Int. J. Man–Machine Studies (1992) **37,** 1–21

A voice- and touch-driven natural language editor and its performance

ALAN W. BIERMANN, LINDA FINEMAN AND J. FRANCIS HEIDLAGE

Department of Computer Science, Duke University, Durham, NC 27706, USA

(Received 20 September 1989 and accepted in revised form 27 June 1991)

The performance of a voice- and touch-driven natural language editor is described as subjects used it to do editing tasks. The system features the abilities to process imperative sentences with noun phrases that may include pronouns, quantifiers and references to dialogue focus. The system utilizes a commercial speaker-dependent connected-speech recognizer, and processes sentences spoken by human subjects at the rate of five to seven sentences per minute. Sentence recognition percentages for our expert speaker and for subjects, were 98 and around the mid 70s, respectively. Subjects had more difficulty learning to use connected speech than had been the case in earlier experiments with discrete speech.

Introduction

The Voice Interactive Processing System VIPX (Biermann, Fineman & Gilbert, 1985; Biermann & Gilbert, 1985) offers the user the ability to display and manipulate text on a computer terminal using spoken English supported by touches to the screen. The user can maintain continuous direct eye contact with the text of interest while pointing and issuing commands to achieve the desired goal. The linguistic facilities of the system enable the user to benefit from many conveniences of natural language including a focusing capability, pronouns, quantifiers and other features. This paper describes the characteristics of the system and the performance it delivers to human users.

The VIPX system was the second voice natural language system built in our laboratory and its design benefitted from our experience. The first system (Biermann *et al.,* 1985) was called VNLC and it enabled users to speak discrete speech sentences, where a 300 ms pause is required after every word, in the solution of matrix problems. That system enabled users trained over a period of 2 h to speak acceptable inputs, and they could be left alone to solve problems in a relatively comfortable and efficient manner. Users input sentences at the rate of approximately three per minute and 77% of those sentences were processed immediately and correctly. Most sentence failures were the result of speech recognition errors or user mistakes. The VNLC design was conservative in that it displayed every input word as it was recognized, for user verification and for user modification where necessary.

The VIPX system is more ambitious. It allows the user to speak connected speech and it omits the display of recognized input for visual confirmation. The hope has been that the recognition rate will be high enough so that the user will think only about task domain issues and will be able to hold eye contact on the domain-related

0020-7373/92/070001 + 21$03.00/0

objects without distraction. The system also includes a focusing feature and it deals with a more complex domain, text manipulation.

The inputs to the system are described here as "natural language" even though they are highly restricted as to variety of syntax and size of vocabulary. The term is probably appropriate because users do not need to memorize vocabulary or legal syntactic forms explicitly, and can, with minimal training, depend on their knowledge of English to guide them to correct usage of the system. However, the reader should understand that a variety of correct English sentences will be rejected by the system. One of the purposes of this study is to discover how quickly and how effectively, in fact, do users adapt to the restrictions of the processor.

The contribution of this paper is to inform interested researchers of the level of performance achievable by a system consisting of a state-of-the-art natural language processor combined with best quality commercial speech recognition and touch-sensitive display hardware and to indicate the nature of the problems associated with it. Specifically, the paper presents results on the learnability of a machine-recognizable voice dialect, the level of correctness achievable with current technologies, the rate at which commands can be entered and tasks completed, and user reactions to such a system. The following sections describe the related literature, the VIPX system, the design of the experiment, the results obtained and conclusions.

Related work

A number of speech recognition systems (Barnett *et al.*, 1980; Brown & Vosburgh, 1989; Chow *et al.*, 1987; Erman & Lesser, 1980; IBM, 1985; Kubala *et al.*, 1988; Lee, 1989; Levinson & Rabiner, 1985; Lowerre & Reddy, 1980; Pierrel, 1981; Walker, 1980; Wolf & Woods, 1980) have been built in recent years. Some of these projects have reported high sentence recognition rates (over 90%) with large vocabularies (several thousand words). However, such recognition systems have not been built into real-time task-oriented processors and actually used by subjects in the completion of tasks. Many of them are not capable of real-time performance and few of them have ever been embedded in task-oriented natural language systems. As explained in Biermann *et al.* (1985) any system performance may be substantially worse for speakers who are concentrating their efforts on domain-related concerns and it is important to test systems under these conditions. Our project is concerned with doing this.

One project that has incorporated voice into a task-oriented system is the "Put-That-There" system (Bolt, 1980) which used a speech recognizer and a motion-sensing device to enable users to navigate information space at their media terminal. They combined voice input and motion sensing attached to the hand to enable joint voice and gesture recognition and execution of commands similar to those of our system. Our system has a more sophisticated language capability and is applied to the more ubiquitous task of text manipulation. We suspect our results are predictive of what would be obtained if similar tests were made on this type of system.

Another recent project aims at the voice-interactive word-processing problem (Thompson & Laver, 1987) but operational statistics have not yet been reported for this system.

There is considerable optimism that large-vocabulary real-time speech recognizers will soon become available which exhibit very high robustness and reliability. Some researchers expressing such opinions are Bernstein (1987), Kaneko and Dixon (1983), Sekey (1984) and Waibel (1982). Early tests on such a system (20 000 word vocabulary) to process discrete speech are described in Brown and Vosburgh (1989) and Danis (1989). As such improved recognizers become available, it will be possible to use them with the VIPX and similar systems to obtain correspondingly improved performance. Our project uses the best available voice processor for our application at any given time, and as better systems become available, we are quick to move them into our application.

Several projects have examined voice interactive office automation applications with single word rather than natural language commands (Mel, 1983; Morrison *et al.*, 1984). Such systems have been shown to be roughly comparable in performance to corresponding typed input systems with the voice systems sometimes losing and sometimes winning time comparison tests, depending on the details of the test. Our emphasis has been on developing a natural language system, and we are not prepared to make any but the most casual comparisons with typed input systems.

