WORKING PAPER 2:
SUMMARY OF USER ERRORS AT THE TERMINAL
by

D.R. Roa i 0. Akin

December, ;932

DRC-43-01 -6.1

Working paper: 2
Summary of user errors at the Terminal[1]

D. Radha Rao and Omer Akin.

Table of Contents

List of Tables

1. Introduction

This paper is part of an effort to develop a model to measure the level of efficiency in interfacing electronic mail systems with users at the terminal. In this paper, we define the guidelines for codifying errors made during the interaction. These findings will suggest ways of developing new systems with greater efficiency and improving the performance of existing systems.

The experimental task was made up of a set of standard tasks necessary for managing electronic mail.[2] The standard operations of Rdmail, the system we studied, includes manipulating large message files. The set of the standard tasks taken into consideration are *composing, editing, searching, reviewing, sending, filing and creating new messages.*

1. Composing: Assembling and formatting the contents of messages.

2. Editing: Editing an already existing message text, changing headers.

a Reviewing: Reviewing newly constructed or edited messages, drafts and blind copies.

4. Sending: Forwarding messages to other users of the system.

5. Searching: Searching a given message file to find particular messages by inspecting their contents, headers, source and dates.

6. File maintenance: Classifying messages according to date, content, subject

All six subjects volunteered to participate in the study were familiar with Rdmail, the electronic mail system widely used in Computer Science Department, Carnegie-Mellon University. Three subjects we considered "experts" are those who are either directly involved in the design and maintenance of the Rdmail system or use the system extensively in their daily work. The other three subjects we considered "routine users" are familiar with the system and use it on a regular basis. All the subjects were asked to perform the tasks outlined above in Rdmail.

2. Summary of Error Categories

There are two basic categories discerned from errors made at the terminal by our subjects: the *symptom* and the *source* of the errors. By *symptom* we mean appearance or form of the error. By *source* we mean the probable cause for the errors to occur. We based our categorization on the *symptom* and further elaborated each category based on possible *sources.*

---

[2] nrhe findings reported in this paper are based on an empiricaJ study observing errors by users of an electronic mail system caJted Rdmail, which is currently in use at the computing facilities of the Department of Computer Science, Carnegie-Moion University. The experiment was the same as the one described in the previous paper.

Mistyped character errors are simple typographical errors most probably caused through the mechanics of typing. There seems to be a multitude of forms these errors come in. We categorized these errors into *transposition, substitution, omitted character* and *extra character* errors.

Grammatical, natural language errors are defined as all errors made while typing text which is not in direct response to system command prompts. Text entered into files, user notes and comments are some examples of such text. The grammatical errors made in this context, then, correspond to natural language syntax errors.

Grammatical, command language errors are all errors made while responding to system command prompts. Some examples are: *mis-typing key-words or arguments* in a system defined command, and *using an inappropriate command.*

2.1. Mistyped character errors

2.1.1. Transposition
Transposition error is the reversing of character positions in a word. For example;

<div align="center">

suing : : »[3] using;

Struab:: = Straub;
seplling::• spelling;
</div>

2.1.2. Substitution
The substitution error is manifested where one or more characters) are typed while other character(s)"were intended. This error seems to generate from many different sources.

Multiple key character error, a form of substitution, is caused by typing the wrong special character key, i.e., "control", "tab", "escape", "shift", "back-space", due to the unfamiliar position of that key on the current key-board. For example;

<div align="center">

ti::» I
</div>

Another common substitution error is *inserting an unintended character* in place of the intended one. For example;

<div align="center">

h from rf ::= h from :Rao;
+73::« :73
</div>

Yet another substitution error occurs when a *special character key is not released* in time to correctly type the next character. For example;

> AFter::* After;
> THis::«This.

Shifting key position error is caused by placing the hands in a position which is one or more keys removed in either direction (up-down or side ways) from the usual hand position. This results in the shifting of each key character on the keyboard, equal to the shift of the hand(s), respectively. For example;

> svibr:: = above
> skdi::»abo[4]

2.1.3. Omitted character

These are errors that occur where one or more characters intended to be typed are completely omitted. For example;

> •d600::= *d6000

Key-position error another possible form of this error category occurs when certain keys typed in a given sequence result in the users inability to produce the adequate motion in making the appropriate key-stroke; often resulting in an insufficiently depressed key and a missing character in the text In the examples below the subject was unable to hit the [N]o" key when it occured after "m," a number of times.

