AN AIDED ADAPTIVE CHARACTER READER FOR
MACHINE TRANSLATION OF LANGUAGES

Paul Baran
Mathematics Division
The RAND Corporation

Gerald Estrin*

P-1990

May 17, 1960

*University of California at Los Angeles,
California

Reproduced by

The RAND Corporation • Santa Monica • California

The views expressed in this paper are not necessarily those of the Corporation

SUMMARY

An elementary procedure for the synthesis of a character recognition device based upon a learning experiment is described. Utilizing information derived from a significant sample of the set of characters to be read and given identification of the samples by a human operator, a computer defines a set of "filters." These filters may then be used to transform unknown characters having similar type characteristics. During the recognition process a probability matrix for each character in the alphabet is used to compute a figure of merit for the hypothesis that an unknown character is the same as a known character.

It is shown that this elementary model might adequately serve for construction of a fast input device for a language translation machine if it were able to make use of frequency distribution characteristics of the dictionary. A possible implementation with a raw character reading rate of up to 500 characters a second appears feasible.

INTRODUCTION

Part of this work was performed in partial fulfillment of the requirements of the M.S. degree in Engineering at UCLA. A companion paper P-1989 describes a digital simulation of the proposed recognition system.

AN AIDED ADAPTIVE CHARACTER READER FOR MACHINE TRANSLATION OF LANGUAGES

HISTORY

There have been several approaches proposed to build reading machines. Two of the favorite methods of constructing such machines in the past have been the simple stencil matching method¹ and the examination of critical points.² In the basic stencil matching method, stencils of the exact shape and size of the character to be recognized are sequentially placed in front of a light source. The intent is that when the character to be identified and the testing stencil are exactly coincident upon one another there is an eclipse and no light is reflected from the examined page. The shortcomings of this method include the need for precise position alignment, and loss of effectiveness when encountering imperfectly printed samples.

The critical points method is a modified form of the simple stencil matching method and has been used to build devices capable of separating a highly stylized set of characters.³ In this method use is made of the knowledge that the information that differentiates each character is not uniformly spatially distributed, and often examination of only a relatively small number of points of a field containing an unknown character is sufficient to differentiate between characters. For example, if one were to distinguish characters of the limited set of the letters "E," the letter "F" and the letter "H" as shown in Figure 1, examination of only two separate points is necessary,

Figure 1

Figure 2

which in this case are shown in lower case letters as examination point "a" and examination point "b." If a black spot is noted for examination point "b" and not "a" the character "E" is implied, similarly not "a" and not "b" implies the character "F," while the presence of "a" but not "b" implies the character "H." By modifying the characters into a highly stylized form more certain differentiation is assumed in the event of a slight positional misalignment or bad printing. In Figure 2, for example, the three characters may be shifted further than in Figure 1 before errors occur. The successful workable character reading machines of today use somewhat distorted or stylized characters to insure successful operation.⁴

The shortcomings of such a method in the language reading application are severe. It is necessary to program a new set of examination points for each type face and alphabet considered. Rather than doing this manually it would be well to automate the formation of the examined points. Before this can be done several interesting questions arise.

What is the best configuration for the examination points? How many points should be examined? Should all points be given the same weight in reaching a conclusion if contradictory patterns occur?

The search for the answers to these problems led to the analysis described in Appendix A.

DESCRIPTION OF THE RECOGNITION METHOD:

Based upon the analysis derived in the Appendix, the basic recognition mechanism is divided into two separate operations. The first is a determination of the probability of occurrence of an ink spot in each elementary sub-area (cell) of a large number of samples of each possible character. The second step is the examination of all cells of the unknown character against the previously determined history of ink spot distribution for each possible character. In more detail, first the a priori probability distribution of each black and white cell for each of the possible characters is computed. Next, a number of hypotheses are established:

H_A = the unknown character is the character "A"

H_B = the unknown character is the character "B"

H_C = the unknown character is the character "C"

.

.

H_Z = the unknown character is the character "z"

Next, the a posteriori probability for each of these hypotheses being true is tested with Bayes' formula (Eq. 5, Appendix A) using the data from each cell position as a separate measurement. The probability of an ink spot occurring in each cell of the unknown character, if it were each of the possible characters, is examined. The measurement of each cell is regarded as an independent measurement. The assumption of

independence is made for simplicity of computer implementation.

The second step in the recognition is the combination of all the a posteriori probabilities for each cell. If each cell is assumed to be independent, then total probability of hypothesis traits is the product of all measurements. The computation of Bayes or a posteriori probabilities requires multiplication and division operations. These are normally slow computer operations. Instead of computing products, the component probabilities are converted into their logarithmic values which permits addition to be used in lieu of multiplication, allowing higher-speed data processing when the unknown characters are being scanned. The operations of division and multiplication are performed only during the time of loading the memory with a new print format.

