MEMORANDUM
RM-4346-PR
APRIL 1965

FACTORS AFFECTING CODING ERRORS

Sidney Owsowitz and Anders Sweetland

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

MEMORANDUM
RM-4346-PR
APRIL 1965

FACTORS AFFECTING CODING ERRORS

Sidney Owsowitz and Anders Sweetland

This research is sponsored by the United States Air Force under Project RAND—Contract No. AF 49(638)-700 monitored by the Directorate of Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this Memorandum should not be interpreted as representing the official opinion or policy of the United States Air Force.

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA • 90406

PREFACE

The development of punched card accounting machinery and of electronic computers has made it disturbingly clear how fallible human beings are in recording data, particularly coded data. Coding errors are often so numerous that they virtually invalidate any analysis of the printed data.

As a step toward solving this problem of coding errors, a number of experiments were conducted in which human subjects coded a variety of data in a number of ways, the object being to determine which methods resulted in the fewest errors.

This Memorandum is one of several stemming from Laboratory Problem IV (LP-IV), a RAND project concerned with Air Force maintenance management. The findings should prove of interest to persons responsible for developing information processing systems in which people record information by coding.

SUMMARY

With the development of electronic computers, both business and scientific data can be processed with enormously greater speed and accuracy than were possible with the best of manual methods.

Given accurate data in an acceptable form (most typically, punched-card input) computerized processing can be dramatic, as when it correctly predicted the outcome of a California election twenty-two minutes after the polls closed. The crux of the matter lies in the proviso, "Given accurate data . . ." More often than not, data inaccuracy is the biggest problem in an information processing system, and is the chief problem in the recording stage, particularly when coding is involved. (Coding is defined as the translation of information into a form suitable for machine processing. For example, "CV" is an Air Force code used for KC-135A tankers and "242" is a code for "failed to operate.")

This Memorandum describes several experiments which sought to identify the factors that contribute to coding errors. The experiments used several kinds of code-stimulus materials: numeric codes, consisting only of numbers; alpha codes, consisting only of letters; alpha-numeric codes of mixed letters and numbers; and mnemonic codes (natural abbreviations, such as "OVH for "overheated"). Only three-character codes were used in the series.

Air Force maintenance personnel were used as subjects of the experiments, in which their coding routine resembled their method of recording real-world maintenance data. Their coded information was keypunched, and the resulting decks were analyzed to determine what factors led to the highest and lowest error rates. The major findings were:

1. Coding errors are proportional to the alpha content. Numeric codes have the smallest error rates. As the alpha content increases, so does the error rate.

2. Errors committed with mixed codes reveal a position effect. Codes with the alpha-numeric-alpha sequence, and the converse, numeric-alpha-numeric, have higher error rates than do codes without this "odd-man-in-the-middle" construction.

3. Perceptual set can increase or reduce error rates. When subjects and keypunchers do not know whether material is letters or numbers, the setting (environment) can predispose the response.

4. There was evidence of interaction (in the statistical sense) among the first, second, and third variables.

5. Unless perceptual set is specifically controlled most errors in recording mixed codes consist of "substitutions of opposites" -- numbers for letters, and letters for numbers.

6. Over half the erroneous substitutions appeared to be non-random, occurring with a limited number of characters.

7. Nonrandom substitutions tended to be unidirectional (e.g., "V" was substituted for "U" much more often than the reverse).

8. Most coding errors (75 to 95 per cent) result from having a single digit in error. The remainder result from reversals or copying the wrong code.*

9. The use of mnemonic codes did not reduce coding errors, apparently because the task involved no learning.

10. Letter-pattern familiarity affects coding errors: codes containing letter pairs in familiar sequence have lower error rates.

11. Coding errors can be substantially reduced by providing keypunchers with a list of codes. This effect appears to be greatest when mnemonic codes are used.

12. The amount of usable information that is retrievable from coded information depends on three factors: (a) the error rate; (b) the number of codes used; and (c) the number of codes possible with the format in question.

Since this study is exploratory, these findings must be considered tentative. They are sufficient, however, to serve as testable hypotheses for further experimentation.

---

*Items 5-8 have significance for efforts concerned with correcting the erroneous code.

ACKNOWLEDGMENTS

A large number of people were very helpful to us in conducting this study. We are especially indebted to CMS R. P. Edwards, Chanute AFB; SMS J. Yeargan, March AFB; Capt. C. H. Mercer, Oxnard AFB; and Col. C. J. Pottebaum, The RAND Corporation.

We wish also to thank both the System Development Corporation at Santa Monica and the Statistical Services Center at Travis AFB for the use of their keypunch facilities.

CONTENTS

I. INTRODUCTION

Information processing generally begins with making observations and recording them. Under modern information processing procedure, they are then keypunched. From this point, the major part of processing is done by machinery, which is almost error-free. The errors occur in the inputs -- the recording and keypunching.

The problem of error can be serious. The Air Force Logistics Command, for example, processes some 2,000,000 maintenance cards a month (not including supply data), with a typical error rate of 5 per cent; that is, about 100,000 cards contain one or more identifiable errors. Although this seems high, one often hears the remark, "If you think that's bad, you should see what it used to be." And in truth, herculean is the right word for the task of holding errors to the 5-per-cent level.

To date, the major effort in solving the error problem has gone toward detecting errors in documents themselves. To supplement the traditional pencil-and-green-eyeshade approach, a number of computer programs and card-sorting techniques have been developed. Both methods work well, the least adequate of them being able to deliver an error rate of less than 1 per cent.

A second approach is to control error instead of eliminating it. The statistical methods used to randomize and balance error are a simple illustration of control, as in the computation of fiducial limits. Generally speaking, statistical approaches are well developed in theory, but so far have had little practical application in data processing. Another way of controlling error is to reconstruct the erroneous information to yield a true record. Very little effort has been expended upon this approach.