Another non-natural language voice input system was developed to augment a VLSI design system by Martin (1989). Voice was used as an addition to traditional input modes to increase user productivity.

A reasonable way to study human factors issues for systems proposed for the future is to build a similator and examine human performance in the proposed environment. Examples of such a studies are those described by Gould, Conti and Hovanyecz (1983) and Newell *et al.* (1980) in which humans were given the task of composing letters using a "simulated listening typewriter". The results indicated that such a system would probably increase the efficiency of letter writers and would be preferred by many users. Another simulation study has been reported by Hauptmann and Rudnicky (1988) which included comparisons of speech modes to humans and to computers with typing. It was noted in this study that people tend to speak more well-formed sentences to machines than had been predicted. While results derived from such simulations are useful, we suggest that the most reliable measurements of speech systems will come from tests of a fully implemented system.

The VIPX system

The VIPX system implements a class of imperative sentences for manipulating text on a display screen. Such sentences begin with an imperative verb and include zero, one, or two operands as in

	"Back up,"	(zero operands)
	"Delete—,"	(one operand)
and	"Change—to—."	(two operands)

The operands are noun phrases such as

	"the title,"
	"the last paragraph,"
or	"the third green word in that sentence."

The noun phrases may be accompanied by touches to the display screen as with

"Capitalize this word," (with one touch)
or "Capitalize these words." (with many touches)

The system also includes a focus mechanism that enables the user to reference objects within the current domain of discussion without wordy explicit specifications. Thus one can say

"Center the title,"
"Capitalize the first letter in each word."

and the system will capitalize only the first letter of words in the title. Or one can say

"Center the title,"
"Color it red."

In each case, an earlier sentence specifies an object or set of interest and the later sentence operates within this context. Finally, the system processes quantifiers to enable the reference to large sets of objects in a single sentence as in

"Indent each paragraph,"
"Capitalize the first letter in each word in each title."

Some inputs to the system must be typed because of the limited total voice vocabulary. For example, if the user wishes to retrieve a file named Z3, he or she can say

"Retrieve what I type,"

and then type "Z3". If the user wishes to change a word, a way to do this is to say

"Change this word to what I type,"

and then to type the revised version of the word.

Finally, for the sake of maximizing voice recognition rates and for the purposes of guaranteeing that the machine and human are in proper synchronization, the system requires that users precede each command with the word "now" and end it with the word "over." These words enable the system to identify with assurance the beginning of each sentence. They also enable the user to pause in midsentence without worry that the machine may proceed prematurely with sentence execution.

The version of VIPX used in the experiment described below had a vocabulary of 81 words including 17 imperative verbs, 20 adjectives and ordinals, 17 nouns, two pronouns and others.

The VIPX system is composed of four sequential processors that (1) acquire input, (2) parse the input and create a semantic representation, (3) execute the sentence meaning and (4) display the results. Some of the more interesting aspects of these modules will be described here.

ACQUIRING THE INPUT

Any voice recognition system with sufficient vocabulary capabilities can be used as the input processor, and this project uses the best equipment for the application at

any given time. This experiment was carried out using a Verbex 3000 speaker-dependent connected speech recognition machine with the capability of handling a total vocabulary of about 100 words. This recognizer uses an internal grammar, an approximation to the more complex VIPX grammar, to predict each next word in an utterance to increase recognition accuracy. Thus each incoming word is compared only with legal words in the given slot instead of with the complete vocabulary. Because of the speaker dependence, each user must register many samples of all vocabulary items with the machine. The user must also speak examples of word sequences to enable the acquisition of word junctures that will appear in normal connected speech.

The touch inputs to the display are registered by a grid of infrared beams that cross the screen. Other technologies are available for registering graphical inputs such as the popular mouse, and we have no great preference for one over the other. We are attracted to the "naturalness" of the direct touch rather than to the indirect control of a pointer through a mouse but we suffer some loss in pointer definition because of it. We found users can easily reference a word on the page with touch input but must be somewhat careful to designate an individual character. A standard keyboard enables the user to input text when needed.

PARSING INPUT AND CREATING A SEMANTIC REPRESENTATION

The VIPX parser is an augmented transition network (Woods, 1970) system which accounts for grammaticality by finding a legal path through a graph and which creates a meaning structure by executing functions attached to the graph transitions. These semantic functions have the tasks of finding the objects referenced in the noun phrases and executing the actions specified by the imperative verbs. The details of the processor are described at length in Biermann, Fineman and Gilbert (1985). That paper gives a *micromodel* of the system that represents all of its essential mechanisms so that the reader can reprogram it for his or her own application.

PROCESSING FOCUS

The focus mechanism is implemented as a stack that holds at sequential levels more and more local information. Thus when the user loads a text file into the system, the lowest level of the focus stack references the whole document.

1. (document)

Then as other entities are referenced, they are added to higher levels of the stack. Suppose the user next says,

"Center the title,"

where the title is "rain forests". Then the stack will become

2. rain forests (title)
1. (document)

If the person next says

"Capitalize the first letter in each word."

then the stack will become

4. R, F (letters)
3. Rain, Forests (words)
2. Rain Forests (title)
1. (document)

The focus mechanism works as follows: If the incoming noun phrase specifies objects available on the highest level of the stack, they are selected and returned as the noun phrase referent. If no objects are found, the next level is checked. Each lower level of the stack is checked until either a satisfactory resolvant is found for the noun phrase or until all levels have failed. In the latter case, the system is not able to process the user's request and an error message is returned.