In the following a mechanization of the above process is described; the characteristics of printed text are introduced to improve performance, and, finally, the use of separate parent and offspring devices are considered to simplify the process.

Summarizing the actions that must be performed by the machine:

Step I: Formation of the Expected Probabilities

1. Insert into the system memory many sample illustrations of each of the characters to be recognized.

See Figure 3.

2. Automatically maintain the position of each sample character so as to be in an approximate rectilinear alignment within a rectangular matrix analyzing field. See Figure 4.
3. Automatically examine each possible character of the set of samples sequentially with reference to the analyzing matrix field, and compute the probability of the absence or presence of a mark in each cell of the analyzing matrix for the entire set of samples of each possible character. See Figure 5.

Step II: Reading Unknown Characters

1. Compare the absence or presence of marks in each cell of the unknown character against the probability of such occurrence in similar positioned cells of each possible character held in the system memory.
2. Compute the most likely match based on the history of the characters in the store.

Figure 3 - Recognition System

Figure 4 - Character to be Analyzed
Projected on an Analyzing
Matrix

Figure 5

MACHINE IMPLEMENTATION:

The reading machine proposed, contains a keyboard, an electronic scanner, a viewing screen, a memory unit and a computational unit and is shown in Figure 5. Input text is examined by the scanner and displayed on the viewing screen character by character. An operator examines each separate character so displayed and presses the corresponding key of the keyboard. Pressing any key causes the next character to be displayed on the viewing screen.

The scanner, which may be a cathode ray flying spot scanner, for example, permits examination of each character displayed upon the viewing screen.

This output signal is a serial representation of the presence or absence of ink spots for each cell of a rectangular field surrounding each character. This signal from the scanner is added to the signals from identical characters stored in the memory unit. Thus, the memory contains, at this time, the total number of filled ink spots for a large number of samples so manually examined.

After a period of manually examining about 50 or so samples of each character the operator actuates a switch disconnecting the manual entry keyboard and enters the automatic mode of operation.

The automatic mode commences with the output of the memory unit feeding the computational unit. First, the a priori probability of each cell of an unknown character being a

black spot is computed by adding the total number of "hits" for each cell for all possible characters and then dividing by the number of possible characters. This value is then stored in the memory. Next, all values of hit counts held in the memory are converted into logarithmic values by use of the computational unit. The resulting values are stored back into the same memory locations as the previous hit counts. Next, the a priori probabilities are subtracted for each cell for all characters. The machine has now processed its data and is ready to commence automatic reading with only the addition operation required at the actual comparison stage.

Each unknown character of the text to be read is examined by the electronic scanner unit, and a serial representation of the inked spots found in each cell of the rectangular field surrounding the unknown character is stored in the memory unit. After storage of a single unknown character the actual recognition takes place by the comparison of the unknown character so stored against the cumulative history of all possible characters. A separate register is provided in the memory unit for each possible character. A "score of match" is computed for each possible character. The score of match is a function of the probability of the unknown character truly being the character presumed and is explained in Appendix A.

After the comparison, the scores of match for all possible characters are examined and the highest score chosen as the most likely correct "guess." Since it is not expected that

each and every "guess" will be correct, two additional operations are necessary to make the machine a useful language processor. The first additional operation utilizes the knowledge that the value of the highest score of match for the "guessed" character provides a good estimate of the confidence of the estimate so a measure of the certainty of each and every guess is known. The second aid is the use of the language dictionary of the language translating machine itself as a second filter to indicate wrong guesses to permit re-reading or re-guessing. Both of these techniques are described in the next section.

PROBABLE SUCCESS AS AN INPUT TO A LANGUAGE PROCESSING MACHINE:

A language processing machine requiring perfect accuracy on a character by character basis is almost certainly doomed to failure. For example, certain German characters are so similar as almost to require a magnifying glass in order to be positively differentiated.

In order to build machines that can recognize words the tremendous redundancy present in language may be used to indicate errors by the simple expedient of searching the memory store of the language processor on a word by word basis. As will be shown, if the word read is found in the dictionary, then all the characters of that word may be assumed to be correctly read. If not, the presumption exists that either one or more characters have been incorrectly read or that the word is not contained in the dictionary used. Assuming the

use of a reasonably large-sized dictionary, the latter possibility may be neglected for the moment. Figure 7 shows the probability of the reading machine reading an entire word correctly on the first trial versus the number of characters per word or the length of the word. Curves are shown for three separate values of P_{rc} , the probability of the reading machine reading a character correctly on its first trial.