A third solution has been almost entirely untouched: error prevention. This might be called "designing" human-factor elements into data processing systems, the object being to make the coding situation as error-free as possible. This Memorandum explores this third avenue, seeking to establish some guideposts that can be used in future developments. The attempts can best be understood by the following paradigm.

Given that a component (system, black-box, bit or piece, etc.) is in a status that can be described and coded, and given a sufficient and adequate code, the gamut of the coding process can be subdivided into a number of loci or steps:

1. The human observer examines the component and judges what its status is.

2. Referring to his manual, he finds a word or phrase that describes his judgment.

3. After finding the appropriate description (or closest substitute) he enters the corresponding code on the form.

4. The form is later reviewed by one or more people who may make corrections.

5. The form is keypunched and verified.

If the description keypunched on the card accurately describes the status of the component, we say the description is valid. If the system consistently records the true statuses of a large number of components, we say the system is a valid recording mechanism.

It is apparent that the validity of the system is vulnerable at a number of loci. This study addresses itself to only a portion of the possibilities: the coding-keypunching sequence (Steps 2-5). The study is restricted to the question, "What kinds of coding tend to reduce the validity of the system?" Specifically, four methods of encoding information are compared:

1. The use of numeric codes: 475, 127, 392.

2. The use of alpha codes: DEG, GAQ, LXE.

3. The use of alpha-numeric codes: 12G, A2E, 5U4.

4. The use of mnemonic codes: BNT (for Bent), VIB (Vibrated), OVH (Overheated).

The study was restricted to three-digit codes for several reasons: (1) psychologists have amassed a large amount of relevant research on the factors involved in learning nonsense syllables (e.g., three-digit codes with wide ranges of association values); (2) the "how-mal" code is a three-digit code, and has been very troublesome to maintenance personnel using the AFM 66-1 system; (3) since a three-digit code is a compromise between small (one or two digits) and large (six to ten digits), we hoped to cover the most ground by using a coding format with some characteristics of both extremes; (4) transcribing three-digit codes (in contrast to six-to-ten-digit codes) involves little memorization.

This study is divided into four sections. First, we compare the error rates among alpha (letter), numeric (numbers), and mixed alpha-numeric codes. Second, we look at individual letters and numbers within these codes to determine the details of the nature of errors. Third, we determine the error rate of mnemonic codes. Fourth, we investigate the relationships among error rates, information content of codes, and data retrievability.

II. ERROR FACTORS FOR ALPHA AND NUMERIC CODES

The subjects used in the first experiment were forty maintenance personnel from Oxnard AFB. Their age-experience range was typical for maintenance personnel, extending from teenagers to men in their forties with two to twenty years of military experience. The subjects were divided into two groups of twenty men.

Both groups were given 20-page recording booklets (see Fig. 1), which were similar except that those for Group I had a numeric index and those for Group II an alpha index. The task was the same for both groups: to look up three-digit codes in the "code book" and enter them in the appropriate spaces in the recording booklet.

The object in using the two separate look-up lists (alpha and numeric) was to determine whether a "look up" factor existed, i.e., whether the method of finding the information had an effect on coding errors. In other words, we were attempting to determine the effect of a perceptual set. Perceptual set is defined as a tendency to respond in a predisposed manner, as a hunter and a photographer are set to respond uniquely to the same stimulus.

Digits were selected on a random-without-replacement basis. The three-digit codes covered the eight possible alpha-numeric* combinations NNN (all numeric); NNA (two numerics and one alpha); NAA; NAN; ANA; ANN; AAN; and AAA (all alpha). Neither the numeric zero nor the letter "O" was used in the codes to be recorded. Two hundred items (25 of each combination) were coded by each subject.

Before keypunching, each group of twenty 20-page recording booklets was unbound and the loose pages were rearranged so that there would be, in Group I, for example, twenty reassembled booklets, each containing a page from each of the 20 subjects. The same arrangement was carried out in the booklets marked by subjects from Group II. Then the 40 booklets were keypunched and verified by eight teams from Travis AFB, and the resulting decks were scored by computer.

---

*The term is used in its ordinary generic sense here and in the pages to follow to embrace numeric, alpha, and alpha-numeric codes.

Numeric Look-up	Alpha Look-up
004.....	D.....
037.....	AK.....
045.....	AS.....
067.....	BØ.....
193.....	GK.....
135.....	EE.....
019.....	S.....
055.....	BC.....
151.....	EU.....
103.....	CY.....

001	D93	A	D93
002	EV4	B	EV4
003	QPM	C	QPM
004	9WV	D	9WV
005	T2S	E	T2S
006	Y6Q	F	Y6Q
007	29Y	G	29Y
008	R54	H	R54
009	75H	I	75H
010	341	J	341
...	...	..	...
198	9V8	GP	9V8
199	ZF1	GQ	Z51
200	PA7	GR	PA7

NOTE: The upper section shows the fourth page of a recording booklet. The lower portion lists the first ten and the last three entries in the code book. In each look-up, the first entry should be "9WV."

Fig. 1 -- Format of Recording Booklets and Codebooks

To comprehend the following discussion, it is necessary to remember that the two groups of subjects differed only in whether the look-up procedure was alpha or numeric, and that there was almost complete confounding of such irrelevant factors as interactions among codes, subjects, and keypunching.

The first question was whether any of the independent variables

Table 1

TESTS OF HOMOGENEITY FOR 500 TRIALS PER CODE-SEQUENCE

Weighted grand mean = 0.095
Unweighted grand mean = 0.095
Chi-square = 116.34
df = 15
Chi-square (arc sin) = 159.57
The probability is beyond the 0.001 level

Alpha Look-up			Numeric Look-up		
Code Sequence	No. of Errors	Mean Errors	Code Sequence	No. of Errors	Mean Errors
NNN	18	0.036***	NNN	6	0.012***
NNA	34	0.068	NNA	38	0.076
ANN	43	0.086	ANN	38	0.076
NAN	53	0.106	NAN	44	0.088
AAN	54	0.108	AAN	51	0.102
NAA	61	0.122	NAA	54	0.108
AAA	61	0.122	AAA	59	0.118
ANA	61	0.122	ANA	82	0.164***

NOTE: Three asterisks denote 0.001 confidence level.

had a significant effect. That there were effects can be seen from Table 1, which, like Tables 3 and 4, contains the results of two tests of homogeneity embracing all code-sequences. The first test is the conventional two-category chi-square test for dichotomous data. The second (arc sin) is that recommended by Eisenhart, et al.[*]

[*] C. Eisenhart, M. W. Hastey, and W. A. Wallis, Selected Techniques of Statistical Analysis, McGraw-Hill Book Co., New York, 1947, pp. 406-411.