> mre *n* = more;
> mve::= move.

2.1.4. Extra character

This error consists of typing one or more unintended characters. A popular instance of this is *the double character error* caused by hitting two keys of the terminal simultaneously which is mis-read as a duplicated electrical signal, causing a duplicated character to appear on the screen. For example;

> subbj::= subj;
> boguus:: • bogus;
> souurce:: = source;

Another common instance of this error is simply *typing an extra character* which is not intended. For example;

> facests::* facets;

---

[4]These examples do not come from the protocols, but was fabrk^ted to Wustrate the point

efforst :: = effort.

2.2. Grammatical Errors of Natural Language

In all of the above examples simple typographical errors are produced and they are quickly recognized by the users and corrected on the spot. Other errors that are of some syntactic significance involve the re-assessment of larger portions of the text to detect and recover from. Usually knowledge of syntactic rules of natural language are necessary to accomplish this. For example in the following cases the subjects alter the grammatical and semantic contents of their sentences:

thus :: = the;
Here is the :: = Here are the messages.

2.3. Grammatical Errors of System Language

These are errors caused by the misuse of the Rdmail system commands. There were four kinds of such errors encountered in the protocols: *sequence of command, improperly specified argument, improperly specified abbreviations,* and *meta-variable errors.*

2.3.1. Sequence of command

Often a command is used in a context where other commands are necessary to perform operations on the data to enable the initial command to work. That is certain logical dependencies exist between commands due to the special circumstances of the data the system is operating on. When such pre-requisite commands are left-out an error occurs. For example, one subject realizes that he has to *create* before *moving* a file, after first attempting to move the file.

2.3.2. Improperly specified argument or key-word

Often a keyword or argument, or both are improperly specified in a command. This is a common error. For example, the *last* command is erroneously specified at first, and then corrected to the *context* command:

forw la :: = forw cont.

2.3.3. Abbreviations of keywords

Improper abbreviations of system-reserved words is also a common error, especially with inexperienced users. For example, below the "Wholeheaders" command is erroneously abbreviated as "wh".

<-wh :: = <-w

2.3.4. Failure to substitute for meta-variable

Sometimes the user takes a meta-variable used in the help message illustrating the use of a regular variable as the variable itself and uses it in constructing his commands. This will cause the erroneous use of the meta-variable as an argument For example;

<- h from <name>::» <- h from "Jones"[5]

3. Frequency of Errors

The above error categories were sufficient to codify all errors made by the six subjects. Appendix A contains a complete list of all codified errors. The examples in each error category imply that certain error recovery strategies would be effective in reducing the time of recovery from errors and/or occurrence of errors. For example, most typographical errors can be identified through a spelling corrector by comparing with a pre-compiled dictionary of words. Subsequently, a "pattern matcher*[9] of sorts can classify each error into a sub-category. If the mistyped words map into a known word by means of transformation of the hand position on the key-board, then, this error category would be "shifting of key-position." If words with mixed cases are encountered, i.e., "AFter", then the system can identify the error as "unreieased special character key." Once error sub-categories are identified, then the system can assist in automatic recovery by substituting the correct spelling of the word, after consulting the user.

Needless to say such automated recovery features are costly to implement For one thing there is the initial overhead of system development But more importantly, the overhead in terms of on line response times may be the most critical deterrent to the implementation of such automated error recovery systems. In fact significant delays will result from this and the purpose of automatic recovery may be defeated, that is less efficient user interface may result

Consequently, two parallel questions must be asked: Can automated recovery or better yet error avoidance be realized with little or no overhead? And what frequently encountered error categories should such efforts be focused on?

The answer to the former question is likely to be affirmative in those cases where autopometric adjustments to key-boards, can remedy the problem, i.e., for "multiple-key character" and "key-position" error. This does not apply to other types of errors as easily and the control of these errors is beyond the scope of this discussion.