A given character recognition machine will produce different P_{rc} 's (i.e., the probability of reading a character correctly on the first trial), which will be a function not only of the machine but also of the set of characters to be differentiated. In a digital simulation⁵, a badly deformed set of ten characters produced a surprisingly good P_{rc} of over 0.9. (With benefit of the improvements possible by positional translation and similar algorithms, higher P_{rc} 's appear obtainable.) In Figure 6, for example, if a reading machine-text combination produced a P_{rc} of 0.9 then 60 per cent of the five letter words would be read correctly on the first trial, and 40 per cent would be incorrectly read. We next seek to determine which words have been correctly read and which have not been, to permit focus of attention upon the incorrectly read words. To do this, we examine the effectiveness of the dictionary as a filter.

Figure 7, drawn on 12-cycle log paper, shows three separate curves plotted against N number of characters per word. The first, curve No. 1, is the number of combinations of characters possible. That is, given 26 possible symbols

P_{RC} = The probability of the reading machine reading a character correctly on the first trial.

P_{RW} = The probability of the reading machine reading an entire word correctly on the first trial.

Figure 6

for each of the characters 1, 2, 3, . . . N, the number of combinations that may be formed is $Y = C^N$ where C is the number of possible symbols for each character and N is the number of characters per word.

Curve No. 2 shows the number of valid words D^6 in the dictionary that can be formed with N separate characters.

Curve No. 3 shows the ratio, $R = \frac{Y}{D}$,

$\frac{(\text{valid words plus invalid combinations})}{(\text{valid words})}$, using the data of

Curve No. 1 and Curve No. 2.

Since most incorrect words are not to be found in the dictionary, we may compute

$$\frac{1}{R} = \frac{D}{Y}.$$

This is the probability of accepting an incorrect word as being correct after examining a read word against dictionary entries.

$$\frac{Y-D}{Y}$$

is the probability of rejecting an incorrect word after examining a read word against dictionary entries.

Thus, the size of the dictionary is a critical factor in filtering out invalid word combinations. What then is the optimum size of the dictionary to be used in the language translator? If the dictionary is too large, it loses its effectiveness as an error filter. If it is too small, the search will be slowed down as iterative attempts are made to find non-existent words.

It would appear that the error component due to inability to find a word in a dictionary should approximately match that component of error due to imperfect dictionary filtering to produce a resultant minimum r.m.s. value of total error. Thus, semi-intuitively, it appears that the dictionary used by an imperfect-reader-language-translator should be smaller than one used by a perfect-input-language-translator. Note, however, that the optimum size of the dictionary should be determined separately for each length of word set. Expected frequency of use also is an important quantity in this determination. Re-stated, the optimum sized dictionary for short words should not contain rarely encountered forms of short words while the optimum size dictionary for long words should contain all rarely encountered words. Cost and access time of the memory are the controlling factors.

Referring to Figure 7, it can be seen that the probability of accepting an invalid character combination is very remote (less than 1 in 2×10^7) for long words (equal to or greater than 7 letters). The value of R for long words is so great that even an increase in the size of the dictionary may be necessary to produce a minimum over-all error rate because of the powerful dictionary filtering performed. However, in the case of shorter words, $N \leq 3$, elimination of rarely encountered forms in the dictionary is indicated since the dictionary filtering effectiveness for short words is very low, $R \sim 25$. Thus, we would expect the error limitation of the reading machine to be caused by short words. In any case, we would expect the information

Figure 7

content ($\sum p_i \log p_i$) of short words to be low because of their high frequencies of occurrence.

Tachistoscopic reading experiments⁷ appear to confirm this assumption as they indicate that a good reader does not really "read" short words but reads individual words only when comparatively rare ambiguities exist. In other words, the high redundancy that exists in a series of words in a sentence permits the good reader to "scan" the text rather than reading word by word so that the occasional loss of a small word adds little error to the read text.

Thus, it is suspected that a suitable form of output for the language reading machine should utilize repeated trials to find the longer high information content words in the dictionary, since the long word error filter efficiency is high. Short words not found in the dictionary in a single trial should be typed out letter by letter. A reader familiar with the subject matter of the text should encounter little added difficulty reading the resultant text since it is expected that the semantic contribution of the smaller words is low. In those relatively few ambiguous cases where the small word becomes critical, only a single character will be expected to be in error even with the preliminary figures of raw character reading rates, P_{rc} , as found in the simulation.⁸ As the number of short words in a language (not frequency of use) is small, the post-editor translator⁹ would probably be able to acquire rapidly a knowledge of this small group of words to resolve most of the residual errors of this nature.

Note that this does not imply that a short word in language A being translated will always be a short word in language B. All that has been said is that a short word in language A probably contains little information, and its partial loss will rarely hinder translation. In such cases as it is needed, the original language is printed out letter by letter, and one or, rarely, perhaps two letters will be incorrect. In any event those letters read incorrectly will be outputted exactly as the machine thinks it sees such letters. For example, translating from English to French, our language translator made a mistake distinguishing between "e" and "o," while reading the word "the" so that "the" was incorrectly read by the machine as "tho." Since it is a small word, it would be outputted as "tho" (underlined to distinguish words not found in the dictionary). On reading the text in French, the post-editor easily would be able to reconstruct the meaning of the text since this false word contains so little information. However, in such cases where the misread word is needed to resolve an ambiguity, the range of possible meanings may be reduced by examining the word "tho" printed out. Note also that when the reading machine makes a mistake, it will output a character that resembles the correct letter, further reducing the range of uncertainty. Tests to verify these assumptions, however, were beyond the scope of this paper.

