Markers
39.48	48.95
30.32	49.35
42.83	29.30
33.75	45.13
48.68	46.53
45.55	48.45
$\bar{X}$ = 40.10	$\bar{X}$ = 44.62
s = 6.42	s = 7.00

Note: All marking times are in minutes. To maintain consistency in times the investigator converted the number of seconds reported by each human marker to its corresponding fraction of a minute.

human marker group is 7.09 minutes.

Timing Data of the WoBaT Marker

The WoBaT Marker was timed by the clocks in the computers used by the typists. Since the WoBaT Marker was timed on a per paragraph basis marking times for each of the 105 paragraphs generated were recorded and are cited in Appendix F.

As noted in Chapter 3 (p. 39) summing the WoBaT Marker scoring times for each paragraph would not be appropriate because the final sum would not accurately reflect the parallel nature in which the text was marked. Instead, the WoBaT Marker timing data will be viewed in the form of *worst case, average case* and *best case* scenarios. The worst case scenario will reveal the longest WoBaT Marker scoring time for each paragraph. Conversely in the best case scenario the minimum WoBaT Marking scoring time for each paragraph will be reported. The WoBaT Marker scoring times in the average case will reflect the mean scoring times for each paragraph. The WoBaT Marker scoring times for the worst, average and best cases are shown in Table 7.

Total times for the three cases noted are relevant. At worst, 33.00 seconds may have elapsed as the WoBaT Marker scored the text of a student. The sum of the three average case times shows that, typically, it took 10.18 seconds to score the three paragraphs generated by each typist. At best, it may have taken 8.40 seconds to mark the text of a student.

Contrasting the Human Markers and the WoBaT Marker Timing Data

The human markers took, on average, 42.36 minutes to score the text whereas the WoBaT Marker took, typically, 0.17 minutes (i.e. 10.18 seconds). In other words, on average, the WoBaT Marker scored the text 249 times faster than the human markers.

Table 7

WoBaT Marker Scoring Times in Seconds

	Worst Case	Average Case	Best Case
Paragraph A	3.70	2.22	2.10
Paragraph B	11.90	2.89	2.40
Paragraph C	17.40	5.07	3.90

The WoBaT Marker Questionnaire Data

The questionnaire used in this study appeared in Figure 17 (p. 45). From Figure 17 it is evident that the Likert scale in the questionnaire varied from one to five. To maintain consistency in the responses the most favourable attitude towards the WoBaT Marker is always assigned a mark of five, the next most favourable attitude is always assigned a mark of four and so on, to the least favourable attitude which is always assigned a mark of one. To accomplish this, the numeric responses to questions phrased in an affirmative manner remain unchanged. However, the numeric responses to negatively phrased questions (i.e. questions 2 - 5) are subtracted from six (the number of scale points plus one) to obtain the appropriate value. For example in questions two through five, a rating of 1, which indicates the strongest disagreement to a negative feeling about the WoBaT Marker, will be transposed to 5 (i.e. 6 - 1) which is indicative of the strongest positive feeling toward the WoBaT Marker. With the data transposed in this manner, values approaching 1 always indicate negative feelings toward the WoBaT

Marker whereas values approaching 5 always denote positive feelings.

The data obtained from the questionnaire will be treated as ratio data as is typically done (see Henerson, Morris & Fitz-Gibbon, 1978). A descriptive summary of the questionnaire data is in Table 8. Raw data from the questionnaire are in Appendix G.

Table 8

Descriptive Statistical Summary of the Questionnaire Data

	1	2	3	4	5	6
Mean	4.17	5.00	4.67	4.83	4.67	3.00
Standard Deviation	0.69	0.00	0.47	0.37	0.47	1.15

Item 1. *A program that contained the WoBaT Marker would be cost effective in a classroom setting if it cost about $40.00 per copy.*

The mean of the six responses to this item is 4.2. As such the cost as listed would seemingly be appropriate.

Item 2. *I would not save time if I used a program that contained the WoBaT Marker.*

The respondents all strongly agreed that the WoBaT Marker would save them time. The mean for the adjusted scores, as described above, is 5.0.

Item 3. *Teachers should mark straight copy typing exercises, not a computer program that contained the WoBaT Marker.*

For this item, the adjusted score mean of 4.7 suggests that the respondents would welcome the marking of straight copy typing exercises by the WoBaT Marker.