A few examples will illustrate how this stack works. Assuming the focus stack has the form given above, a person could say, "these letters" or "them" and reference the first letters of the words "Rain" and "Forests" of the title. Thus either of the sentences

"Color these letters red,"
or "Color them red,"

would result in the coloring of those two letters. If a person says

"Color the third letter red,"

then level 4 of focus fails, the stack is popped and the third letter at this level would be selected. It would be the letter "i" in "Rain". If the user said,

"Color the last word red,"

the mechanism would find no words at level 4, but would succeed by finding "Forest" at level 3. If the user had said,

"Color the last paragraph red,"

then levels 4, 3, and 2 would all fail. If level 1 has any paragraphs, the last one would be selected and colored.

In summary, the VIPX processor will properly execute sentences of the form described above utilizing the focus stack to help disambiguate short noun phrases. As long as the grammar and vocabulary constraints are not violated, sentences will be processed correctly (usually in about 2s). If the user fails to speak utterances within these guidelines, an error message is provided. If the user speaks correctly but the input device misrecognizes the speech, the system either gives an error message or fails to respond at all.

The experiment

A. INTRODUCTION

While the technologies now exist to build voice and touch natural language editors, the question arises as to what performance they may be able to deliver. The purpose

of the testing program was to use subjects not experienced with voice equipment:

(1) to obtain human factors data related to the learnability of the system, speed of interactions, recognition rates and user reactions, and
(2) to gain intuition concerning the rate at which work can be done with such a system.

B. APPARATUS

The VIPX system was set up in an experimental room: a color display terminal with a touch-sensitive screen connected to a nearby mainframe computer. Near the subject work area, a terminal was set up for the experimenter who would start the system on each task and administer the experiment. A tape recorder collected all user utterances and the machine internal clock time stamped all interactions.

C. METHOD

Three tests were designed to gather different kinds of information. The first (Test 1) required the user to use natural language and touch inputs to do atomic tasks that are equivalent to single standard keyboard operations. These tasks involve such operations as deleting or modifying a single word. Here noun phrases in the natural language were singular as in "this word" (with touch) or "the last paragraph," and they were often accompanied by a touch input designating the object to be affected. The second test (Test 2) allowed the user to use more powerful noun phrases which designate sets of objects. These noun phrases are plural, and they select operands that cannot be specified by traditional text editors (such as "the first letter in each word"). Such commands can in one utterance make an unlimited number of modifications to a document. Both Tests 1 and 2 required that a single operation type be performed repeatedly so that accurate timing measurements could be made on each operation type. Test 3 allowed a subject to do typical editing tasks with a mix of operations to gather information in a more natural environment and is described in the next section.

Test 1 provides one-command-at-a-time data so that comparisons can be made with other editing systems. Test 2 shows off the natural language capabilities where the amount of work per command is not comparable to ordinary text editing. Test 3 gives performance data when a mixture of command types is used.

The format of the experiments was similar to that followed earlier (Biermann *et al.*, 1985) in the test of the discrete speech natural language processor. Before beginning the experiment, the subject was required to register multiple examples of each word in the VIPX vocabulary with the voice recognition machine. Then the experimenter read instructions related to the task to be performed and allowed the user to speak sample sentences to the system and observe its response. Finally the subject was released to do the experimental task. After completion of the task, instructions were given for the second task, the subject would try out additional voiced commands, and then the second experimental set would be done. Four tasks were given following the same format. The fourth one was more complicated than the others in that it combined several earlier ones. (A fifth task was given to calibrate the touch facility but it need not be described here.) Subjects were also asked to do the same tests with a standard typed editor, the UNIX VI system, to

obtain a very rough idea of how long the tasks might take with a traditional method. The experimenter was on hand throughout the test to answer questions that might arise relating to misrecognitions or peculiarities of the system. Data was recorded concerning the numbers of such interactions and is reported below.

In order to indicate how high the performance level can be, one of our laboratory personnel was asked to do Test 2, and those data are also reported.

Test 1: One action per voice command

This test comprised the following four tasks:

Task 1. Deletion of words. The subject was asked to delete all proper nouns on a specified page. This was done with the sentence "Now delete this word over," accompanied by a touch input or some paraphrase of this sentence.

Task 2. Modification of words. The subject was asked to change certain words in a specified way. This was done with the sentence (or some paraphrase of) "Now change this word to what I type over," accompanied by a touch input and typing. (Note: "what I type" was used on VIPX to specify any string of characters input at the keyboard.)

Task 3. Re-arranging of sentences. The subject was asked to re-arrange the sentences on a page according to a given specification. The VIPX command might be: "Now put this sentence after that sentence over," accompanied by two touches.

Task 4. Combination. The subject was required to do a composite task including samples from the earlier tasks. The user also spoke the commands "Now backup over," or "Now clear touches over," to undo an action or clear the touch indicators from the screen.

Test 2: Several actions per voice command

The second test was designed to allow the voice natural language system to demonstrate one of its strengths, the ability to easily reference multiple objects in a single command. The four tasks were as follows:

Task 1. Deletion of words. Here the subject could say "Now delete these words over," and touch as many items on the screen as desired.

Task 2. Multiple deletion of characters. The subject was asked to delete all of the capital letters on a page. The voice command was "Now delete each capital letter over."

Task 3. Capitalizing the first letter in words. The subject was asked to capitalize the first letter in each word in all titles of a document and to capitalize the first letter in each sentence. The required commands (or their paraphrases) were "Now consider this title over," (with touch input) "Now capitalize the first letter in each word over," and "Now capitalize the first letter in each sentence over."

Task 4. Capitalizing titles and indenting paragraphs. The subject was asked to sequentially load three files and capitalize all titles and indent all paragraphs. VIPX commands: "Now capitalize each title over" and "Now indent each paragraph over."

D. SUBJECTS

The subjects were volunteer Duke students from a variety of majors. Fourteen were used in Test 1 and 14 different subjects were later used in Test 2. Many had had

occasional experience with word processing systems but none with a voice editor or the VI editor used in this experiment. An expert speaker from our laboratory also carried out Test 2 in order to indicate the capabilities of the system.

