II. SIG — The Significance Programs for Testing the Evaluation Output

M. E. Lesk

1. Introduction

Previous evaluation programs of SMART retrieval runs depended largely on data averaged over sets of requests. Comparisons between retrieval algorithms based on such averaged data do not reflect variation among individual questions. For example, if a small superiority should be attributed to one method rather than another, it is unclear whether every request performed slightly better or whether a small majority of the requests performed much better and the remainder of the requests performed much worse. And in the case of very close comparisons, it is not known whether the results are statistically meaningful. To provide such new information, the program SIG has been added to the SMART evaluation system. [1]

The SMART evaluation process now involves three steps: the evaluation of individual runs, the tabulation of various runs for the same requests, and the comparison of runs with statistical checking. The steps are performed (respectively) by the programs EVAL [2], MORVAL [3], and SIG. EVAL identifies the relevant document rank positions in one run, and lists the correlations and rank positions of the relevant documents for each request. MORVAL tabulates information on rank positions for sets of runs. It also provides averaged values of measures and graphs. It does not, however, give any indication of the amount of scatter in the measures that are averaged, nor of the expected error in the final curves. Thus, although comparisons may be readily made from the pages of averaged results, the significance of such comparisons is unknown. To see how reproducible the conclusions are, therefore, the individual request data must be used. Although the study of individual requests and even individual relevant documents is the surest way of evaluating a run, it is a tedious task even with the MORVAL tabulations as an aid.

It is therefore desirable to add to the MORVAL output another form of analysis, in which the necessity for analysis of individual requests is removed. Instead, statistical tests are used to analyze the scatter in the data, and to indicate the probable reliability of the experimental results. In addition, listings of individual request performance are provided in a more convenient form for rough checking. This is performed by the SIG program, working with data cards punched by the previous evaluation programs.

Unfortunately, statistical tests are difficult to perform on a retrieval system, because of the large variation in performance between different requests. For example, in a typical run, the value of the rank recall (over 34 requests) varies from 0.01 to 0.75, and has a standard deviation of about 0.19. This indicates that the expected performance of any given request is an uncertain quantity. To avoid this problem of request variation, the simplest solution is to look only at differences between requests. For two sample runs with standard deviations of the averaged rank recall of 0.19 and 0.22, the standard deviation of the difference of the rank recall between the two methods was only 0.13. The SIG program, therefore, uses differences in performance for all its calculations.

The basic input to SIG consists of the values of four different evaluation measures and the recall-precision curve for each request in each processing system. This makes a set of fourteen numbers describing the performance of the request: the rank recall; the log precision; the normalized recall; the normalized precision; and ten ordinates on the Quasi-Cleverdon recall-precision curve at intervals of 0.1 from 0.1 to 1.0. Each of these numbers is between 0.0 and 1.0 representing the performance of a perfect system and 0.0 representing the performance of a theoretically worse system.

In practice, however, these measures do not all vary over a range of 1. The rank recall, for instance, is usually found to lie between 0.0 and 0.9; while the normalized recall generally ranges from 0.6 to 1.0. Because of the different ranges observed in practice, these measures cannot be combined directly. Instead, the SIG program processes each measure separately and generates fourteen sets of statistical data for individual measures.

There are two basic statistical tests in the SIG program: the _t-test_ and the _sign test_. Each test is computed for each measure, comparing two methods on a set of requests with respect to that measure of performance. Any pair of methods may be compared in this way to yield fourteen t-tests and fourteen sign tests of their relative performance. Each test indicates which method is superior (for that measure) and whether the superiority is statistically meaningful.

2. T-tests

The t-test operates as follows, considering for simplicity only one of the fourteen tests, the one using the rank recall as the performance measure: [4] The value of the rank recall is tabulated for each request on each method, denoted by method A and method B. A difference is then taken for each request, i.e.

Difference in rank recall for request i = rank recall (request i, method A) – rank recall (request i, method B). These differences are then averaged over the request set, and the standard deviation of the average is computed in the normal way. The value of the t-test is then obtained by dividing the average difference in performance by its standard deviation, and multiplying by the square root of the number of requests. This t-test parameter is large if the difference between the two methods is large and has a large degree of certainty. For example, if the difference between the two methods, averaged over 25 requests, is 0.2 with a standard deviation of 0.1, the test value is 10. Conversely, if the differences between methods are small compared to their errors, the t-test value is small (e.g. an average difference of 0.01 ± 0.5 gives a t-test of 0.1). With the aid of a program to generate Student's T-distribution, [5] this t-test value can be converted to the probability that a random variation of the results would give equally good results. That is, if the distribution indicates that the t-test value achieved corresponds to a probability of 0.05, it may be expected that random variations between the systems would produce a difference as great as that observed only one time out of twenty.

The results of the 14 t-tests are then combined by the program to yield one overall significance test for the comparisons of the two methods. In many cases, this simplifies the final results, since one method may be superior to the other on all of the fourteen tests. In other cases, the combined test may be non-significant, because the different methods are not consistently better for one measure than for another. For example, one measure may be better at high recall while another is better at low recall. This is not usually the case, since usually the superior method is superior on all measures.