Item 4. *The WoBaT Marker does not score straight copy typing exercises accurately enough for classroom use.*

The mean of the adjusted scores for this item is 4.8. This indicates that the respondents consider the WoBaT Marker to be accurate enough for classroom use.

Item 5. *I would not incorporate a program that contained the WoBaT Marker into my typing classes.*

The adjusted score mean for this item is 4.7. As such it would appear that the respondents would be willing to incorporate a program that contained the WoBaT Marker into their classrooms.

Item 6. *I would be willing to spend fifteen hours learning how to use a program that contained only the WoBaT Marker.*

The variability of responses to this item was widespread. As such the mean response of 3 does not accurately depict a general attitude of the respondents. Three respondents checked the 2 Point, one other respondent checked the 3 Point, another respondent the 4 Point and the remaining respondent checked the 5 Point.

The mean of the six items' means is 4.2. Thus it appears that the respondents had a generally favourable attitude toward the WoBaT Marker.

<u>Respondents' Comments</u>

Four of the six questionnaire respondents wrote comments. All of the comments pertained to the WoBaT Marker and were favourable toward it. For example, one respondent wrote:

> [The WoBaT Marker] would certainly reduce one's marking time and provide more consistency in the marking.

As a second example, another respondent wrote:

> [The WoBaT Marker] is an excellent program. We certainly need a program like this - it will save hours of marking.

Summary

The error tally data were analyzed through the use of t-tests. The t-tests showed that a significant difference existed between the actual error tallies and the error tallies generated by each of the experienced and inexperienced marker groups. On the other hand, the t-test comparing the actual error tallies and the WoBaT Marker error tallies did not reveal any significant difference. Therefore it was concluded that the WoBaT Marker scores text produced during straight copy typing exercises more accurately than human markers.

Analysis of the timing data revealed that the WoBaT Marker scored the text approximately 249 times faster than the human markers. Much of the WoBaT Marker's speed advantage is attributable to the parallel nature in which it scores text.

Responses to the items on the WoBaT Marker questionnaire indicated that the respondents held a generally favourable attitude toward it.

Chapter 5

DISCUSSION

Results in the last chapter indicated that the marking sources were imperfect. With respect to the human markers, this is understandable. However, it is not clear why the software-driven WoBaT Marker erroneously generated inflated error tallies. As such a discussion of the scoring inequities of the WoBaT Marker appears in the first section of this chapter. In the second section factors affecting the performance of the typists are examined. In the third section the timing data are discussed. The fourth section contains comments about potential benefits of the WoBaT Marker to teachers and students. Then suggestions for further research are stated and, finally, conclusions based on this study are drawn.

WoBaT Marker Scoring Inequities

The ten errors made by the WoBaT marker were of three different types. The first error type, which accounted for seven of the errors made by the WoBaT Marker, is exemplified in Figure 23. Since the WoBaT Marker logged three errors it inflated the correct error tally of two by one. When scoring the text that appears in Figure 23 the WoBaT Marker marked the user-entered words 'Predicate ', 'Logic." ', 'to ' and 'be ' as correct because they match precisely the corresponding original words.

The WoBaT Marker then assessed the three errors in the following manner. The user-entered word "more " was marked as incorrect because it is not the same as the original word "even ". The word "technicl " was

Figure 23. WoBaT Marker error type I.

Correctly Marked Text

O:	Predicate Logic." To be *even* more technical Prolog was actually
U:	Predicate Logic." To be more *technicl* Prolog was actually

Text as Marked by the WoBaT Marker

O:	Predicate Logic." To be even *more* technical Prolog was actually
U:	Predicate Logic." To be *more technicl* Prolog was actually

Note: Words marked as errors are in italics.

marked as incorrect because it is not the same as the word "more ".

After logging two consecutive errors the WoBaT Marker checked to determine if the word "even " had been skipped. It did this by comparing the user-entered words "more " and "technicl " to the original words "more " and "technical ". Of the fifteen potential character matches, the characters in the two user-entered words correspond 12 times with the characters in the two original words (see Figure 24). Phrased alternatively, the percentage of compatibility between the two user-entered and original words is 80. In the WoBaT Marker the threshold value at or above which words are deemed to be the same is 81 percent. As such the WoBaT Marker rejected the notion that the words "more " and "technicl " were the same as the words "more " and "technical ". Therefore it also rejected the notion that the word "even " was skipped.