E. SPECIFIC DIMENSIONS TO BE MEASURED

Our goals in these tests were to gather information on the following issues. We were interested in these issues both as absolute entities and in comparison to observations already made with our previous system using discrete speech (Biermann *et al.*, 1985).

1. Learnability

We wanted to know whether subjects could easily learn to speak machine recognizable connected speech. Specifically, we wondered whether the easy learnability previously observed with discrete speech would be repeated with connected speech.

2. Correctness

We wanted to know whether subjects could complete tasks with the system. We also wanted to know whether a later generation of recognizer would do a substantially better job of recognizing speech and whether the change to connected speech would greatly reduce recognition rates.

3. Timing

Another extremely important observation relates to timing and the rate at which the work gets done. How fast would subjects speak sentences and how fast would tasks be completed.

4. User response

How would users feel about the experience of voice control of the machine? What strong points of the system would they appreciate and what complaints would they make?

F. RESULTS AND ANALYSIS

The results of the tests are given primarily in terms of timing information and error statistics. Table 1 gives the average command completion times for the tasks. The elapsed time for a command includes the time to find the next object to modify, to utter the command and to see the result displayed on the screen.

An obvious question is to ask what these timings would be using a traditional editor? Unfortunately, a fair comparison between such diverse types of systems is impossible. The natural language system uses verbose commands, touch input and suffers from command failures because of voice misrecognition. The traditional editor uses single keystroke commands, requires the cursor to be moved to each new point of change and fails only when the user hits the wrong command key. Still, an order of magnitude comparison is interesting to give a rough idea of the comparative rates at which work can be done under the constraint that the natural language system was allowed to reference only one object at a time. Thus data was gathered

TABLE 1
Average time (s) per command for the voice and conventional editors

Task		VIPX	VI
1	Deletion of words	9·8	8·3
2	Modification of words	13·1	12·2
3	Re-arranging sentences	12·0	13·1
4	Combination of above	13·4	14·3

in an environment identical to the one for the voice tests with the same subjects, similar training and the same timing and scoring procedures.

For the purposes of classifying errors, the spoken inputs of the subjects to VIPX were separated into categories as follows:

User success, system success
Success-unqualified. A totally successful transaction in which all input components were present, and the computer carried out the desired instruction.
Misrecognition but success. A transaction in which the voice recognizer substituted a word for one actually spoken by the subject, but the resulting sentence was intelligible and the computer performed the desired action.

User success, system failure
Misrecognition failure. A transaction in which the voice recognizer output an utterance different from the one actually spoken by the subject resulting in a sentence nonsensical to the VIPX parser.
Non-recognition failure. A transaction in which the speech recognizer could not interpret the utterance within the constraints of its grammar and consequently did not produce any output to the VIPX parser.
System failure. A transaction in which a correct subject input was correctly recognized, yet the desired response was not generated.

User failure
No-touch error. A transaction in which the utterance was correctly interpreted by the speech recognizer and the computer, but which could not be carried out because the subject neglected to make a necessary entry via the touch screen.
User error. A transaction in which the subject's utterance did not conform to the requirements of the system. This includes starting over in the middle of a sentence, voicing an utterance which was not in the grammar, or leaving off the initial "now." (If the subject omitted the word "over", the experimenter would prompt for it; this usually resulted in a successful transaction.)

Table 2 gives the overall summary of recognition statistics for Test 1 using these seven categories. In addition to the tasks described above, the users spoke many utterances practising system usage, or to retrieve or store the test files. To obtain the initial file for each test, the subject uttered "Now retrieve what I type over." and keyed in the filename for the test. To store a file, the subject would say "Now store

TABLE 2
Test 1 sentence recognition statistics summary

Category	Within task Number	%	Overall Number	%
USER SUC, SYS SUC				
Success-unqualified	1087	75·2	1548	67·4
Misrecognition but success	0	0·0	6	0·3
USER SUC, SYS FAIL				
Misrecognition	135	9·3	294	12·8
Non-recognition	179	12·4	370	16·1
System failure	2	0·1	7	0·3
USER FAIL				
No touch error	13	0·9	17	0·7
User error	29	2·0	54	2·4
Totals	1445	99·9	2296	100·0

the document in what I type over." The additional utterances spoken outside of the timed tasks were also scored, so both the within-task and overall statistics are given.

A major discovery in this series of tests was that using machine-recognizable connected speech is substantially more difficult for subjects than using machine-recognizable discrete speech with word-by-word feedback. In our earlier experiments with discrete speech (Biermann *et al.* 1985), subjects were given a tutorial session and then left to operate the system without further intervention. With the current system, we found such a short tutorial inadequate and subjects were not generally able to function independently. The typical subject would experience a misrecognition early in the test sequence and be confused by the lack of proper response. This would lead to repeated commands with substantially varied vocabulary, raised pitch and volume and a general divergence of behavior from that learned in the tutorial. Our method for repairing the problem was to place the experimenter beside the subject and give the subject error messages and additional tutorial comments to guide him or her toward successful interactions. While this is clearly an undesirable experimental methodology, it was, at the time, found to be necessary to enable subjects to proceed. The number of experimenter–subject interactions was counted for the complete test, and the within-task average was found to be 20·8 per subject. This means that, with each subject speaking about one hundred sentences, a helpful comment from the experimenter came on the average once every five sentences. A better experimental procedure was later developed as explained in Test 3 below. (It turns out that the subjects also needed occasional help on the traditional editor, 8·9 times per subject on average.)

The results of Test 2 are given in Tables 3 and 4 in the same format as those of Test 1. Comparisons with a traditional editor are not very meaningful here because each English sentence can modify an unlimited number of items.