---

^Hus example does not come from protocols, but was fabricated to llustrate the point

However, we can answer the later question here. The implication of the later question is that recovery from error should focus on the most frequently encountered error categorie(s). We can guide the system designer's efforts by examining the frequencies of occurrence and identifying most common errors. In this way error recovery strategies can take into account the frequency with which these errors are expected to occur.

Table 3-1: Error Categories

Subject type	Typographic	Natural language	System syntax	Total
Expert	64	4	6	74
Routine User	62	5	16	83
Total	126	9	22	157

The first question we asked was, if the frequency of errors varied as a function of subject categories and/or error categories as defined above (table 3-1). Analysis of variance indicated that while the subject types did not account for variance in these frequencies ($F(2,1) = 0.69$), error categories did ($F(2,1) = 105.44$). In other words, the frequencies observed for different error categories varied significantly. The frequencies of occurrence for sub-categories of errors also fluctuated considerably:

Transposition	14
Multiple key character	1
Simple substitution	35
Extraneous special-character key	2
Shifting key position	0
Omission	23
Omission due to key-position	11
Duplication	33
Extra character	6
Natural language grammar	9
Command sequence	2
Improper key-word	19
Improper abbreviation of key-word	1

The categories of some significance (those with frequencies higher than 10) are: *transposition, simple substitution, omission, duplication* and *improper keyword*. The only statistically significant difference between the number of errors made by subjects is in the "system syntax" category. Cumulatively the routine users made much more syntactic errors (16) than the experts (6) (t = 3,057;

P < .025). These error categories, then are candidates for further study to accomplish user sensitive error recovery strategies. That is users with different skill levels should be dealt with differently in this case.

While these findings provide a sound point of departure for hardware and software designers interested in error recovery or avoidance at the terminal (especially those interested in mail systems) they are based on a small sample and are intended to be exploratory rather than conclusive.

Appendix: A

SUBJ: 1

Location on Transcription	Error	Error category
1.35	h from rf :: = h from :Rao:	3-inserting extra char.
1.39	SMIT ::= smith	4-not releasing sp.key
2.30	foh :: = ofy	1-transposition
&55	•e :: = w:foo.txt	11 -command sequence error
4.56	requeest :: = request	8-double char.
14.10	*d600 :: = *d6000	6-complete omission
20.30	<contwiith .:scontwith	8-double char.
22.00	<• forw la :: = forw cont	12-improper command
	<- forw la(st) :: = forw cont/to:rao/ subj:spelling corrector information/cc	

Subjects says that he is not perticulariy good at handling this keyboard.

SUBJ: 2

Location on Transcription	Error	Error category
4.40	delete 71+73 ::»:73	3-inserting unintended char
11.04	ty177 ::*ty17	8-double char.
14.12	reqest :: = request	6-complete omission
15.52	Grad :: = Graceful	3-inserting unintended char.
19.44	*d100+5 :: = d100:500	3- inserting unintended character
23.26	•2500+2800 :: = 2500:2800	3- inserting unintended character
23.54	Vrm^n :: = Vrm85	2-multiple key char.
	•jf/.p	12-Improper command
24.08	•/seenurn :: = */jf.p	12-improper command
25.40	*a2000 :: = 22 ::«2300	3- Inseting unintended char.
26.28	*d2400+2500 :: = 2400:2500	3-inserting unintended character
26.58	Vseenum+n :: = seenum:no	3- inserting unintended character
27.35	•r ::=Vrm+ ::=Vrm:79	3- inserting unintended character
27.40	V:: «*Jf/	6- complete omission
28.55	•seenum :: = Vseenum +	6- complete omission
28.55	*seenum+ :: =/seenum:yes	6and3-complete omissionand" inserting unintended char.
34.30	<-help topic ::=<-?	12-improper command
38.33	Spp :: = spelling	8-Double char
39.00	Here is the ::= Here are the messages	10-grammatical natuJanguage error
40.00	help headersearchh ::= help	8 -double char.
44.40	headersh :: = headersearch from "cohen"	3-Inserting unintended char.