In terms of actual scoring it would have been correct to cite the words "more " and "technicl " as no different from the words "more " and "technical " because in so doing it would have been evident that the word "even " was skipped. By incorrectly rejecting the notion (or hypothesis) of no difference when it was actually true the WoBaT Marker made an error in scoring. This is the essence of the WoBaT Marker Error Type I.

After the WoBaT Marker rejected the notion that the word "even " was skipped it continued to score the text by comparing the word "Prolog " with the word "technical ". Since those words are not the same the user-entered word "Prolog " was initially logged as incorrect (see Figure 25). Then a test was made to determine if the original word "more " was skipped. To do this the user entered words "technicl " and "Prolog " were compared to the original words "technical " and "Prolog ". As per Figure 26 those user-entered and original words are 82 percent compatible because 14 of the 17 characters match. Therefore it was deemed that the word "more " was skipped.

Further, since a word was skipped it was known that the words

Figure 24. Comparing user-entered and original words I.

O:	more technical
U:	more technicl

Note: The two words "more " and "technical " contain 15 characters because each word contains a space bar character.

The user-entered characters that match the original characters are underlined.

Figure 25. Logging the word "Prolog " as incorrect initially.

O:	Predicate Logic." To be even more technical Prolog was actually
U:	Predicate Logic." To be *more technicl Prolog* was actually

Note: Words marked as errors are in italics.

Figure 26. Comparing user-entered and original words II.

O:	technical Prolog
U:	technicl Prolog

Note: The two words "technical " and "Prolog " contain 17 characters because each word contains a space bar character.

The user-entered characters that match the original characters are underlined.

"technicl " and "Prolog " were not compared to the appropriate original words. As such, after realignment, the word "technicl " was subsequently compared to "technical " and found still to be in error. However, the user-entered word "Prolog " was then compared to the original word "Prolog ". When the WoBaT Marker determined that those words matched, the initial incorrect assessment of the user-entered word "Prolog " was rescinded as was evident in Figure 23.

The WoBaT Marker then correctly scored the text remaining in that document. Lastly, with respect to this error type, it is important to note that the other errors of this type did not always result in the incorrect citing of a skipped word as evident in this example.

The second WoBaT Marker error type, which accounted for the only actual error missed by the WoBaT Marker, is the conceptual converse of error type I. That is, the WoBaT Marker accepted a notion (or hypothesis) of no difference as true when it was actually false. The instance of this error type is cited in Figure 27. The WoBaT Marker in this case logged just one

Figure 27. WoBaT Marker error type II.

Correctly Marked Text

O:	Computers can run a lot of different application programs. Of
U:	Computers can run a lot of different application *programs. jOf*

Text as Marked by the WoBaT Marker

O:	Computers can run a lot of different application programs. Of
U:	Computers can run a lot of different application *programs. j*Of

Note: Words marked as errors are in italics.

error by erroneously associating the letter "j" with the word "programs. " to form the word "programs.j ".

The WoBaT Marker scored the text in Figure 27 in the following manner. The user-entered words beginning with the word "Computers " and continuing through to the word "application " were marked as correct because they are exact copies of the corresponding original words. The user-entered word "programs. " was marked as incorrect because it contains just one space bar character after the period as opposed to the original word "programs. " which contains two spaces after the period. The word "jOf" was marked as incorrect initially because it does not match "Of". (This is depicted in Figure 28.) Since the lengths of the user-entered words "programs. " and "jOf" were one less and one more than the lengths of the expected original words "programs. " and "Of" respectively (see figure 29), a transposition error test was conducted. A transposition error occurs when characters are typed out of order. For example, typing "do g" instead of "dog " is a transposition error. The transposition test was conducted to determine if the letter "j" was transposed with the space bar character.

To conduct the test the letter "j" was removed from the word "jOf" and attached to the word "programs. " The newly formed words "programs.j " and "Of" were compared to the original words "programs. " and "Of". Those user-entered and original words are 92 percent compatible because 12 of the 13 characters match (see Figure 30). Therefore the hypothesis of no difference between the user-entered and original words was upheld in contrast to proper marking practice. As a result the letter "j" was deemed, by the WoBaT Marker, to belong to the word "programs. " instead of belonging to the word "jOf". As such the "j" was removed from the user-entered word "jOf" to form the word "Of". This newly formed user word matches precisely the original word "Of". Consequently the WoBaT Marker rescinded the error initially logged to the word "jOf" and reported one error as was shown in Figure 27.