In Tables 1, 3, and 4 the fiducial limits of each observed frequency are determined at the 0.05, 0.01, and 0.001 levels, and the theoretical frequency is tested to determine whether it falls inside these limits. Statistically significant cells are tagged with one, two, or three asterisks representing the 0.05, 0.01, and 0.001 levels of confidence, respectively. (In the actual tests, none of the items fell within the 0.01 confidence level.) The conventional binomial tests are used, in which m = np with variance npq.

The data in Table 1 suggest that: (1) the numeric codes have the lowest error rate; (2) error rates increase as the alpha content of the codes increases; (3) there appears to be a look-up factor; (4) there is a position effect with mixed codes: those with the odd code in mid-sequence (i.e., ANA, NAN) tend to have the higher error rates; (5) the range of errors is great: 1.2 to 16.4 per cent. Each of these possibilities will be explored in the following discussion. It is worth noting that both groups of subjects (alpha and numeric look-up) took about the same amount of time to complete the task: an average of 25 minutes.

THE LOOK-UP EFFECT

In evaluating the results, the first task was to determine whether there was a "look-up" factor (perceptual set). The answer can be seen in Table 2, where each cell tabulates the relative frequency of errors based on 500 responses. The significance level is based on the conventional test of differences between proportions.

The test on the totals (in Table 2) shows no significant difference. However, there are two significant differences between the pairs of samples: in numeric look-up there is a marked decrease in the number of errors for the all-numeric code (NNN) and also a marked increase in errors for the mixed code ANA.

It is reasonable to conclude that perceptual set can contribute to coding errors. Even the very mild condition of the manner used here is sufficient to produce significant differences. Because the effect did not occur under all circumstances, and because one cannot completely eliminate the possibility that the differences shown are false positives (i.e., type II errors), these conclusions cannot be said to be firm. A cross-validation study is highly desirable.

The presence of the "look-up" factor in some of the pairs will be the source of some awkward comparisons in the subsequent discussion. For the sake of simplicity, the alpha and numeric look-up data will be combined despite the fact that two of the eight pairs (NNN and ANA) were significantly different. While this combination is defensible on rational grounds, it is not so readily defensible statistically. To circumvent this problem, each case was tested separately and any stated

Table 2

RELATIVE FREQUENCY OF CODING ERRORS IN
16 DIFFERING CIRCUMSTANCES

Code Type	Alpha Look-up	Numeric Look-up	Significance Level
NNN	.036	.012	.01
NNA	.068	.076	NS[a]
ANN	.086	.076	NS
NAN	.106	.088	NS
AAN	.108	.102	NS
NAA	.122	.108	NS
AAA	.122	.118	NS
ANA	.122	.164	.05
Total	.096	.093	NS

[a]Not significant.

conclusions are based on this more detailed study.

ALPHA VERSUS NUMERIC CODING

Inspection of Table 1 strongly suggests that the error rate increases as alpha content increases. The two exceptions to this observation are the reversal of codes AAA and ANA in the numeric look-up phase, and the identical results for codes NAA, AAA, and ANA in the alpha look-up. A number of comparisons were made by combining the data of Table 1, all leading to the same general conclusion: error rates increase with alpha content. One such comparison is detailed in the following paragraphs.

Inspection of Table 1 indicates that the error rate tends to cluster in a manner consistent with the logical division of (1) no alpha, having 1 to 4 per cent errors; (2) one alpha, 7 to 11 per cent errors; and (3) more than one alpha, with 10 to 16 per cent errors. The exercise summed up in Table 3 was undertaken to see if there was statistical support to the division based on logical considerations.

Table 3

TESTS OF HOMOGENEITY FOR SELECTED CODE-SEQUENCE AGGREGATIONS

Weighted grand mean = 0.095
Unweighted grand mean = 0.076
Chi-square = 94.55
df = 2
Chi-square (arc sin) = 130.86
The probability is beyond the 0.001 level

Code Sequence	No. of Trials	No. of Errors	Mean Errors
NNN	1000	24	0.024***
NNA	3000	250	0.083*
{AAN, AAA}	4000	483	0.121***

NOTE: * indicates 0.05 confidence level; *** indicates 0.001 confidence level.

First, the means of the single-alpha results were tested for homogeneity. The same test was made for the means of the multi-alpha results. In both, the distributions were homogeneous, although that for the ANA code was of marginal homogeneity. Following this (see Table 3), the means of the three groups (no-alpha, single-alpha, and multi-alpha) were compared. The test shows the presence of three statistically different groups.

It must be remembered, when one considers Table 3, that the NNN code is the result of combining two look-up groups that were statistically different (Table 2). Further, the multi-alpha combination is also marginal. Hence, the data of Table 3 must be interpreted in strict accord with the thesis in question: that error rate is proportional to the amount of alpha content. It is interesting to note that there are five statistically independent clusters in the data:

(1) Numeric, numeric look-up.
(2) Numeric, alpha look-up.
(3) Single alpha, combined look-up.
(4) Multi-alpha, combined (excepting item 5).
(5) Code ANA, numeric look-up.

This list suggests that three factors determine error rate:

(1) Proportion of alpha content.
(2) Perceptual set (i.e., look-up factor).
(3) The interaction of alpha proportions and perceptual set.