Table 4 gives the error statistics in this test. They were very similar to Test 1 except that the number of misrecognitions was nearly cut in half. This is because some hard-to-recognize sentences in Test 1 did not appear in Test 2. For example, the sentence "Now change this word to what I type over." was spoken 503 times in

TABLE 3

Average time (s) per command for subjects and the expert. The average time for VIPX commands was often long because they were handling multiple objects. Thus in 1, *users would request word deletion and then take considerable time touching all the words to be removed*

Task		Subjects	Expert
1	Multiple deletion of words	65·1	87
2	Multiple deletion of characters	7·9	6·5
3	Capitalizing letters	8·8	6·8
4	Capitalizing, indenting	6·9	6·1

Test 1 with a recognition rate of 66·2%. This sentence did not appear in Test 2. Also the sentence "Now store the document in what I type over." which was recognized only 31·2% of the time in Test 1 was shortened to "Now store over." in Test 2, yielding a success rate of 90·7%.

Test 2 was administered by a less verbose experimenter than in Test 1 and the number of within-task experimenter–subject interactions dropped to an average of 5·8 per subject or approximately one helpful comment per 10 input sentences.

Subjects indicated their level of satisfaction with the system on an exit interview form by indicating agreement or lack of agreement with a series of statements. Each statement could be marked with a value from 1–5 where 1 indicated maximum disagreement with the statement and 5 indicated maximum agreement.

Subjects were also encouraged to write down remarks related to the system and those remarks are reported in the appendix.

G. DISCUSSION

Learnability

Subjects had substantially more trouble learning to speak machine-recognizable connected speech than they had learning to speak discrete speech with word-by-

TABLE 4

Test 2 *sentence recognition statistics summary. The expert's are given in parentheses*

Category	Within task Number	%	Overall Number	%
USER SUC, SYS SUC				
Success-unqualified	661(53)	80·6(98·1)	1446(109)	76·1(99·1)
Misrecognition but success	7	0·9	10	0·5
USER SUC, SYS FAIL				
Misrecognition	42	5·1	117	6·1
Non-recognition	95(1)	11·6	290	15·3
System failure	1	0·1	2	0·1
USER FAIL				
No touch error	1	0·1	1	0·1
User error	13	1·6	35	1·8
Totals	820	100·0	1901	100·0

TABLE 5
Users' responses on a scale from 1–5

Statement	Level of agreement (out of 5)
I enjoyed learning the editor	4·52
I found it easy to learn	4·34
I found it easy to use	4·30
I found it tiring to use	2·17
I preferred using the typing editor to using the talking editor	2·43

word feedback as in earlier experiments with the VNLC systems. In the previous case (Biermann *et al.*, 1985), the tutorial session was sufficient to enable subjects to use the system without additional experimenter aid during the experimental task. With connected speech, the subjects required occasional added instruction during the task in order to achieve success.

The reasons for having more difficulties in learning connected speech in this environment than with discrete speech in VNLC can be attributed to several factors.

1. In training subjects to speak discrete speech, the requirements of a space after every word is easy to explain and encourages a stiff mechanical enunciation which machines can recognize. In training connected speech, the experimenter does not know what to tell the subject. Any advice along the lines of speaking more slowly or more distinctly could lead to reduced recognition. The best strategy is to speak naturally yet distinctly in both the voice registration and system usage phases. But this is a difficult strategy for the experimenter to explain and for the subject to carry out. Some subjects assumed that speaking "naturally" meant mumbling along in the most casual manner.
2. The lack of incremental feedback at the word-by-word level in connected speech, made it difficult for the subjects to converge on acceptable behavior. Unfortunately, such detailed information is not necessarily easy to provide with a connected speech system since, with a failed sentence, the system may have numerous hypotheses as to what was said and a presentation of them may not help. Our solution in these tests was to have the experimenter provide the error messages and suggestions for improvement.
3. The word processing domain of VIPX is more complex than the matrix computation domain of VNLC. The semantics of the hierarchy of the text domain led to a greater syntactic variety of input sentences which was more difficult for a machine to cover.

We view these observations to be extremely important and a long-term problem for the development of voice-driven systems. If voice input is used in any environment, the recognition system will need to return error messages that effectively encourage the user towards machine-recognizable speech. For discrete speech, this is a tractable problem but for connected speech, it is not, for the time being.

Correctness
All subjects were able to do all tasks correctly. However, the system failed to process approximately 20–25% of sentences properly because of speech recognition errors. The expert speaker achieved a 99% sentence recognition rate. These results are astonishingly similar to those obtained with discrete speech where Biermann *et al.* (1984) reported 77 and 90% recognition rates for subjects and expert respectively. The similarity of these figures has led us to wonder whether subjects have a certain tolerance for error of around 20–25% and that they speak faster and faster until that rate of error is reached. Perhaps the same error rate will be observed for wide variations in the quality of the recognition system.

Timing
Subjects were able to enter sentences at the rate of about five to seven per minute whereas with discrete speech the rate had been about three per minute. This improvement is what should be expected from the more streamlined mode of speech. Our informal comparisons between our voice system and a traditional editor indicate that where comparability in atomic operations exists, the two modes seem to give comparability in timing results.

User response
Users were extremely interested in doing this rather exotic task and responded with enthusiasm. They overwhelmingly enjoyed learning and using the system. They, however, expressed some reservations related to the exacting requirements of the voice recognizer for successful operation. It is not easy to predict what long-term behaviors would be with such a system.

Similar results were obtained with the discrete speech system (Biermann *et al.*, 1985) where average scores were reported as follows. The scores varied from 1 (indicating the highest degree of disagreement) to 7 (indicating the highest degree of agreement).

Enjoyed learning system	6·7
Found learning easy	6·3
Enjoyed using the system	6·3
Found use easy	5·0
Found system tiring	3·1
Prefer the voice system to typed input	5·7

System performance in a format-free session

A. INTRODUCTION

While the above tests give detailed data on specific tasks, the question remains as to what typical behavior might be in ordinary usage of the voice-driven editor. Test 3 was designed to give a sample of user behavior in less constrained tasks with more user decision making and a mixture of commands. It was also run longer in order to observe user learning.