Figure 28. Logging the word "jOf" as incorrect initially.

O:	Computers can run a lot of different application programs. Of
U:	Computers can run a lot of different application *programs. jOf*

Note: Words marked as errors are in italics.

Figure 29. Comparing word lengths.

Case # 1

Original word:	"programs. "	{ Length: 11 characters, including the 2 space bar characters }
User-entered word:	"programs. "	{ Length: 10 characters, including the 1 space bar character }

Case # 2

Original word:	"Of"	{ Length: 2 characters }
User-entered word:	"jOf"	{ Length: 3 characters }

Note: In Case # 1 the user-entered word is one character less than the original word. The converse is true in Case # 2.

Figure 30. Comparing user-entered and original words III.

O:	programs. Of
U:	programs.j Of

Note: The two words "programs. " and "Of" contain 13 characters because "programs. " contains two space bar characters and "Of" does not contain any space bar characters.

The user-entered characters that match the original characters are underlined.

The final WoBaT Marker error type, which accounted for the remaining two errors that the WoBaT Marker made, is unrelated to the first two error types. It is exemplified in Figure 31. In this instance the WoBaT Marker incorrectly cited the user-entered word "games " as a repeated word error. At the time this study was conducted the WoBaT Marker did not have any facility through which it could differentiate between spuriously inserted text and repeated text. It always marked text as repeated text if word realignment were attained by comparing the current user-entered word to a previous original word. Such was the case here. The user-entered word "playing " was marked as incorrect because it does not match the original word "games ". Then the user-entered word "games " was marked as a repeated word because it matched the previous original word "games ".

Figure 31. WoBaT Marker error type III.

Correctly Marked Text

O:	people do not always favor games with fancy graphics and
U:	people do not always favor *playing* games with fancy graphics and

Text as Marked by the WoBaT Marker

O:	people do not always favor games with fancy graphics and
U:	people do not always favor *playing games* with fancy graphics and

Note: Words marked as errors are in italics.

Factors Affecting the Performance of the Typists

As noted in Chapter 3 (p. 47), since this study compares marking sources it was critical that the student typists generate errors. Indeed, if no errors existed in the text generated by the typists, it is unlikely that useful results would have been attained. As such, also as noted in Chapter 3 (p. 40), to promote the production of a variable number of errors on a per paragraph basis the copy characteristics and lengths of the straight copy typing material were varied. Moreover, the instructions detailing the pace at which the paragraphs were to be copied were also varied.

It is likely that the students typists were most affected by the paragraph lengths and the instructions they received before typing the paragraphs. Certainly the probability for error increased with every paragraph because the shortest paragraph was copied first, the next shortest paragraph was copied second and the longest paragraph was copied last.

Evidence indicating that typists will make more errors than usual if they type faster than their normal pace and fewer errors than usual if they type slower than their normal pace can be found in Robinson, Erickson, Crawford, Beaumont & Ownby (1979). West (1983, p. 374) also stated, "*speed and accuracy* scores can change substantially under differing instructions." Lastly, in separate studies by Bell, Diehl, McInturff and Robinson (all documented in West, 1983) copy characteristics, such as syllable and stroke intensity, were never shown to have affected accuracy scores significantly. Those studies and others led West (1983, p. 369) to conclude, "accuracy is essentially unaffected by copy characteristics."

The Timing Data

In Chapter 4 (p. 59) it was reported that the WoBaT Marker scored the text, on average, 249 times faster than the human markers. It is important to note that the parallel nature of the marking accounts for the greatest percentage of this time saving. For example, if only one computer were available, the text generated by the students would have been scored consecutively rather than simultaneously and approximately 5.95 minutes [0.17 minutes (typical marking time per student) * 35 (number of students) = 5.95] would have elapsed during the marking process. Such a marking process would be approximately 7 times as fast as human markers.