THE POSITION EFFECT

The behavior of the ANA error rate was so dramatically different from the behavior of the other codes that it provoked further search for clues. One possibility was a position effect. Given two-alpha and one-numeric characters, three sequences are possible: those in which the numeric element is at the beginning, at the end, or in the middle. (The single-alpha, two-numeric case is analogous.)

A perusal of Table 1 strongly suggests that the singleton-in-the-middle arrangement is unusual: in all instances the error equals or exceeds the error under the singleton-at-one-end arrangement.

Tests were made for the single-alpha-in-the-middle data against the other single-alpha data; a similar comparison was made with the single-numeric data. A third comparison, shown in Table 4, was made for the combined singleton-at-one-end data. In all three comparisons, the singleton-in-the-middle codes yielded the largest error rate.

The data in this Section lead to the following conclusions, which may be used as hypotheses for testing in subsequent experiments:

(1) Perceptual set affects coding error rates.

(2) Coding error rates are proportional to alpha content: the smaller the alpha, the lower the error rate. Numeric codes have the lowest rate of all.

Table 4

TESTS OF HOMOGENEITY FOR POSITION EFFECT:
SELECTED CODE-SEQUENCE AGGREGATIONS

Weighted grand mean = 0.102
Unweighted grand mean= 0.107
Chi-square = 10.40
df = 1
Chi-square (arc sin) = 10.05
The probability level is between 0.01 and 0.001.

Code Sequence	No. of Trials	No. of Errors	Mean Errors
XXO	4000	373	0.093
XOX	2000	240	0.120*

NOTE: XXO represents all mixed alpha-numeric codes in which the singleton falls at either the beginning or the end. XOX represents singleton-in-the-middle codes. XOX can therefore represent either ANA or NAN codes.

* indicates 0.05 confidence level.

(3) The relative positions of characters in alpha-numeric codes affect the error rate. The singleton-in-the-middle codes have the highest error rates.

(4) These main variables interact.

III. SINGLE-CHARACTER ANALYSIS

The findings of Sec. II suggested the desirability of further research into the causes of coding errors. This, in turn, points up the need for deeper understanding of the nature of the coding errors themselves. Further, since we were interested in the possibility of getting a computer to correct errors in coding, two response characteristics were of interest: nonrandom errors, and "directional errors." (Directional errors will be defined in the discussion.) As a result, the following exploration was made.

In the exercises described in Sec. II, an error was scored when any keypunched three-character code did not agree with the master deck. This disagreement resulted from any of several kinds of errors: transposition, copying the wrong line, substitution of other numbers or letters, omissions, and so on.

The second phase of this study involved looking at each character (letter or numeral) and determining the nature of the error. With one or two exceptions, the error consisted of substituting:

(1) A letter for a letter;

(2) A letter for a number;

(3) A number for a number;

(4) A number for a letter.

In the following discussion, the material to be coded by the subjects is referred to as the "stimulus" and the coded entry as the "response." The question addressed is: what effect does the kind of stimulus material have on response errors?

The results of the first single-character exploration are listed in Table 5. The table reveals a new finding: during alpha look-up, when the stimulus material is letters, the error rate for letters is lower and for numbers higher than expected (e.g., .057 instead of the expected .058, and .023 instead of .020). When the look-up is numeric, the reverse is true. Accordingly, the single-character errors were analyzed further to see what kind of errors they were.

An error for a letter stimulus can result from the substitution of another letter or the substitution of a number. Similarly, numbers can be erroneously represented by letters or by other numbers. The results of the single-character analysis are shown in Table 6.

To facilitate comparison, Table 6 includes the theoretical frequencies (calculated from the marginal totals). That the computed Chi-square is highly significant is indicated by the fact that the contribution of any one of the cells is sufficient to reach the 0.10 level.

The major fact revealed in Table 6 is that, regardless of the look-up procedure, the largest number of errors occurs when letters are the stimulus material. The predominant error in this case is the substitution of a number for a letter.

At the other end of the error-rate continuum is the substitution of erroneous numbers for numeric stimuli. This yields the smallest error rate. It is interesting that the ratio of error rate of these extremes is ten to one.

Further inspection of Table 6 indicates that the major source of error is the "substitution of opposites" (numbers for letters and letters for numbers). The ratio of "substitutes-of-opposites" to "substitutes-of-sames" is 634 to 294 (2.15 to 1.0). This difference is highly significant. The computed Chi-square is 124 -- far beyond the tabled value of Chi-square.

The finding in regard to substitution of opposites may have considerable practical value: it may be possible to get the perceptual-set factor to work <u>for</u> system accuracy, rather than against it as it does here. Our experimental setup was such that neither the coders nor the keypunchers were able to develop a perceptual set about the alpha-numeric format of the data because of the random selection procedures used in determining the stimuli sequences. It seems reasonable to believe that substitutions of opposites would fall off sharply if both coders and keypunchers were made aware of the format of the data -- by instructing them, for example, that "the first two characters in this field are always numeric and the last character is always alpha. There is no exception." An analogous case is reported in Sec. IV: where the experimental setup enabled the keypunchers to determine if the code were alpha or numeric, the substitutions of opposites dropped to zero.

The single-character data were examined next to determine what

Table 5

ERROR RATES FOR SINGLE CHARACTERS

Look-up	Type of Stimulus	Total Responses	Error Count	Proportion
Alpha	Letters	6,000	341	.057
	Numbers	6,000	135	.023
	(Total)	(12,000)	(476)	(.040)
Numeric	Letters	6,000	353	.059
	Numbers	6,000	99	.017
	(Total)	(12,000)	(452)	(.038)
Combined	Letters	12,000	694	.058
	Numbers	12,000	234	.020
	(Grand Total)	(24,000)	(928)	(.039)

Table 6

ERROR FREQUENCIES CHARACTERIZED BY TYPE OF SUBSTITUTION

Look-up	Stimulus	Letters	Numbers
Alpha	Letters	103 (153.6)	238 (187.4)
	Numbers	103 (60.8)	32 (74.2)
Numeric	Letters	136 (159.0)	217 (194.0)
	Numbers	76 (44.6)	23 (54.4)

NOTE: Theoretical frequencies are shown in parentheses.

specific characters were substituted. (Because of the small numbers involved, no statistical tests were made.) One sample of these examinations -- the alpha look-up tabulation -- is shown in Table 7.