B. APPARATUS

The experiment was carried out following the same format as that described for Tests 1 and 2.

C. METHOD

A single subject was trained using a tutorial that was an amalgamation of all parts of Test 2 with some additions and then released to edit 14 single page documents. The instructions for these 14 tasks were simply to format the given page identically to a model document. The model contained a title, an indented abstract and several paragraphs all left and right justified.

A summary of the various operations needed to complete each task is given here:

1. Move abstract paragraph; indent it; center title; capitalize first letters, right-justify the document, delete the Xs. (As an exercise in deletion, sequence of Xs were randomly placed on the document.)
2. Center title; move the abstract paragraph; right-justify the document; delete the Xs.
3. Center title; capitalize first letters; capitalize "abstract"; indent abstract paragraph; capitalize the first letter in each sentence in paragraph two; right-justify the document.
4. Capitalize first letters of title; move abstract paragraph; switch sentences in abstract paragraph; right-justify the document.
5. Center title; capitalize first letters; capitalize "abstract"; indent abstract paragraph; capitalize the first letter in each sentence in the document; delete Xs; right-justify the document.
6. Center title; capitalize first letters, capitalize the first letter in each sentence in the document, capitalize "abstract"; indent abstract paragraph; move abstract paragraph; right-justify the document.
7. Center title; capitalize first letters; switch sentences in abstract paragraph; capitalize the first letter in each sentence in the document; delete Xs, right-justify the document.
8. (repeat 2 above)
9. (repeat 3 above)
10. (repeat 1 above)
11. (repeat 5 above)
12. (repeat 4 above)
13. (repeat 7 above)
14. (repeat 6 above)

D. SUBJECTS

A single woman undergraduate Duke student was used from the same pool as described above.

E. DIMENSIONS TO BE MEASURED

The issues of learnability, correctness, timing and subject response were to be examined again, but in this case in the environment of a longer and more free format test.

TABLE 6
Summary statistics for the format-free test

Task no.	Time (s)	Number of commands	Number of seconds per command	Per cent successful commands	Experimenter interactions
1	134	8	16·8	87·5	Many
2	75	4	18·8	100	0
3	277	10	27·7	90·0	1
4	114	5	22·8	80·0	2
5	277	18	15·4	55·6	4
6	200	19	10·4	57·9	5
7	208	15	13·9	53·3	5
8	58	4	14·5	100	0
9	55	8	6·9	100	0
10	77	8	9·6	87·5	0
11	158	17	9·3	64·7	0
12	101	12	8·4	66·7	0
13	106	12	8·8	75	0
14	75	11	6·8	90·9	0
Total	1915	151	(12·7 average)		

F. RESULTS AND ANALYSIS

Table 6 gives summary statistics for the 14 tasks. The subject proceeded at her own pace doing individual parts in the desired order. The experimenter went through Task 1 with the subject in a step-by-step fashion. In all later tasks, the experimenter intervened as seemed necessary. The right-hand column shows the number of interventions in each task; there were none after Task 7. During the first eight tasks, the rate of inputs for commands tended to be slower than it had been in Tests 1 and 2. However, the last six tasks were done with remarkably high sentence rates. The subject's recognition rate dropped for no apparent reason to 55·6% in Task 6 and remained near there for two more tasks before jumping up again to more acceptable levels. This is a common phenomenon with voice systems. The subject had seen one example of a pronoun (in "Center it.") in the tutorial session. She began using this pronoun in Task 9 and used it comfortably thereafter. The subject used the focus mechanism quite effectively.

Considerable learning occurred during the 2h period of the test. This can be observed by examining the times of the final seven tasks which were copies of the first seven (but re-arranged in order). Table 7 shows the total task time for the seven tasks the first and second times they were administered.

Table 8 gives the error breakdown for this subject. Misrecognition levels were somewhat higher than the average subject in the earlier tests.

This subject reported 5, 5, 5 and 3 on the first four post-test questions given in Table 5. Concerning her likes and dislikes of the system, she wrote, "It makes editing much easier than it would otherwise be; also, it's easy to learn and fun to use." and "It's frustrating when it repeatedly misunderstands, and after a while I got tired of talking."

TABLE 7
Total required time for each task on its first and second repetition

Task	First repetition	Second repetition
1	134	77
2	75	58
3	277	55
4	114	101
5	277	158
6	200	75
7	208	106

G. DISCUSSION

Learnability
The indications are that the subject was able to converge on acceptable speech after about 100 sentences were input and after the task became somewhat repetitive.

Correctness
In the environment of a wider variety of inputs, recognition rates were slightly but not catastrophically lower.

Timing
The wider variety of inputs appeared to reduce the rate of sentences until the task became repetitive.

Subject response
No new observations were made beyond those described from Tests 1 and 2.

TABLE 8
Test 3 sentence recognition statistics summary

Category	Within task Number	%	Overall Number	%
USER SUC, SYS SUC				
Success-unqualified	110	72·9	181	70·6
Misrecognition but success	0	0	1	0·4
USER SUC, SYS FAIL				
Misrecognition	25	16·6	49	19·1
Non-recognition	14	9·3	24	9·3
System failure	0	0	0	0
USER FAIL				
No touch error	0	0	0	0
User error	2	1·3	2	0·8
Totals	151	100·0	257	100·0

Conclusions

This paper describes the Voice Interactive Processing System VIPX and the results of tests to determine its performance in editing tasks with inexperienced users. The tests only exercised a fraction of the VIPX capabilities but yielded representative statistics on its capabilities as a system.

Subjects were able to input sentences with connected speech to VIPX at the rate of about five or more per minute in completing tightly constrained tasks and obtained sentence recognition rates on the order of 75%. The majority of failed sentences were caused by misrecognition and non-recognitions by the voice recognizer. However, all subjects were able to complete all tasks with reasonable efficiency. For example, when tests were comparable with a traditional system, similar timing results were obtained.