Since the marking process described contains no simultaneous marking component this speed gain is attributable solely to the software driven marking algorithm. In reality, business education computer laboratories contain more than one computer. Therefore the parallel nature of the WoBaT Marker scoring method should not be neglected. In the final analysis, the overall speed gain of the WoBaT Marker can be computed as 7 (the speed gain of the software-driven marking algorithm) times the number of computers running the WoBaT Marker. For example, if the WoBaT Marker is scoring text on 30 computers, it will mark the text, on average, 210 times as fast as human markers.

Potential Benefits of the WoBaT Marker to Teachers and Students

The potential benefits of the WoBaT Marker for both teacher and student groups originate from the gains in accuracy and speed exhibited by the WoBaT Marker. With the gain in accuracy, students may review accurate assessments of their typing performance. In so doing they should gain insights into ways in which their typing performance can be improved.

In particular they should be able to cite character sequences that pose difficulties. After some practice, they should then be able to type those character sequences with relative ease. Further, as a result of the WoBaT Marker speed gain, students may participate in this revision process without delay.

The gain in accuracy demonstrated by the WoBaT Marker should help teachers diagnose problems which are hampering their students. Further, teachers should be able to consistently relate grades for straight copy typing exercises that closely resemble each students' true score. Lastly, due to the WoBaT Marker speed gain, teachers should save a substantial amount of time as anticipated by all of the teachers who completed the WoBaT Marker Questionnaire. As noted in Chapter 4 (p. 62) all of the respondents strongly agreed with the notion that the WoBaT Marker would save them time.

Suggestions for Further Research

Further research into the effectiveness of straight copy typewriting marking sources could proceed along several avenues. First, the materials used in this study could be modified. In so doing a researcher could, for example, determine if the same results would be obtained for different straight copy typing materials. Alternatively, the procedures followed in this study could be changed. For example, the human markers could be randomly selected, the number of human markers could be increased, the inexperienced markers could be trained in a different manner or expert typists could be selected instead of novice typists. It is also possible to create several worthwhile experimental designs by modifying both the procedures followed and the materials used in this study.

However, further research based on this study need not be confined to permutations of its methodology. Potentially, a researcher may wish to study the effects of the user-interface of SCOTT on typists. Along this potential research avenue are questions such as:

Do student typists, in general, prefer to practice copying text from a monitor or from a book?

Do student typists respond appropriately to the prompts in SCOTT?

Does the facility in SCOTT that provides its users with immediate results of their typing performance affect the rate at which they learn to type?

Lastly, another potential research area, distinct from the first two, would concern researchers likely unrelated to the educational researchers who may wish to pursue the first two research avenues. Researchers concerned about computing issues may wish to develop a straight copy typing marking algorithm that differs from the WoBaT Marker. Towards that end they may wish to pursue, for instance, algorithm analysis.

Conclusion

This study demonstrated that the WoBaT Marker corrects text generated during straight copy typing exercises more accurately than human markers. In addition, it showed that the WoBaT Marker corrects text generated during straight copy typing exercises faster than human markers. As such one would expect that incorporating the WoBaT Marker into classroom settings would be beneficial. Students should benefit through immediate and valid assessments, beyond that which human markers could be expected to provide, of their typing performances. Teachers should reap time saving benefits.

References

Alberta Education. (1984). Computer courseware evaluations. Edmonton, AB. pp. 3-4, 21-24, 33-36.

Alberta Education. (1985, May - 1986, March). Computer courseware evaluations. Edmonton, AB. pp. 105-106.

Alberta Education. (1985). B.Q.R.P.: Senior high business education equipment and the elementary/junior high school computer projects. Edmonton, AB.

Alberta Education. (1986, April - 1986, December). Computer courseware evaluations. Edmonton, AB. pp. 85-86.

Alberta Education. (1986). Microcomputers in alberta schools. Edmonton, AB.

BeMent, L., Hirsch, J., & Johnson, R. E. (1985-1986). Superkey (A software review). The Computing Teacher, 13(4), 53-55.

Bridger, M. (1986, February). Turbo Pascal 3.0. Byte, pp. 281-286.

Business Education Council of the Alberta Teachers' Association. (1987). Computers in business education. Edmonton, AB.

Cameron, J. (1985, October). Key to typing (A software review). Computers in Education, pp. 28-29.

Cohen, A. (1987, November). IBM vs. Apple: Product developments. PC Magazine, p. 100.