Two things are of interest in Table 7. First, two kinds of errors appear: random and constant. By "random" is meant that erroneous substitutions are apparently not caused by selective factors. Thus the letters "C," "F," and "H" show no consistent translation into other letters or numbers. The reverse is true for "I," often coded as "1," and "U," often coded as "V." The latter are "constant" errors.

Second, constant errors often show unidirectionality. Thus, under alpha look-up conditions, "I" is coded as "1" much more frequently than the reverse. Similarly, "U" is coded "V" and "S" is coded "5" much more often than the reverse. Such unidirectionality is important in error-correction programs, in which we attempt to discover what the correct code is, because it provides "most likely choices" (as do constant errors, of course).

The reader may note that the one-character error rates reported in Table 5 are not completely consonant with those reported on the three-character codes. The latter are not exactly three times the single-character error rates. The reason is that more than one character of a three-character code may be in error. This usually occurs when there is a reversal of the characters, and when the wrong line of the code-book is copied.

Computers cannot definitely identify the latter types of errors, because it is impossible for humans to make these qualitative judgments with any consistency. However, it is possible to project the data of Table 5 and estimate what the three-character error rate should be if there were only one error per code. The difference between this projection and the actual results is the result of multiple errors within three-character codes.

To illustrate: the alpha look-up error rate for letters is 0.057, for numbers 0.023 (see Table 5). Consequently, the projected error for an alpha look-up three-character code consisting of two letters and one number is:

$$2(0.057) + 1(0.023) = 0.137$$

Table 7

RESPONSES SCORED BY SINGLE-DIGIT STIMULI, ALPHA LOOK-UP

Stimulus	Responses			Substitution Error
	Total	Correct	Incorrect	
A	240	235	5	C X 4 6 8
B	220	213	7	3 2-5s 4-8s
C	240	233	7	D Z 1 4 6 8 9
D	240	235	5	G L 1 2-9s
E	240	233	7	3-Fs 2-3s 7 (+1 illegal character)
F	240	236	4	Q 3-1s
G	240	224	16	3-Cs Z 11-6s 8
H	240	240		
I	245	108	137	3-Ns T 127-1s 6
J	240	229	11	B 1 9-5s
K	240	232	8	F M Q V 3-Xs 8
L	260	256	4	M Y 1 6
M	240	237	3	N P Y
N	240	237	3	U V 9
Ø		Unused		
P	240	233	7	D E M W 2 2-9s
Q	240	231	9	D H S T U 4 3-9s
R	240	238	2	P 9
S	240	228	12	1 10-5s 9
T	240	236	4	V 2-2s 7
U	240	206	34	H J N R 26-Vs Y 1 3 9
V	240	230	10	J 5-Us 3-Ys 3
W	240	238	2	1 8
X	240	228	12	K 6-Ys 1 3-7s 8
Y	240	226	14	E H L 3-Vs 3-Xs 3 4 2-7s 9
Z	240	217	23	2-Is L 17-2s 2-6s 7
1	700	679	21	2-Cs 6-Is J M N 2 4 3-6s 4-7s 9
2	680	639	41	D S 33-Zs 3-1s 4 2-9s
3	680	673	7	2-Bs V W Z 5 8
4	680	663	17	A B F 3-Hs I N S 2-Us X 2-Ys 1 3 7
5	640	624	16	2-Bs H 3-Js K 5-Ss T 2-1s 8
6	700	689	11	A C F 4-6s K 2-5s 8
7	600	593	7	E L M T V 2-Zs
8	640	633	7	3-Bs K Y 1 6
9	680	672	8	D 2-Fs J Z 1 4 8
0		Unused		

NOTE: Line 2 is read: for the stimulus "B," 213 correct responses were given as well as 7 erroneous responses. The erroneous responses were 3, two 5's, and four 8's.

The results of these computations are shown in Table 8. The conclusion is that about 25 per cent of the errors with three-digit codes are due to multiple errors.

The findings of Sec. III may be summarized thus:

(1) In the main, errors resulted from the substitution of opposites (a number for a letter -- the most common error -- or a letter for a number).

(2) Substitutes were of two kinds: (a) random, in which the substitution was made with equal frequency from the list of erroneous characters; or (b) constant, in which a specific character was substituted repeatedly for another specific character (e.g., "1" for "I").

Table 8

3-DIGIT ERROR RATES PREDICTED FROM SINGLE-DIGIT ERRORS

Look-up	Stimulus	Predicted[a]	Actual
Alpha	3 alpha	.170	.122
	2 alpha	.136	.117
	1 alpha	.102	.087
	0 alpha	.068	.036
Numeric	3 alpha	.176	.118
	2 alpha	.134	.125
	1 alpha	.092	.080
	0 alpha	.049	.012

[a]The differences between the predicted and the actual result from multiple error within the same 3-digit code.

(3) Constant errors tended to be unidirectional; e.g., "V" was frequently substituted for "U," but not vice versa.

(4) About 25 per cent of three-digit-code errors were the result of multiple errors within codes. Inversion and transcriptions of the wrong line appeared to be the most frequent cause of multiple errors.

(It should be noted that the data were retested after removing the "I-1" errors. The conclusions remain the same.)

IV. MNEMONIC ERROR RATES

In the third of the series of experiments we had three main objectives:

(1) To determine the relative effectiveness of mnemonics (compared with alpha and numeric codes).

(2) To determine the effects of letter-pattern familiarity, as represented by digram value.

(3) To determine the effects when the act of coding is free from degrading perceptual set, i.e., when the code is unequivocally known to be either alpha or numeric.

To make these determinations, four additional sets of stimulus materials were constructed:

(1) A set of mnemonics for 50 "how mal" codes.