One of the most important questions that must be asked is, "How effective is the score of match technique for indicating

which characters are probably in error and must be re-examined?"

To evaluate the effectiveness of the proposed techniques it is necessary to compute the cumulative multiple word reading error rate; i.e., the cumulative probability of the reading machine reading a word and finally guessing its value as a function of the number of trials. Unfortunately, this computation requires a knowledge of the difficulty of the text, the size of the dictionary, and the length-frequency distribution of the particular language used. These values are best obtained by measurement.

However, to provide a crude estimate of the effectiveness of the proposed technique, the following extrapolation rationale was used. A value of raw character correct reading rate of about 0.9 was found using a set of 10 noisy, badly shifted characters whose type width varied over a wide range.¹⁰ Further, the measured value of the score of match proved to possess a powerful capability for indicating probable errors.

Suppose the same raw character correct reading rate could be obtained using a set of 70 (normal language) better printed characters as normally found in books and newspapers. In such an eventuality, how would such a machine perform when reading known English text?

ESTIMATE OF PERFORMANCE:

(1) Extremely bad single character error rates are tolerable when encountering long words (six characters or more) provided the machine is "fast" and can retry other possible combinations.

(2) Short words of 1, 2, or 3 letters will be the source of most of the errors. As a worst case limiting figure, for English, a one-letter word will be incorrectly read about once in 166 one-letter words encountered with a reading machine with a P_{rc} of 0.9. The composite error rate for long and short words is probably of the same order of magnitude that a human typist would make. The worst case error rate was computed as follows:

The probability of an undetected error passing through the machine is $\frac{Q_{rc}}{R}$, where

Q_{rc} = raw character reading error rate

$\frac{1}{R}$ = probability of a randomly chosen set of N characters being found in the dictionary.

Thus, the probability of not permitting the undetected error to pass through the machine is

$$P_x = 1 - \frac{Q_{rc}}{R}$$

$$P_x = 1 - (1 - P_{rc}) \left(\frac{D}{Y} \right).$$

For different values of P_{rc} , raw character reading rates are:

P_{rc}	P_x	Errors/1000 one-letter words
0.95	0.997	3
0.9	0.994	6
0.8	0.988	12
0.7	0.982	18

(3) If the machine makes no attempt to reread short words that produce errors but rather outputs the "guesstimated" individual characters, linguistic errors would be rare.

(4) If, for a well printed foreign text, P_{rc} 's develop as high as those found in the study of a limited, but very noisy set of characters, then a practical aided adaption reading machine may be feasible.

THE SIBLING MACHINE:

The machine so far described possesses the attribute of aided adaption to permit recognition of a wide variety of type faces. Although this degree of flexibility was the goal sought in the original generation of the system concept, there is a large number of applications where either the same or a limited number of type faces is expected. To avoid the necessity of typing a new sample of text each time the reading machine is used, the data describing the optimum stencil may be stored on a permanent, but removable, memory medium. After a cursory examination of the type face used the proper memory unit is placed into the reading machine. For example, one form of "stripped down" machine envisioned is the complete parent machine less the manual entry keyboard plus a magnetic tape playback unit to permit loading of the memory.

An even greater saving of equipment would result from using a photographic medium of storage. The necessity of scanning the characters to be recognized may be eliminated by organizing the photographic memory in the form of stencils. Light

projected through the stencils would permit predetermined weights to be assigned to each character cell.

Thus, although all the operations described may be performed on a single machine, it appears more convenient to divide the functions and use two separate mechanisms. These two machines are called the "parent" or "sire" machine and the "offspring" machine.

The sire machine performs the digital analysis of a set of known characters to form a history of possible shapes to be encountered. This history is used to form the basis of a later automatic recognition of unknown characters. The digital information representing the history of known characters is then used to establish the optical transparencies for use on a simpler machine known as the "offspring" machine. Although this form of offspring machine greatly resembles the simple stencil machine described in the introduction of this paper, there is a major difference. The offspring stencil used is not of the outline of the character being recognized but is distorted to emphasize those subareas which are significant in differentiating one character from all others.

One problem that is encountered in the manufacture of these stencils photographically is that photographic film light density is a most difficult quantity to control accurately (although being logarithmic in form). It appears possible to utilize the high inherent resolution of the photographic film in the form of a series of white dots (or black dots) to

obscure portions of an area corresponding to a fixed fractional portion of a cell so as to provide a reproducible and controllable light mask. Thus, an "area modulated" stencil is prepared where the relative areas between white and black of a larger area corresponding to the amplitude of the cell of the quantizing field is produced. Many such masks can be prepared from a single master so that stock photographic stencil masks may be used by simple reading machines to read a given type face.