Coon, D. (1986). Introduction to psychology (4th ed.). St. Paul, MN: West Publishing.

Dick, W., & Carey, L. (1985). A systematic approach to instructional design. Glenview, Illinois: Scott, Foresman and Company.

Dickinson, C. (1986). Dungeon doom typing (A software review). Electronic Learning, 5(6), 56.

Edmunds, R. A. (1985). The prentice-hall standard glossary of computer terminology. Englewood Cliffs, New Jersey: Prentice-Hall.

Etier, F. (1971). Typewriting by electronics. Business Education Forum, ?(??), 20-24.

Getts, J. (1987, August). A pc genealogy. PC World, pp. 200-205.

Henerson, M. E., Morris, L.L., & Fitz-Gibbon, C.T. (1978). How to measure attitudes. Beverly Hills, CA: Sage.

Hlynka-Laskiewicz, G. (1985, September). Mastertype (A software review). Computers in Education, pp. 61-66.

Illingworth, V. (1984). Dictionary of computing. New York, NY: Market House Books.

Johnson, I.W. (1981). Effects of keystroking, planning, and error correction on proficiency at typing business letters of varying difficulty. Dissertation Abstracts International, 46/05A, 1169. (University Microfilms No. DER85-14695).

Knapp, L.R. (1984). Finding the best typing tutorials. Classroom Computer Learning, 5(2), 70-71.

Koch, H. (1987). Word processing skills. Unpublished master's thesis, The University of Calgary, Calgary, AB.

Lambrecht, J. J., & Pullis, J. M. (1983). Computer assisted instruction in typing. Educational Computer Magazine, pp. 42-68.

Martin, J. (1973). Design of man-computer dialogues. Englewood Cliffs, NJ: Prentice-Hall.

McLean, G. N. (1984). Teaching keyboarding/typewriting. St. Paul, MN: Delta Pi Epsilon.

Mehlmann, M. (1981). When people use computers: An approach to developing an interface. Englewood Cliffs, NJ: Prentice-Hall.

Nie, N.H., Hull, C.H., Jenkins, J.G., Steinbrenner, K., & Bent, D.H. (1975). Statistical package for the social sciences. New York, NY: McGraw-Hill.

Robinson, J. W., Erickson, L. W., Crawford, T. J., Beaumont L. R., & Ownby, A. C. (1979). Typewriting: Learning and instruction. Palo Alto, CA: South-Western.

Rowe, J. L., Lloyd, A. C., & Winger, F. E. (1972). Typing 300. Toronto, ON: McGraw-Hill Ryerson.

Schmidt, J. B. (1983). Keyboarding: The state of the art. ERIC Document 236 352.

Shapiro, W. (1986). Five tutorials promise faster student typing. Electronic Learning, 6(3), 34.

Standish, T. A. (1980). Data structure techniques. Don Mills, ON: Addison-Wesley.

Ubelacker, S. D., Guest, R. M., & McConaghy, G. W. (1983). Mastering keyboarding skills. Toronto, ON: Copp-Clark-Pitman.

Unger, B. W., Lomow, G. A., & Birtwistle, G. M. (1984). Simulation software and ada. La Jolla, CA: Simulation Councils.

Wasylenki, L. (1985, October). Superkey (A sotware review). Computers in Education, p. 27.

West, Leonard (1983). Acquisition of Typewriting Skills. Indianapolis, IN: Bobbs-Merrill Educational Publishing.

Weston, Cynthia B. (1986). Formative evaluation of instructional materials. Canadian Journal of Educational Communication, 15 (1), 5-17.

Appendix A

Typewriting Software Reviewed by the Investigator

Name	Publisher	Year of Publication
Typing (in Business Volume 1)	Minnesota Educational Computer Consortium St. Paul, MN	1980
Gregg Keyboarding for Information Processing	McGraw-Hill Book Co. New York, NY	1982
Microcomputer Keyboarding	South-Western Publishing Co. Cincinnati, OH	1983
The Computer Keyboard	Control Data Publishing Company San Diego, CA	1983
Typing Made Easy	At Your Pace Software Mississauga, ON	1983
Keyboarding Plus (Second Edition)	Merit Audio Visual New York, NY	1984
Mastertype	Scarborough Systems, Inc. New York, NY	1984
Touch Typing for Beginners	I.B.M. Corp. Boca Raton, FL	1985
Typing for New Typists	Woodstock Software Co. Woodstock, IL	1984
Typing Intrigue	Forethought Inc. MountainView, CA	1984
Microtype: The Wonderful World of Paws	South-Western Publishing Co. Cincinnati, OH	1985
Stickybear Typing Norfolk, CT	Optimum Resource, Inc.	1985
Superkey	Bytes of Learning Toronto, ON	1985
Touch Type (Version 1.01)	Periscope Software Berkeley, CA	1985