(2) A set of codes made from the 50 mnemonics but with the letter combinations scrambled.

(3) A set of random three-digit alpha codes.

(4) A set of random three-digit numeric codes.

The set of 50 mnemonic codes were obtained by selecting 150 single-word "how mal" adjectives from the official Department of Defense list. The 150 words were then presented to 77 March AFB airmen and NCO's, who were asked to write down the three letters they felt best stood for each word. (We were attempting to construct a list of codes that would come most readily to the typical user.)

From these sets of responses, 50 items were selected which (a) had high modal response: the average mnemonic was selected by 50 per cent of the airmen and NCO's (the range of the final list was 19.6 to 90.9 per cent); and (b) were not given as mnemonics for other words. (BRN, selected as a mnemonic for both "broken" and "burned," was not used.)

The data summarized in Table 9 were treated, statistically, with a variety of approaches. These all pointed to the same conclusions: the codes using letters yield essentially the same error rates, but codes using numbers yield much lower error rates.

As an interesting aside: the lowest alpha errors were for scrambled mnemonics. There is a strong suspicion that this result was fortuitous --

that it was almost entirely due to the handwriting of the scrambled-mnemonic test group -- since in a follow-up study, two keypunchers spontaneously commented about the "good handwriting."

To those familiar with research dealing with nonsense syllables, the findings in Table 9 will come as a surprise. High-association nonsense syllables (essentially identical to our mnemonics) are commonly associated with low error rates of recall. These are such syllables as LOV, MEX, and VAL, which are identified as words virtually 100 per cent

Table 9

ERROR OF 52 SUBJECTS (100 TRANSCRIPTIONS FOR EACH) IN EACH OF THE FOUR EXPERIMENTAL CONDITIONS

Item	Mnemonic Codes	Scrambled Mnemonics	Random Alpha	Random Numbers
Errors	293	238	300	71
Proportions	.056	.046	.058	.014

of the time by subjects used in nonsense-syllable experiments. Low-association nonsense syllables (the same as our scrambled mnemonics and random alpha) are commonly associated with high error rates of recall. These are such syllables as LAJ, MEQ, and VEC, identified as words less than 1 per cent of the time. Nevertheless, the use of mnemonics in the coding experiments had no value in reducing coding error, a conclusion at variance from that suggested by nonsense-syllable experiments.

This discrepancy is undoubtedly due, in a large part, to the difference in the experimental task. In the coding experiments we were dealing with the phenomenon of immediate recall with little opportunity, if any, for forgetting factors to operate and thus degrade recall. By contrast, the typical nonsense syllable experiment requires the subject to learn a list of nonsense syllables (generally 10 to 20). Since the response to any stimulus is repeated only after a substantial amount of intervening learning, the opportunity for the disruption of learning is high.

THE EFFECTS OF LETTER-PATTERN FAMILIARITY

A second approach to the relationship between error rates and association value was to investigate the effect of digram value on error rate. Digram value is a measure of letter-pattern familiarity. For example, the sequence "th" is common in the English language, while "ht" is rare. Frequency counts of these sequences in English have been tabulated.* These frequency counts are the digram values: high digram values indicate high occurrence in the language.

A digram value for each three-letter code was obtained by summing

Table 10

SPEARMAN CORRELATION BETWEEN LETTER FAMILIARITY AND CODING ERROR

Code	Correlation	t	Significance
Mnemonics	-0.244	1.74	0.05
Scrambled mnemonics	-0.308	2.24	0.025
Random alpha	-0.344	2.53	0.01

the value of the two digrams making it up. In effect, each three-letter code was assigned a letter-pattern familiarity score. Since the error count for each code was also available, it was possible to determine the relationship between error-frequency and letter-pattern familiarity in each condition (see Table 10).

The results in Table 10 indicate there is a small but significant relationship between coding error and letter-pattern familiarity: the greater the letter-pattern familiarity, the less coding error.** This finding, coupled with the previous one, leads to the conclusion that in

*B. J. Underwood and R. W. Schulz, Meaningfulness and Verbal Learning, J. B. Lippincott Co., New York, 1960.

**See also S. E. Owsowitz, The Effects of Word Familiarity and Letter Structure Familiarity on the Perception of Words, The RAND Corporation, P-2820, November 1963.

simple coding tasks (where no learning is involved) association value has no effect on error rate while letter pattern familiarity does.

We also repeated the study of the single-digit errors with the data; the results are shown in Table 11. The following items are noteworthy:

(1) There was only one substitution-of-opposite error since the perceptual setting was unambiguous: it was either alpha or numeric. This fact undoubtedly contributed substantially to the lower error rates in this phase.

(2) Both the random and constant error types occurred as before.

(3) There appeared to be some reduction of the unidirectionality of the constant errors.

(4) By far the most frequent source of error was the "U-V" confusion.

CONTROLLING PERCEPTUAL SET

The experience with mnemonics suggested that meaningfulness of the code to the coder had no effect on coding error. This provoked the question: Would this also be true from the keypunching side of the team? -- i.e., would mnemonics facilitate the winnowing out of error during keypunching? In an attempt to answer this, the following procedure was devised. The coding booklets were separated into four groups: the mnemonic, the scrambled mnemonic, the random alpha, and the random numbers groups. The keypunch operators were provided with the appropriate "code manual" and instructed, "If the handwriting proves difficult, consult the codebook to see if you can figure out what code was intended." The cards, as before, were not verified.

The results of this procedure are summarized in Table 12, which reveals a sizable reduction in error in every instance. The implication is, then, that error rates can be substantially reduced simply by providing keypunchers with the codes in question. In all cases, the error rates drop to acceptable levels, even without verification.