The chief virtues of such sibling machines is that they are simple, fast, and theoretically provide the same high degree of effectiveness of character separation as the full-size machine without the necessity of readaption of memory each time before use.

Studies of the implementation of the reading machines described reveal that raw character reading rates of the order of 80 to 500 characters per second, not including paper moving speed, appear feasible with a parent machine having about 4000 transistors. It is, of course, conceivable to have a general purpose computer design the filter for the sibling machine. The reading speed of the sibling machine should be even higher, with the upper bound set by paper movement and alignment considerations.

REFERENCES

1. Goldberg, E., "Statistical Machine," U.S. Patent No. 1,838,389, December 31, 1931.
Handel, P. W., "Statistical Machine," U.S. Patent No. 1,915,993, June 27, 1955 (C1 235-92).
2. Weaver, A., "Telegraph Reading Machine," U.S. Patent No. 1,815,996, July 28, 1931 (C1 178-4).
Parker, R. D., "Telegraph Reading Machine," U.S. Patent No. 1,815,986, July 28, 1931 (C1 178-4).
Shepard, D. H., "Apparatus for Reading," U.S. Patent No. 2,663,758, December 22, 1953 (C1 178-17).
Shepard, D. H., and C. C. Heasley, "Photoelectric Reader Feeds Business Machines," Electronics, February 1956.
3. Harrel, Russel L., "The Role of Character Recognition Devices in Data Processing Systems," Intelligent Machines Research Corporation.
4. Ibid.
5. Baran, P., "A Probability Matrix Pattern Comparator," M.S. Thesis, University of California at Los Angeles, May 1959.
6. Oxford Dictionary.
7. Cherry, C., On Human Communication: A Review, A Survey and A Criticism, Wiley, 1956.
8. Baran, P., Op Cit.
9. Locke, W. N., and A. D. Booth, eds Machine Translation of Languages, Technology Press of MIT, and John Wiley and Sons, 1955, p. 206.
10. Baran, P., Op Cit.

Appendix A

THE THEORETICAL BASIS OF THE RECOGNITION SYSTEM

NOMENCLATURE

A pattern representing an unknown character, U , is spatially divided into a set of R cells where the r 'th cell contains a measured value corresponding to the presence or absence of an ink spot for each of the R cells, $r = 1, 2, 3, \dots, R$. (See Fig. 8).

Figure 8

The probability density distribution of ink spots for known text is measured for a large sample for each c 'th character of the set of C characters, $c = 1, 2, 3, \dots, C$. This probability density is computed on a cell by cell basis. For each r 'th cell there is a value representing an estimate of the probability of the occurrence of an ink spot in the r 'th cell given the c 'th character. In Figure 9 the density of black diagrammatically illustrates this distribution.

We seek to determine the value of c , or indicate which of $c = 1, 2, 3, \dots, C$ characters most closely resembles the unknown character, U . Thus, we establish a set of hypotheses:

$$H_0 : U = 0, H_1 : U = 1, H_2 : U = 2, \dots H_C : U = C \quad (1)$$

Next, all hypotheses are tested, and the ones with the highest probability of being true are selected. The chosen hypotheses are ordered and then issued as the output of the recognition machine, together with a value called the "score of match" for each test.

In the following synthesis no assumption is made as to the number of dimensions of the pattern, as the values of $r = 1, 2, 3, \dots, R$ may be grouped in any manner to represent values in any number of dimensions. Thus a time-varying as well as a fixed spatial pattern can be handled by the following analysis without modification. For convenience, the binary assumption is made that each cell either contains an ink spot, if more than

Character #1
c=1

c=2

c=C

Figure 9

50 per cent of the cell is black, or otherwise it is called white. A multilevel or shades of grey approach could have been used as well at the price of increased complexity.

There are three measured distributions that are useful to the reading machine:

1. The first measured distribution is the probability that the area corresponding to a single cell, r , is black assuming that we are examining the c 'th character, $P_r(b_r|c)$.

To make it easier to visualize the notation used:

2. The second known quantity is the a priori probability of the occurrence of each of c characters from a population of C different characters = $p(c)$. If, for example in text, 12 per cent of the 26 different characters were "A's" then $p(A) = 0.12$ where $A = c$ and $C = 26$.

3. The third known distribution is the knowledge derived from the measurement of ink occupancy for each cell of the "unknown" character being examined. The assumption is made in the following that each such cell examined constitutes an independent measurement. This assumption, while less than wholly satisfactory, is necessary to reduce machine complexity.

For each such measurement we seek, $p(c|b_r)$. That is, what is the probability of the unknown character "U" being "c" if the r'th cell was measured and found to be black?