Name	Publisher	Year of Publication
Rainbow Keyboarding	Scholastic Inc. New York, NY	1986
Success With Typing	Scholastic Inc. New York, NY	1986
Type!	Broderbund Software San Rafael, CA	1986
Type to Learn	Sunburst Communications Pleasantville, NY	1986
Typing Tutor IV	Simon and Schuster Inc. New York, NY	1987

Appendix B

The Straight Copy Typing Material

Paragraph 1

All computers are made up of many units. One such unit is called the central processing unit. It is used to control all of the other parts in a computer. As such it is fitting that the word "central" is in its name.

Paragraph 2

Computers can run a lot of different application programs. Of course many people use computers to play games. The graphics and sound effects in some games are quite spectacular. However, people do not always favor games with fancy graphics and pleasing sounds. In fact, some games display only text.

Paragraph 3

Prolog is a computer programming language used in many artificial intelligence programs. The name "Prolog" was derived by combining the first three letters of the words PROgramming and LOGic. The type of logic from which Prolog was derived is "First Order Predicate Logic." To be even more technical Prolog was actually based on the subset of first order predicate logic known as "Horn Clauses." With respect to classification, each horn clause is either a denial, an assertion or an implication.

Appendix C

The Instructional Materials used to Train the Inexperienced Markers

(Reduced to 75% of Original Size)

Marking Text Produced During Straight Copy Typing Exercises

General Idea: Proofread the text and cite as an error any departure from perfection.

To cite an error, you simply draw a circle around the word that was typed incorrectly.

Example # 1

Original: The house on the corner is well constructed.

Entered: The house on teh corner is well constructed.

A General Marking Rule

Count only one error per word, no matter how many errors it may contain!

What is a word?

A word, in this context, is a word, as typically defined, and the punctuation and spacing that follow it.

A list of the *words* in the statement below appear subsequent to it.

If I get started today, I could finish by tomorrow.

'If '	'I '	'get '	'started '	'today, '
'I '	'could '	'finish '	'by '	'tomorrow.'

Specific Marking Rules

- Count a word as an error if the punctuation is incorrect.
- Count a word as an error if the spacing is incorrect.
- Count consecutive words that are omitted as one error.
- Count consecutive words that are repeated incorrectly as one error.

Examples 2 - 5

Original: I like to hang around groovy groovy people. So too does Lou Rawls.

Entered: I like to hang around groovy groovy people, So too does Lou Rawls.

Entered: I like to hangaround groovy groovy people. So too does Lou Rawls.

Entered: I like to hagn around groovy groovy people. So does Lou Rawls.

Entered: I like to like to hang around groovy groovy people. So too does Lou Rawls.

Practice

Find the error(s) in each of the statements, if any.

Intended: The house of the corner is well constructed.

The houseón the corner is well constructed.

The house on the corner is wel l constructed.

The house on th ecorner is well constructed.

The house on the corneri s well constructed.

The little green house on the corner is well constructed.

The corner is well constructed.

The house on the corner is well consgracted.

the hou se pm yje fornst os as well xomdtj en

The house on the corner is well constructed

The house on the corner is well constructed.

The house on the

Appendix D

Joint Ethics Committee - Approval Notification

(Reduced to 75% of Original Size)

THE UNIVERSITY OF CALGARY

EDUCATION JOINT RESEARCH ETHICS COMMITTEE

CERTIFICATION OF INSTITUTIONAL
ETHICS REVIEW

This is to certify that the Education Joint Research Ethics Committee at The University of Calgary has examined and approved the research proposal by:

Applicant: Ken Luterbach

of the Department of: Curriculum and Instruction

entitled: Investigating Keyboarding Marking Methods

(the above information to be completed by the applicant)

May 20, 1987
Date

W. R. Unruh
Chair, Education Joint Research
Ethics Committee

Appendix E

Actual Error Tally Sums

and

Error Tally Sums from all Marking Sources

Note: Each row contains the actual total number of errors and the total number of errors assigned by the thirteen marking sources for one of the participating typists.