An additional observation is the sharp drop in the error rates of mnemonics. This fact may have practical implications. Mnemonics have the advantage of being readily learned. It is interesting to note that

Table 11

RESPONSE ERRORS FOR THE RANDOM ALPHA AND RANDOM NUMERIC DATA

Stimulus	Responses Total	Correct	Incorrect	Substitution Error
A	624	621	3	D L P
B	624	621	3	2-Ds V
C	520	505	15	4-Gs 3-Ls 2-Ps Q S V 3-Xs
D	520	508	12	F 2-Øs 8-Ps R
E	624	615	9	4-Cs 3-Fs 2-Gs
F	624	616	8	A 2-Ds 2-Es I T W
G	520	503	17	A B 5-C E F 2-Js K 2-Øs Q Z 6
H	624	616	8	A K 5-Ms N
I	624	613	11	2-Es 3-Fs 2-Qs 3-Us V
J	520	502	18	D 2-Is K 2-Ls 5-Ss 3-Ts 3-Us V
K	624	618	6	H I L M R X
L	624	610	14	4-Gs F 2-Is K Ø T 2-Us 2-Zs
M	624	621	3	B 2-Ns
N	520	505	15	2-As 2-Hs M U V 8-Ws
Ø	624	602	22	B 13-Ds 2-Qs 4-Us (+ 2 illegal characters)
P	520	512	8	C 3-Ds E 2-Ls R
Q	624	606	18	Q 4-Gs 8-Øs U 3-Ws Z
R	624	619	5	3-Bs 2-Ts
S	624	619	5	A B 20-Js X
T	624	620	4	2-Is W X
U	624	593	31	2-Cs H J 2-Øs 25-Vs
V	624	587	37	C 3-Js N Ø 21-Us 10-Ys
W	624	615	9	E L 3-Ss T 2-Us V
X	624	600	24	4-Cs H 2-Ks N S T V 6-Ys 7-Zs
Y	624	606	18	I J R T 7-Vs 7-Xs
Z	624	612	12	C I K 3-Ls 2-Ps Q U V X
1	1560	1554	6	2, 2-3s 5 7 8
2	1560	1545	15	3-1s 4-3s 4 2-5s 2-7s 8 2-0s
3	1560	1555	5	2-5s 2-6s 8
4	1560	1548	12	2 2-6s 4-7s 8 2-9s 2-0s
5	1560	1547	13	3-1s 5-3s 2-7s 3-9s
6	1560	1550	10	3-3s 2-4s 3-9s 2-0s
7	1560	1542	18	3-1s 8-2s 3 2-5s 3-9s M
8	1560	1550	10	1 2 3 3-9s 4-0s
9	1560	1552	8	2-2s 2-4s 2-7s 8 0
0	1560	1553	7	3 5 4-6s 9

NOTE: Line 2 is read: for the stimulus "B," 621 correct and 3 incorrect responses were given. The incorrect ones were two D's and one V.

the keypunch operators did not learn to relate the word with the mnemonics; rather, they learned what they called the "pattern" -- a familiar letter sequence, but with no other meaning. With the "pattern" learned, they no longer punched individual letters; they punched the "pattern" and at this stage they also readily recognized "non-pattern" codes. One suspects that at this end of the sequence: (1) learning did take place; (2) learning was facilitated by high association value; and (3) the learned materials facilitated elimination of error.

Table 12

THE EFFECT OF KNOWLEDGE OF THE CODE BY KEYPUNCH OPERATORS

Code	No Codebook Error Rate (A)	Codebook Error Rate (B)	B/A
Mnemonics	.056	.015	(26.8%)
Scrambled Mnemonics	.046	.022	(47.8%)
Random Alpha	.058	.027	(46.6%)
Random Numbers	.014	.009	(64.2%)

One further observation is noteworthy: an analysis of the multiple errors for the random digits and random numbers showed that about 10 per cent of errors were of the multiple-entry class (e.g., transposition, copying the wrong line, etc.). This occurred most frequently with random numbers: 25 of 71 were multiple errors, while random letters had only 11 of 300 multiple errors. Note that this 10-per-cent multiple-error rate is considerably less than the 25 per cent quoted in Sec. III.

The following observations seem germane for the exercises described in Sec. IV:

(1) Mnemonic codes were of no value in reducing coding errors. This result appears to be due to the absence of a learning factor.

(2) Letter-pattern familiarity had a slight but significant effect: the more familiar letter patterns had a lower error rate.

(3) Providing keypunch operators with the code manual substantially reduced errors. This reduction was greatest with mnemonic codes.

(Perhaps the learning factor was then in operation.)

(4) Handwriting appeared to have a small effect; the better handwriting had the fewest errors. Compared with other variables, this effect was slight.

(5) Perceptual set can be used to reduce errors.

(6) Numeric codes, as before, showed the lowest error rates.

V. THE ERROR-DISCERNIBILITY FACTOR

Despite the fact that the error rate for alpha codes is generally several times greater than that for numeric codes, we are reluctant to summarily dismiss alpha codes, since they can transmit a good deal more information per character than numeric codes can. In the first experiment, for example, we used 25 letters having an information rate of 4.64 bits per character (i.e., $\log_2 25 = 4.64$), and nine numbers having 3.17 bits per character. The bit/character ratio was therefore approximately 1.5 to 1.

With alpha codes, in other words, we can increase the information transmission substantially but at the cost of a substantial increase in error. In terms of the codes used in this experiment, the information rate was increased by 50 per cent, but the error rate by 500 per cent. Thus the question becomes: Would you rather have 150 units of information, 10 per cent of which are erroneous, or 100 units of information with 2 per cent error? Fortunately, as will be shown, this question does not have to be asked too often, since the number of card columns (with EAM) are generally not at a premium. For example, a single IBM-type card column is sufficient to allow for 10 discriminations or unique codes using any numeric code (i.e., numeric 0 through 9). If these are insufficient, the 26 alpha codes will allow 26 discriminations and alpha-numeric combinations will allow 36; but in each instance, as we have seen, there is a cost in increased errors (from less than 1 to more than 16 per cent).