At the start of the process we know $p(b_r|c)$ but not $p(c|b_r)$. $p(c|b_r)$, however, may easily be derived to create Bayes' well-known equation.

$$p(b_r c) \equiv p(c)p(b_r|c) \equiv p(b_r)p(c|b_r). \quad (2)$$

Thus,

$$p(c|b_r) = \frac{p(c)p(b_r|c)}{p(b_r)}. \quad (3)$$

A few definitions are now in order:

c_1 is defined as the character, 1

c_2 is defined as the character, 2

c_i is defined as the character, i

Note that c_i has a different meaning than c . c_i is an unknown character and uniquely refers to the i'th character.

Since b_r may occur not only for character c but also for the occurrence of any character, c_i , $i = 1, 2, 3, \dots, C$, and further, these are mutually exclusive events:

$$\begin{aligned} p(b_r) &= p(b_r c_1) + p(b_r c_2) + p(b_r c_3) + \dots + p(b_r c_C) \quad (4) \\ &= p(c_1)p(b_r|c_1) + p(c_2)p(b_r|c_2) + \dots + p(c_C)p(b_r|c_C) \\ &= \sum_{i=1}^C p(c_i)p(b_r|c_i). \end{aligned}$$

Inserting this value of b_r into equation 3,

$$p(c|b_r) = \frac{p(c)p(b_r|c)}{\sum_{i=1}^C p(c)p(b_r|c_i)} \quad (5)$$

Since we wish to extract useful information when the cell of the unknown character is also white, we may write a second equation for such a case by replacing w_r in lieu of b_r in the above equation where w_r = the presence of a white cell in the r 'th cell position of the unknown character. w_r and b_r are mutually exclusive conditions so we may state the probability of the truth of the original hypothesis, $H_c: U = c$, $p_r(H_c) =$ a posteriori probability of the truth of the hypothesis, $H_c = c$, based entirely upon the information derived in the examination of a single cell, r .

$$p_r(H_c) = \begin{cases} p(c|b_r), & \text{if cell "r" of } U = \text{black} \\ p(c|w_r), & \text{if cell "r" of } U = \text{white} . \end{cases} \quad (6)$$

To provide a more compact notation, let

$U_r = 1$ represent "if cell r of U is black"

$U_r = 0$ represent "if cell r of U is white."

Next, by Bayes' Rule:

$$p_r(H_c) = \begin{cases} \frac{p(b_r|c)p(c)}{\sum_{i=1}^C p(b_r|c_i)p(c_i)} , & U_r = 1 \\ \frac{p(w_r|c)p(c)}{\sum_{i=1}^C p(w_r|c_i)p(c_i)} , & U_r = 0 . \end{cases} \quad (7)$$

To convert all statements into terms of black cells only,
use:

$$p(w_r|c) + p(b_r|c) = 1, \text{ since they are mutually exclusive.}$$

Similarly:

$$p(w_r|c_i) + p(b_r|c_i) = 1 \quad (8)$$

$$p_r(H_c) = \begin{cases} \frac{p(b_r|c)p(c)}{\sum_{i=1}^C p(b_r|c_i)p(c_i)} , & U_r = 1 \\ \frac{p[1 - p(b_r|c)] p(c)}{\sum_{i=1}^C [1 - p(b_r|c_i)] p(c_i)} , & U_r = 0 . \end{cases} \quad (9)$$

For any c , the denominators of equation (9) are constants for each separate cell, r . Hence, for later convenience, let:

$$k_{r_b} = \frac{1}{\sum_{i=1}^C p(b_r|c_i)p(c_i)} \quad (10)$$

$$k_{r_w} = \frac{1}{\sum_{i=1}^C [1 - p(b_r|c_i)] p(c_i)} \quad (11)$$

$$\therefore p_r(H_c) = \begin{cases} k_{r_b} p(b_r|c)p(c), & U_r = 1 \\ k_{r_w} [1 - p(b_r|c)] p(c), & U_r = 0. \end{cases} \quad (12)$$

To this point we have investigated the probability of the hypothesis being correct solely on the basis of the examination of a single cell of the unknown character. Much more information is yet to be extracted by examining all of the other cells of the unknown character, U.

Rewriting the previous equation and factoring out, p(c):

$$p_r(H_c) = p(c) \begin{cases} k_{r_b} p(b_r|c), & U_r = 1 \\ k_{r_w} [1 - p(b_r|c)], & U_r = 0. \end{cases} \quad (13)$$

Note that only the upper term may exist, or that the lower term may exist, but never both simultaneously. This equation defines the probability of the truth of the hypothesis on the basis of the information content of the r 'th cell only. If the assumption is made that each cell is an independent experiment from which to determine, $H_c: U = c$, with ever increasing certainty where the true (unknown) value of c remains fixed throughout the trials, then the a posteriori probability for the last examined cell becomes the a priori probability for the following cell examined.