The VIPX system has been developed to the point that it can function as a demonstration editor but many of the conveniences of a commercially developed editor are not available on it. Thus, it is quite adequate for experimental and demonstration purposes, but it is not currently being used in our laboratory as a means for producing text. The system would need additional input–output facilities, a wider vocabulary, control over fonts, superscripts and subscripts and other features to be a preferred editor for daily use.

More recent phases of our project have been to move to another application domain, equipment repair, where context information can be used to predict user inputs for better error correction and where more co-operative responses can speed up the rate at which work can be done. Our belief is that the greater semantic structure in this domain will make better performance possible than was achieved here.

In summary, present day commercially available connected speech recognition technologies are sufficient to support natural language applications where very limited vocabulary constraints can be tolerated. Users will have difficulty learning to speak acceptable connected speech and sentence recognition rates may be moderate, only about 75% in our tests. However, user performance does improve with time and sentence input rates can exceed five per minute. Overall rate of completion of work can be at least comparable to the performance of conventional systems.

References

BARNETT, J., BERNSTEIN, M. I., GILLMAN, R. & KAMENY, I. (1980). The SDC Speech Understanding System. In W. A. LEA, Ed. pp. 272–293. Englewood Cliffs, NJ: Prentice Hall, Inc.

BERNSTEIN, J. (1987). Oral comments. Workshop on Spoken Language Systems, University of Philadelphia, PA, USA May 6–7.

BIERMANN, A. W., FINEMAN, L. & GILBERT, K. C. (1985). An imperative sentence processor for voice interactive office applications. *ACM Transactions on Office Information Systems,* **3,** 321–346.

BIERMANN, A. W., GILBERT, K. C. (1985). An imperative sentence processor for office applications–demonstration, Video tape, Department of Computer Science, Duke University, Durham, NC, USA.

BIERMANN, A. W., RODMAN, R. D., RUBIN, D. C. & HEIDLAGE, J. F. (1985). Natural language with discrete speech as a mode for human-to-machine communication. *Communications of the ACM,* **28,** 628–636.

BOLT, R. A. (1980). "Put-That-There": voice and gesture at the graphics interface. *Computer Graphics,* **14,** 262–270.

BROWN, N. R., VOSBURGH, A. M. (1989). Evaluating the accuracy of a large-vocabulary speech recognition system. *Proceedings of the Human Factors Society 33rd Annual Meeting. pp* 296–300, *Denver, Colorado, USA, Oct.* 16–20, 1989.

CHOW, Y. L., DUNHAM, M. O., KIMBALL, O. A., KRASNER, M. A., KUBALA, G. F., MAKHOUL, J., ROUCOS, S., SCHWARTZ, R. M. (1987). BYBLOS: the BBN continuous speech recognition system. *IEEE International Conference on Acoustics, Speech, and Signal Processing,* pp. 89–92. Dallas, Texas, USA. April, 1987.

DANIS, C. M. (1989). Developing successful speakers for an automatic speech recognition system. *Proceedings of the Human Factors Society 33rd Annual Meeting.* pp. 301–304, Denver, Colorado, USA Oct 16–20 1989.

ERMAN, L. D. & LESSER, V. R. (1980). The hearsay-II speech understanding system: a tutorial. In W. A. LEA, Ed. *Trends in Speech Understanding,* pp. 361–381, Englewood Cliffs, NJ: Prentice Hall, Inc.

GOULD, J. D., CONTI, J. & HOVANYECZ, T. (1983). Composing letters with a simulated listening typewriter. *Communications of the ACM,* **26,** 295–308.

HAUPTMANN, A. G., RUDNICKY, A. I. (1988). Talking to computers: an empirical investigation. *International Journal of Man–Machine Studies,* **28,** 583–604.

IBM SPEECH RECOGNITION GROUP (1985). A real–time, isolated-word, speech recognition system for dictation transcription. *IEEE International Conference on Acoustics, Speech, and Signal Processing.* pp. 858–861 Vol. 2. Tampa, Florida, March, 1985.

KANEKO, T. & DIXON, N. R. (1983). A hierarchical decision approach to large vocabulary discrete utterance recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing.* **ASSP-31,** 1061–1066.

KUBALA, G. F., CHOW, Y., DERR, A., FENG, M., KIMBALL, O., MAKHOUL, J., PRICE, P., ROHLICEK, J., ROUCOS, S., SCHWARTZ, R., VANDEGRIFT, J. (1988). Continuous speech recognition results of the BYBLOS system on the DARPA 1000-word resource management database. *IEEE International Conference on Acoustics, Speech, and Signal Processing.* pp. 291–294, Vol. 1. New York, April, 1988.

LEE, K.-F. (1989). *Automatic Speech Recognition.* Boston, MA: Kluwer Academic Publishers.

LEVINSON, S. E. & RABINER, L. R. (1985). A task-oriented conversational mode speech understanding system. *Bibliotheca Phonetica,* **12,** 149–196.

LOWERRE, B. & REDDY, R. (1980). The harpy speech understanding system. In W. A. LEA, Ed. *Trends in Speech Understanding,* pp. 340–360. Englewood Cliffs, NJ: Prentice Hall, Inc.

MARTIN, G. L. (1989). The utility of speech input in user–computer interfaces. *International Journal of Man–Machine Studies,* **30,** 355–375.

MEL, A. (1987). *Talking to computers–voice recognition as a mode of interaction for executives,* Ph.D. thesis, The Wharton School, University of Pennsylvania, Philadelphia, PA USA.

MORRISON, D. L., GREEN, T. R. G., SHAW, A. C. & PAYNE, S. J. (1984). Speech-controlled text editing: effects of input modality and of command structure. *International Journal of Man–Machine Studies,* **21,** 49–63.