Another factor must be included in this discussion before this dilemma can be resolved: what we have termed the "error-discernibility factor." This concept is closely related to the problem of Type I and Type II errors in the statistical domain. As will be shown, it is the combination of the information content and what we call the error-discernibility factor that provides some leverage on the vexing problem of how to get data that are reasonably error-free. This is best explained by illustration.

Taking an example similar to the current AFM 66-1 "how mal" codes, assume that 100 numeric codes are being used and that the true error rate is 2 per cent. If 10,000 items are coded, 200 cards will be in error. However, two kinds of errors are possible: detectable and undetectable. A person may enter a code that is wrong for his purpose but is nevertheless to be found in the list of legal codes, or his entry may be unlike any code in the list. Since the only way to detect an error is to compare the entry with the list of legal codes, the first error is not detectable, but the second is.

Error detectability involves two things: the number of codes being used and the information capacity of the system. In this example, we are using 100 three-digit numeric codes; thus we use only 100 of 1000 possible discriminations. It follows that the odds of making an undetectable error are 100/1000 = 0.1, and the odds of making a detectable error are 900/1000 = 0.9. Hence, of the 200 true errors made, (0.1)(200) = 20 are undetectable and (0.9)(200) = 180 are detectable. Error-editing will yield two decks; the first will contain 180 cards which are all true errors, and the second will contain 9820 cards of which 20 are undetectable errors.

To extend the illustration to alpha codes: assume the coding is alpha, with a 10-per-cent true error rate. With 26 letters, the number of possible discriminations is 17,576. In this case, 0.005 of the errors will not be detectable (100/17,576) and 0.995 will be (17,476/17,576). Of the 1,000 cards in error, 5 will be undetectable errors and 995 detectable. We end with two decks: the first contains 995 true errors; the second contains 9,005 cards, of which 5 are errors. Comparing the two conditions: the numeric codes yield a working deck of 9,820 cards containing 20 erroneous cards, while the alpha codes yield a working deck of 9,005 cards of which 5 are in error. A crisper decision (quantity versus cleanness) is now possible.

This, then, is what we call the error-discernibility factor. It is based on the number of discriminations possible (which in turn is related to the information content), the number of discriminations used, and the error rate of the code in use.

We have used the concept of "true error" to facilitate the discussion. In the experimental conditions we could compare stimulus and response and determine the true error. In actual practice this is not possible. It is possible, however, to obtain an "estimated true error rate" for real-world data by taking advantage of the error discernibility factor. Using the numeric example given above, the deck of true and detectable errors contains 180 cards or 1.8 per cent of the total (i.e., 180/10,000 = 1.8 per cent). The detectability factor is 0.9; hence the 1.8 per cent represents 9/10 of the errors; hence (1.8 per cent)(10/9) = 2.0 per cent, the estimated true error rate.

The importance of the discernibility factor is perhaps best illustrated by the "action-taken" codes of AFM 66-1. These single-character codes have always been among the high-error contributors. One clue is given by the fact that both alpha and numeric codes are used -- at present, 8 numeric and 22 alpha. Since alpha "Ø" and "I" are outlawed (leaving a 24-letter alphabet), the action-taken codes use 30 of 34 possible combinations. This implies that 30/34 of the errors will not be detected and that the error-detectability factor will be 4/34, i.e., that the cards isolated as errors represent 4/34 of the total errors.

The particular volume of the AFLC-K-97 error analysis used as background research in this study shows the detected error rate to be 3.86 per cent. The cards are checked for the legal alpha characters F through Z (except I and Ø); hence the detectability factor for these cards is 15/34. (F-Z less I and Ø = 19; 34 - 19 = 15.) The true error rate can be approximated as (3.86)(34/15) = 8.75 (assuming that 3.86 is representative for these kinds of cards). Thus the true error rate is over twice what it appears to be.

The AFM 66-1 "when-discovered" codes show even greater discrepancies than those of the "action-taken" codes. Twenty-two of the 24 legal alpha codes are used, yielding a detectability ratio of 2/24. (Illegal numeric codes appear so infrequently as not to be a factor.) The detected error rate was 1.08 per cent in the report cited previously. The true error is 24/2 times this or slightly less than 13 per cent.

We should hasten to add that this estimated "true error rate" is undoubtedly conservative. It is based on the idea that the errors are random. We doubt this. If a man makes an error there is a good possibility that he will enter a code that is among the list of legal codes. This is simple translation of current learning theory. Hence, the undetectable-error rate will be higher than can be estimated by the error-discernibility factor. Unfortunately, since this involves a learning factor, we could not include it in our study. It is an experiment that needs to be done, however.

VI. CONCLUSIONS

The following conclusions appear to be warranted by the study:

(1) Perceptual set affects the amount of coding error. Depending on the circumstances, perceptual set may increase or decrease error.

(2) Coding errors are proportional to the amount of alpha content; numeric codes have the least error; as alpha content increases, coding error increases.

(3) The relative position of the alpha and numeric characters within a mixed code has an effect: the "singleton-in-the middle" condition results in increased error rates.

(4) There is evidence of interaction among these major variables.

(5) Most errors result from the "substitution of opposites" (i.e., numbers for letters and vice versa) unless the possibilities for it are controlled by controlling perceptual set.

(6) Erroneous substitutions appear to be of two kinds: (a) random, in which the substitution is not predictable; and (b) constant, in which a specific character tends to be repeatedly substituted for another (e.g., "V" for "U").

(7) Constant substitutions often show unidirectionality (e.g., "V" for "U" much more often than the reverse).

(8) Single-digit substitutions account for most (75 to 95 per cent) coding errors.

(9) The use of mnemonic codes does not reduce coding error.

(10) Letter-pattern familiarity affects coding error; familiar letter patterns (i.e., high digram values) show reduced error rates.

(11) Coding errors can be reduced by providing keypunch operators with knowledge of the codes. This effect is greatest with mnemonic codes.

(12) The amount of usable information that is retrievable is related not only to the error rate but also to the error-detectability factor; this latter concept includes consideration of the ratio of the number of codes used to the number of possible codes.