A - Actual
W - WoBat Marker
EM - Experienced Marker
IM - Inexperienced Marker

A	W	EM	EM	EM	EM	EM	EM	IM	IM	IM	IM	IM	IM
6	6	2	4	4	4	5	6	5	5	2	6	6	6
6	6	6	4	6	6	5	6	6	4	4	7	5	6
8	10	7	7	8	6	7	7	8	7	7	7	6	7
6	6	5	5	5	4	5	6	6	6	3	6	5	5
10	10	6	8	9	10	7	9	8	8	7	8	7	7
5	5	4	4	4	4	4	4	4	4	4	5	5	4
4	4	2	2	3	1	3	3	2	3	3	2	2	3
6	6	4	5	6	5	5	6	4	5	4	6	4	5
9	9	8	7	8	8	9	9	8	6	4	7	6	7
12	12	8	9	10	12	9	12	11	10	8	10	9	7
6	6	4	2	3	3	4	4	4	5	3	5	5	4
1	1	0	0	1	0	0	1	0	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	9	8	7	8	9	7	9	9	7	5	8	5	7
5	6	4	3	4	3	4	4	3	3	1	2	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	1	1	1	1	0	0	0	0	0	0
37	41	34	29	34	34	34	35	33	32	33	33	35	34
0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	11	9	11	10	12	10	11	11	11	8	11	11	12
6	6	4	4	5	5	4	4	4	5	4	2	4	4
1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	3	2	2	3	2	1	3	2	2	2	2	3	3
2	2	2	1	1	1	2	2	2	1	1	0	2	1
6	6	5	6	6	6	5	6	5	6	6	6	6	6
6	6	5	5	6	5	3	6	6	5	6	6	5	6
5	5	4	4	4	3	5	5	4	4	5	5	4	2
12	12	11	10	11	11	11	11	9	11	6	8	10	8
8	8	7	6	8	8	8	8	8	8	7	7	8	8

A	W	EM	EM	EM	EM	EM	EM	IM	IM	IM	IM	IM	IM
12	11	8	8	10	10	10	9	7	9	7	8	9	9
37	39	32	32	33	31	35	33	29	33	34	32	33	34
6	6	5	4	4	6	4	6	5	4	3	3	4	6
5	5	4	3	4	4	3	5	5	5	5	1	4	4
10	10	9	9	9	9	9	9	8	9	5	7	10	8

Appendix F

WoBaT Marker Scoring Times in seconds

Note: Each row contains the time it took the WoBaT Marker to score each of the three paragraphs generated by one of the participating typists.

Paragraph A	Paragraph B	Paragraph C
2.20	2.70	4.40
2.10	2.50	4.30
2.20	2.50	11.50
2.10	2.70	4.40
2.10	2.50	4.40
2.10	2.60	4.50
2.10	2.50	4.50
2.10	2.60	4.40
2.10	2.70	4.40
2.20	2.70	4.20
2.30	2.80	4.20
2.10	2.50	4.40
2.10	2.50	4.00
2.20	2.50	4.60
2.30	4.40	4.50
2.10	2.50	4.00
2.10	2.50	4.30
3.70	11.90	10.50
2.10	2.50	4.00
2.10	2.50	4.40
2.10	2.60	3.90
2.10	2.50	4.50
2.10	2.40	4.20
2.10	2.50	4.30
2.30	2.50	4.10
2.30	2.70	3.90
2.10	2.60	4.00
2.10	2.50	4.40
2.20	2.70	4.00
2.10	2.50	4.30
3.20	2.70	4.30
2.20	2.70	17.40
2.10	2.50	4.30
2.10	2.50	4.20
2.10	2.60	5.60

Appendix G

WoBaT Marker Questionnaire Data

Note: Each row contains the raw score responses for one of the participating experienced teachers.

Questionnaire Item Numbers

1	2	3	4	5	6
5	1	2	1	1	3
4	1	1	1	1	2
4	1	1	1	1	4
5	1	1	1	2	2
4	1	1	1	1	2
3	1	2	2	2	5