44.40	<- h headersearch from "cohen"	12-improper command Illegal message sequence or "headersearch"- missing quoted string. Headersearch from "cohen"
	<- hheadersearch "cohe"	8-doublechar.
	^hto-cohen-	12-improper command
	<- h field	12-improper command Illegal message sequence at "FIELD" - unexpected and of line FIELD
	^hto-cohen"-	12-improper command illegal message sequence at --" "unknown symbol TO "COHEN"
50.06	lmrp :: = Improvements	1-transposition
51.12	abour :: = about	3-inserting unintended char.
52.48	help :: = Ithink	10-grammatical natu.language
53.32	b: :: = written	7?

SUBJ: 4

Location on Transcription	Error	Error category
....	<- htu :: s run from carbonell	12-improper command
2.32	belon :: = below	3-inserting unintended char.
ais	cold :: = could	6-complete omission
4.10	pru ʌsprint	3-inserting unintended char.
ais	te :: = the	6-complete omission
8.55	syystems ::* systems	8-doublechar
9.10	fi ::»?	????
9.45	Thankyou :: = Thankyou!	10-grammatical natu.language
17.00	synopsus ::=* synopsis	3-inserting unintended char.
17.25	Ga ::« Grateful	6-complete omission
17.40	notnion :: = notion	9-extra char, not intended
18.40	tryying :: = trying	8-doublechar.
18.50	thus :: = the	10-grammatical natu.language
19.00	facests :: = facets	9-extra char, not intended
19.20	uuse :: = use	8-doublechar.
20.00	se :: = specification	6-complete omission
20.50	specifible :: = specifiable	7-key position
21.30	had :: = has	Other. 10OR 3-grammatical natu.language OR inserting unintended char.
21.35	bbeen ::=*been	8-double char.
21.40	approah ::» approach	7-key position
22.05	efforst ::» effort	9-extra char. not intended
22.55	oor ::»for	Other. 3- OR 8 inserting unintended char. OR double char
24.35	usr :: = user	7-key position
26.15	sppelling :: = spelling	8-doublechar.
	<-put field totu ::s<-h	12-improper command

	with cohen	
	<- put foo/donc	11 -command sequence
		"done lookup error. No existing file.
		Put aborted"
28.40	cohel :: = cohen	3-inserting unintended char.
31.40	<-s :: = <-deall	12-improper command
33.00	<ddea :: = <-da :: = deall	8and6-double char.andcompplete omission
		double char.
35.15	oof :: = of	Other: 6 OR 7-complete omission
35.25	ad :: = and	OR key position
35.40	of coure :: = ofcourse	10-grammatical natu.language
36.00	usr*s ::» user's	7-key position
36.45	thig :: = thing	7- key position
36.50	hed:: = headers	7- key position
37.50	tes ::»text	3-insering unintended char.
38.30	syys :: = system	8-double char
39.00	sourlce :: = source	9-extra char, not intended
39.15	ppic :: = picture	8-double char.
39.30	ec :: = excetp :: = except	6and1 -complete omission and transposition
39.40	mak :: = manage	*177!*
	ht :: = think	1-transposition
40.00	workking :: = working	8-double char
40.05	suppii :: =-support	3and8-«nserting unintended char and
		double char.(unintch = next key)f22.03
40.20	user:: * used	3-inserting unintended char.
41.00	sum :: = substantive	3-inserting unintended char.
42.10	moe :: = more	7-key position
42.10	keystroked :: = keystrokes	3-inserting unintended char.
42.50	bl :: = plesant	3- - ᴹ
43.00	add :: = and	3and8-inseerting unintended char
		and double char
43.10	suus :: = system	3and8-inserting unintended char and
		double chaar.
43.40	reaally <--really	8-doubkle char.
43.45	ht :: = them	1-transposition
43.55	inwoca :: = invocation	8-double char.
44.00	exaamination :: = examination	8-double char.
44.10	craig :: = Craig	10-grammatical natu.language

SUBJ:5

Location on Transcription	Error	Error category
8.00	subbj ::=subj:	8-double char.
	"graceful interaction"	
12.25	mre :: = more	7-key position
12.50	seplling :: = spelling	1-transposition
14.00	su ::=* using	1-transposition

18.58	Strua :: = Straub	1-transposition
	<<-to "struab"	12-improper command
		No such command
20.09	souuce :: = source	8-double char
20.15	editted :: = edited	8-double char
20.30	THis :: = This	4-not releasing sp.char.key
20.50	dispIlay :: = display	8-double char.