For $r = 1$:

$$p_1(H_c) = p(c) \left\{ \begin{array}{l} k_{1_b} p(b_1|c), U_1 = 1 \\ k_{1_w} [1 - p(b_1|c)], U_1 = 0 \end{array} \right. \quad (14)$$

= the posteriori probability using information based upon the examination of the first cell only. Now, using the data from the first cell to form the a priori probability for the second cell and applying Bayes' Rule:

$$p_2(H_c) = \begin{cases} \frac{p(b_2|c)p(c')}{\sum_{i=1}^C p(b_2|c_i)p(c_i)} , U_2 = 1 \\ \\ \frac{p(w_2|c)p(c')}{\sum_{i=1}^C p(w_2|c_i)p(c_i)} , U_2 = 0 \end{cases} \quad (15)$$

where $p(c')$ is the "new" a priori probability of the unknown character being the c' th character based upon better information than the "old" a priori probability, $p(c)$.

But:

$$p(c') = p_1(H_c)$$

$$p_2(H_c) = \begin{cases} \frac{p(b_2|c)p_1(H_c)}{\sum_{i=1}^C p(b_2|c_i)p(c_i)} , U_2 = 1 \\ \\ \frac{p(w_2|c)p_1(H_c)}{\sum_{i=1}^C p(w_2|c_i)p(c_i)} , U_2 = 0 . \end{cases} \quad (16)$$

Repeating this process using all the information in all
R cells:

$$p_R(H_c) = \left\{ \begin{array}{l} k_{R_b} p(b_R|c), U_R = 1 \\ k_{R_w} p(w_R|c), U_R = 0 \end{array} \right\} \left\{ \begin{array}{l} k_{(R-1)_b} p(b_{(R-1)}|c), U_{(R-1)} = 1 \\ k_{(R-1)_w} p(b_{(R-1)}|c), U_{(R-1)} = 0 \end{array} \right\} \dots \quad (17)$$

$$\dots \left\{ \begin{array}{l} k_{2_b} p(b_2|c), U_2 = 1 \\ k_{2_w} p(w_2|c), U_2 = 0 \end{array} \right\} \left\{ \begin{array}{l} k_{1_b} p(b_1|c), U_1 = 1 \\ k_{1_w} p(w_1|c), U_1 = 0 \end{array} \right\} p(c)$$

$$= p(c) \prod_{r=1}^R \left\{ \begin{array}{l} k_{r_b} p(b_r|c), U_r = 1 \\ k_{r_w} p(w_r|c), U_r = 0 \end{array} \right\} \quad (18)$$

It is to be noted that the quantity, k_{r_b} , is a function of the "importance" of each cell when the cell is black. Hence, this quantity is named the "black cell weight factor." It is not a function of the individual character being examined. For example, the 95th cell of any character would always have the same value of black cell weight factor if it were black, regardless of whatever character is chosen. The quantity $p_r(b_r|c)$ on the other hand, the probability that the given cell of a given character is black, is a function of both the cell position and the character. The same is true with the corresponding quantities k_{r_w} and $p_r(w_r|c)$.

Since the criterion for selection of the most likely hypothesis is the selection of the hypothesis with the highest probability of truth, several equipment implementation short cuts are possible.

$$p_R(H_c)_1 > p_R(H_c)_2 > p_R(H_c)_3 \quad (19)$$

is a monotonic decreasing series representing the ranked probability of match for the highest "score of match," the second highest and the third highest. Thus:

$$k_1 p_R(H_c)_1 > k_1 p_R(H_c)_2 > k_1 p_R(H_c)_3, \text{ for } k_1 \text{ greater than } 0 \quad (20)$$

$$\text{and, } \log_{k_2} k_1 p_R(H_c)_1 > \log_{k_2} k_1 p_R(H_c)_2 > \log_{k_2} k_1 p_R(H_c)_3 \quad (21)$$

k_1 and k_2 are arbitrary constants; thus, our computation may be performed by the addition of logarithms and without the

necessity of either multiplication or conversion into logarithmic values during the period the ordering comparison is made.

A further timesaver is that the individual character weighting channels need not be normalized in terms of exact values of probabilities, and the use of relative values will suffice.