NEWELL, A. F., ARNOTT, J. L., CARTER, K., CRUICKSHANK, G. (1990). Listening typewriter simulation studies. *International Journal of Man–Machine Studies.* **33,** 1–19.

PIERREL, J. M. (1981). *Etude et mise en oeurre de constraintes linguistiques en comprehension automatique du discours continu,* Ph.D. thesis, University of Nancy, France.

SEKEY, A. (1984). Building a model for large vocabulary isolated word recognition. *Speech Technology,* **1,** 71–81.

THOMPSON, H. S. & LAVER, J. D. (1987). The Alvey speech demonstrator–architecture, methodology, and progress to date. *Speech Tech '87,* pp. 15–18. New York, USA, April 28–30, 1987.

WAIBEL, A. (1982). *Towards very large vocabulary word recognition.* Carnegie-Mellon University Computer Science Department, Speech Project, November 1982.

WALKER, D. E. (1980). SRI research on speech understanding. In W. A. LEA, Ed. *Trends in Speech Recognition,* pp. 294–315. Englewood Cliffs, NJ: Prentice Hall, Inc.
WOLF, J. J. & WOODS, W. A. (1980). The HWIM speech understanding system. In W. A. LEA, Ed. *Trends in Speech Understanding,* pp. 316–339. Englewood Cliffs, NJ: Prentice Hall, Inc. 1980.
WOODS, W. A. (1970). Transition network grammars for natural language analysis. *Communications of the ACM,* **13,** 591–606.

Appendix: subject comments about the system

Subjects answered some questions on their exit forms. In response to the question "What did you like about the talking editor?", the following representative answers were given. (Many subject's answers repeated their colleagues points and those are omitted.)

1. It was very interesting! I thought it was fun—to think a computer is "understanding" what I'm saying! It was easy to use—you could make changes very quickly.
2. Your fingers don't get tired as easily and you can do multiple steps in one quick action (or voice command). When doing a number of same commands, the talking editor would be faster.
3. Made computing easy enough to handle by a 5 year old. Very good concept; it's easy to work with and saves valuable time spent on a computer. I could see writing a program may no longer take so many hours. (Unreadable comments followed.)
4. You could go immediately to any place; you didn't have to deal as much with the keyboard; the commands were pretty basic and not hard to learn.
5. Basically, I liked the fact that I didn't have to touch a keyboard nor do any typing. I enjoyed the minimal effort it took to "point" out a word on the screen and tell the computer by voice commands what to edit within the text. Also, with the talking editor, one can jump around the entire text easier and quicker because a cursor did not have to be moved up and down the screen.
6. I found the talking editor to be much easier to learn and to use as well. The tasks seemed much shorter on the talking editor and also more natural—for a non-computer user as well as non-typists.
7. It wasn't boring. You could get actively involved (e.g. touching screen for word to delete) and talking makes you concentrate because you hear yourself.
8. It was so easy! Just tell it and it's done. It went much faster than I thought it would.

In response to the question "What did you dislike about the talking editor?", subjects responded as follows. (Repetitive comments are omitted.)

1. It was tedious saying "now" and "over" all the time but I guess there has to be some start and stop signal—it's not that bad. If you're not in the mood to articulate clearly, the typing editor would be better—but this one is fun—and appears to be faster.
2. The only thing that I didn't like was that it had a tendency to misread words (couldn't match pattern?) When editing a long document this could slow you down. It also would take a while to become accustomed to saying "now" and "over" in the commands.

3. Nothing. The only problem that I had (with both editors, though) was trying to remember the exact wording of each command.
4. You have to say certain phrases in exactly the same way or they don't register.
5. It took me a little while to become really comfortable with it.
6. It had a hard time recognizing commands, so I had to keep repeating, which was frustrating. It also took a lot of time to do things which could have been done quicker on the type editor because of this reason.
7. *a*) Tendency to touch wrong word. *b*) Need to repeat commands or articulate very carefully. *c*) As a writer, I felt a little more distant from actually writing—but since the machine only edits that's probably all right.
8. I disliked having to concentrate on saying one particular word "the" the same way over and over again, but this wasn't too painful. In both instances, it would have been more helpful to have had a written list of the commands, so that short-term memory would not have been enlisted. I also liked that with the talking editor you could look at the text as you edited, not at a keyboard. I suppose that if I typed better I wouldn't have to look down at my fingers anyway, but this is geared towards those who don't type too well. I really liked not having to use the cursor arrows to indicate where I wanted to edit—this touch mode is invaluable in this instance.
9. The touch editing is sometimes hard to get correct because your finger has to point straight at the words.
10. Not really anything—it was fun to work with it, although your voice does get worn out after a while.
11. I see the greatest hindrance to the talking editor being that I felt greater "pressure" to recall directions etc. than using the typing editor. It may be due to my inexperience, but I felt that if I forgot the verbal command *verbatum,* I'd mess everything up. For some reason, I didn't feel this way with the typing editor.
12. The limitations of the Verbex, i.e. setting up vocabulary. One would have to spend some initial time to get vocabulary down. It can be very frustrating to have to repeat something to a machine. The talking editor is not good for capitalizing one specific letter. But it is very, very good at doing many such tasks, such as indenting entire pages. One also only has to learn a few words to edit with the talking editor, whereas with the typing editor there are as many typing sequences (i.e. different keys to push) as there are commands.

The authors are indebted to K. C. Gilbert who developed portions of the VIPX system and provided consultation on the experimental design and paper. The original VIPX system was constructed for IBM Corporation under Grant GSD-260880.

This paper was re-organized and rewritten according to suggestions by anonymous reviewers. It has been much improved by the changes.

This research was Supported by National Science Foundation Grant No. 8803802 and IST-83-01994.
Linda Fineman is currently at SAS Institute, Cary, NC, USA.