SUBJ: 3

Location on Transcription	Error	Error category
1.30	form :: = fron :: = from	1and3-transposition and inserting unintended char.
	<- send	12-improper command. No such command
3.00	mss :: = message	6-complete omission
7.35	subju :: = subject	3-inserting unintended char.
23.30	Hays :: = Hayes	6-complete omission
23.39	suggestiions :: = suggestions	8-double char.
24.00	oo :: = of	Other: 3 OR 8-inserting unintended char. OR double char.
24.53	*s :: = *d2300	12-improper command
25.00	*d3300 + :: = *d3300:3600	3-inserting unintended char.
31.00	tri <--trying	3-inserting unintended char.
32.00	oo :: = of	Other: 3 OR 8-inserting unintended char. OR double char
33.30	CMUD :: = CMUA	3-inserting unintended char.
34.00	<-h112 :: = <-h12	8-double char.
36.00	Messagge :: = Messags	8- double char.
37.00	essagees :: = Messages	8- double char.
38.00	during :: = during	10-grammatical natu.language
39.40	ch :: = cohen	6-compplete omission
	<- hall	Other: 12 OR 6-improper command OR complete omission
41.52	dcon :: = dcohen	6-complete omission
42.00	"ch :: = "cohen"	6- complete omission
	<- mov from "cohe", to "cohen"/dcohen	12-improper command
		Illegal message sequence at " " – unknown symbol FROM "COHE", TO "COHE"
	<- mov to "cohen"/dcohen	12-improper command
		Illegal message sequence at " " - unknown symbol. TO "COHEN"
44.45	<-ehp :: = <-help	1-transposition
	<- hetu :: = helptu	1and6-transposition and complete omission

	<- helptu :: = h to "cohen"	12-improper command Illegal message sequence at " " – unknown symbol TO "COHEN"
50.10	"ch :: ="cohen" <-hto "cohen"	6-completre omission 12-improper command
56.40	/de :: = dcohen	9-extra char, not intended
58.30	"ch :: ="cohen"	6-complete omission
58.40	imprive :: = improvements	3-inserting unintended chaar.

(SUBJ: 3 continued on next tape)

1.2	sentt :: = sent	8-double char
5.10	main ::: = mail	3-inserting unintended char.
8.46	taks :: stakes	6-complete omission
9.49	redeiiverd :: • redeiivered	6-complete omission

SUBJ: 6

Location on Transcription	Error	Error category
12.00	R.- Rao ::» R.Rao <• headers from carbonell	10-grammatical natu.language 12-command error.Illegal message sequence at "FROM"- missing quoted string FROM carbonell
15.00	Macdonals ::» Macdonaids	6-complete omission
18.00	*ii::»*i	8-doubte char.
22.03	carbonnell :: • carbonell	8-double char.
22.25	ch :: = cohen	7-key position
23.50	Rah :: = Radha E :: = Rdmail	. • *777.*
24.54	a ::»lam	7-key position
25.20	: .. ⌐:	Other: 10 OR 6-garmmatical natu.language OR complete omission
25.40	doesnt :: = donesn:t donesnrt :: = doesn't	3-inserting uninteended char. 3and9-inserting unintended char and extra char, not intended
26.00	fles) :: = files ⌐ <-mail 48,51,62	9-extra char.[)]not intended 12-impproper command "48" not found in addresse file "51" not found in addresse file "62" not found in addresse file
26.20	Alo :: = Also	6-complete omission
26.40	filed ::~fieM	1-transposition
26.55	wuu :: = would	8-double char
27.50	meaningu :: = meaningful	6-complete omission
28.15	Alos :: = Ateo	1-transposition
29.00	-FRAZZEL- ::« "FRAZZLE[11]	1-transposition