Hence, we are interested in the function:

$$f_1 [p_R(H_c)] = \text{Log}_{k_2} k_1 p_R(H_c) \quad (22)$$

The absolute "score of match" for the c'th character, $S_c' =$

$$\begin{aligned} \text{Log } p_R(H_c) &= \text{Log} \left\{ p(c) \prod_{r=1}^R \left\{ \begin{array}{l} k_{r_b} p(b_r|c), U_r = 1 \\ k_{r_w} p(w_r|c), U_r = 0 \end{array} \right\} \right\} \quad (23) \\ &= \text{Log } p(c) + \sum_{r=1}^R \text{Log} \left\{ \begin{array}{l} k_{r_b} p(b_r|c), U_r = 1 \\ k_{r_w} p(w_r|c), U_r = 0 \end{array} \right\} \\ &= \text{Log } p(c) + \sum_{r=1}^R \text{Log} \left\{ \begin{array}{l} k_{r_b}, U_r = 1 \\ k_{r_w}, U_r = 0 \end{array} \right\} + \sum_{r=1}^R \text{Log} \left\{ \begin{array}{l} p(b_r|c), U_r = 1 \\ p(w_r|c), U_r = 0 \end{array} \right\} \end{aligned}$$

By noting that $U_r = 1$ and $U_r = 0$ are mutually exclusive conditions and "defining" the nonexistent $\text{Log } 0 = 0$ for ease of implementation, S_c' may be described as a sum of five terms.

(The rationale in setting Log 0 to 0 is that if an infinitely large sample were available upon which to estimate the probability density distribution, then Log 0 would never occur. This step merely provides a minimum value for the case of less than infinite size distribution.)

$$\begin{aligned}
 S_c' = & \text{Log } p(c) + \sum_{r=1}^R \text{Log } k_{r_b}, U_r = 1 \\
 & + \sum_{r=1}^R \text{Log } k_{r_w}, U_r = 0 \\
 & + \sum_{r=1}^R \text{Log } p(b_r|c), U_r = 1 \\
 & + \sum_{r=1}^R \text{Log } p(w_r|c), U_r = 0.
 \end{aligned} \tag{25}$$

To evaluate Log k_{r_b} :

$$\begin{aligned}
 k_{r_b} &= \frac{1}{\sum_{i=1}^C p(b_r|c_i)p(c_i)}, U_r = 1 \\
 \text{Log } k_{r_b} &= - \text{Log} \sum_{i=1}^C p(b_r|c_i)p(c_i), U_r = 1
 \end{aligned} \tag{26}$$

In the digital simulation performed, the number of samples of each character was the same, so the following simplification was permitted:

$$p(c_i) = 1/C, (i = 1, 2, 3, \dots, C), \text{ since,} \quad (27)$$

$$\sum_{i=1}^C p(c_i) = 1, \text{ thus:} \quad (28)$$

$$\text{Log } k_{r_b} = - \text{Log} \left\{ \sum_{i=1}^C [p(b_r|c_i)] [1/C], U_r = 1 \right\} \quad (29)$$

$$\text{Log } k_{r_b} = - \text{Log} \left\{ 1/C \sum_{i=1}^C p(b_r|c_i), U_r = 1 \right\} \quad (30)$$

Similarly,

$$\text{Log } k_{r_w} = - \text{Log} \left[1/C \sum_{i=1}^C p(w_r|c_i), U_r = 0 \right]$$

$$\text{Log } k_{r_w} = - \text{Log} \left\{ 1/C \sum_{i=1}^C [1 - p(b_r|c_i)], U_r = 0 \right\} \quad (31)$$

Observing that if,

$$p(x_1) > p(x_2) > p(x_3), \text{ then} \quad (32)$$

$$[p(x_1) - k] > [p(x_2) - k] > [p(x_3) - k] \quad (33)$$

Hence, the term $\text{Log } p(c)$ may be deleted from the formula for S_c' , and a new quantity performing the same purpose as S_c' may be defined, S_c (unprimed)

$$= \sum_{r=1}^R \left\{ \text{Log } p(b_r|c) - \text{Log } 1/C \sum_{i=1}^C p(b_r|c_i) \right\}, U_r = 1 \quad (34)$$

$$+ \left\{ \text{Log } [1 - p(b_r|c)] - \text{Log } 1/C \sum_{i=1}^C [1 - p(b_r|c_i)] \right\}, U_r = 0.$$

This last equation completely defines the basic comparison routine proposed. The implementation of the equation for the solution of S_c , score of match, may be performed in a plurality of ways. For example, one system that will meet this requirement contains:

1. A device to examine the r 'th cell of the unknown character for the presence of an ink spot. $r=1, 2, 3, \dots, R$.
2. A register to store $\text{Log } \sum_{i=1}^C p(b_r|c_i)$ for each of R cells.
3. A register to store $\text{Log } \sum_{i=1}^C [1-p(b_r|c_i)]$ for each of R cells.
4. A register to store $\text{Log } p(b_r|c)$ for each of R cells for each of C characters.
5. A register to store $\text{Log } [1-p(b_r|c)]$ for each of R cells.
6. A register to store $\text{Log } p(c)$ for each of C possible characters.

To simplify the amount of equipment needed, the data contained in registers 2 and 4 may be combined, and the data contained in registers 3 and 5 may be combined. Hence, the total memory requirement for the computer is $2CR + R + C$ words where C is the number of characters in the set to be distinguished, and R is the number of quantizing cells per character.