The tests are cumulated by converting each of the probabilities of the t-test to a chi-square and adding the chi-squares, as described in Fisher. [4] Before this can be done, the "two-tailed" t-test described before is converted to a "one-tailed" test. The previous test measured the probability of a significant difference between the methods, but did not consider the sign of the difference. The cumulated test hypothesizes that one of the methods (A or B) being compared is superior, and tests for the significance of a difference between the methods in the indicated direction. Such a test is called a "one-tailed" test. The determination of the direction of superiority is made by adding up the fourteen differences and using the sign of the sum. Thus, if the sum of all differences between a measure on A and a measure on B is positive, the program hypothesizes that A is superior to B. If the sum of all the differences is negative, it assumes B superior to A. The differences are used only to compute the direction of the test and the actual cumulation is done using the probabilities.

The actual conversion to one-tail probabilities is simply a division by two, unless the specific test in question disagrees with the hypothesis, in which case, the result is subtracted from 1. That is, if the two-tailed test had a probability of 0.03 for A superior to B, and the program is hypothesizing A superior to B, the result is 0.015. If the two-tail test had indicated 0.75 and the difference indicated B superior to A, then the one-tailed value (assuming A was to be greater than B) would be 0.875.

These fourteen one-tail probabilities are now converted to chi-square values as described in Fisher, [4] and the chi-squares are added and converted back to probabilities. The validity of this procedure depends on the independence of the fourteen measures, which is true to a limited extent. Some experiments have been run to determine correlations among the measures. It was found that

a) the rank recall is strongly correlated with the high recall points on the Q-C curve;

b) the log precision is most strongly correlated with the middle (.5, .6) points on the Q-C curve, but that the correlation is not as high as that for the rank recall under a) (.8 instead of .9);

c) the normalized recall is most highly correlated with the 0.3 or 0.4 region of the Quasi-Cleverdon curve, at about a 0.7 level; and

d) the normalized precision is correlated at about a 0.7 level with the center of the recall precision curve and peaks at 0.3 or 0.4, correlating at over 0.8. The points on the curve correlate about 0.9 with adjacent points on the same curve and .3 with distant points.

It would appear, therefore, that the tests do have some degree of independence, and that the entire curve is not predictable from one point.

It is true that the fourteen measures are not entirely independent, and therefore a stricter criterion is applied to the final probability; instead of using 0.05, only comparisons with a final probability of less than 0.0005 are generally accepted as significant. The exact formula for the t-test operations are as follows: Define m_{ijA} to be the value of measure i on request j with method A. i is defined as follows:

i	definition
1	rank recall
2	log precision
3	normalized recall
4	normalized precision
5	Quasi-Cleverdon graph at recall = 0.1
6	Quasi-Cleverdon graph at recall = 0.2
7 to 14	Quasi-Cleverdon graphs at recall = 0.3, ..., 1.0

Then the program computes (assuming n requests)

$$d_{ij} = m_{ijA} - m_{ijB}$$

And the averages $M_{iA} = \frac{1}{n}\sum_{j}^{n} m_{ijA}$, $M_{iB} = \frac{1}{n}\sum_{j}^{n} m_{ijB}$, $D_i = \frac{1}{n}\sum_{j}^{n} d_{ij}$

Also the standard deviation of d_{ij}

$$S.D._i = \sqrt{\sum_{j} (d_{ij} - D_i)^2 / (n-1)}$$

And then the t-test values:

$$T_i = (D_i/SD_i)\sqrt{n}$$

A probability P_i is now obtained from Student's t-distribution with n degrees of freedom. The standard deviation is not actually computed from the formula above, but from the equivalent

$$SD_i = \sqrt{\left(\sum_j d_{ij}^2 - \frac{\left(\sum d_{ij}\right)^2}{n}\right) / (n-1)}$$

These probabilities are now combined. First, they are converted to one-tail probabilities.

Define $S = \text{Sign}\left(\sum_j D_i\right)$.

Then: if Sign $D_i = S$, $P_i = \frac{1}{2}P_i$; if Sign $D_i \neq S$, $P_i = 1 - \frac{1}{2}P_i$.

Each probability is converted to a chi-square and summed

$$\chi^2 = -\sum_{i=1}^{14} - 2\log_e P_i \ .$$

The chi-square is now converted back to probability P which is the total significance test. [7]

3. Sign Tests

The other tests which are performed by the system are sign tests, using the binomial probability distribution. These tests are performed only by comparing the signs of the differences in the performance measures for different methods; no actual computations are performed using the numerical values of the measures. Tables are generated showing how many requests perform better with method A, and how many with method B. The program then computes the probability that if the requests were to perform better on A, or better on B solely for chance reasons, then an imbalance of requests preferring one method or another would occur which is at least as large as the one actually observed. Thus, if this probability is 0.05, it means that 19 times out of 20, a random assignment of requests to the categories "better on A" and "better on B" would have produced a small numerical difference between the populations of the two categories than was actually observed.

This test is cumulated directly. The number of requests preferring method A is summed over all measures, and the number of requests preferring method B is summed over all measures. The same independence problem affects this test. These totals are then subjected to exactly the same sign test.

The details of the sign test are:

Define a tolerance <u>t</u>, which is usually taken as 0.001.
Define n_{ai} = the number of d_{ij} $(j=1,\ldots,n)$ greater than $+t$;
n_{bi} = the number of d_{ij} $(j=1,\ldots,n)$ less than $-t$;
n_{ci} = the number of d_{ij} $(j=1,\ldots,n)$ where $|d_{ij}| < t$.
$(n_{ai}+n_{bi}+n_{ci}=n)$

Now define $n_{vi} = n_{ai}+n_{bi}$ and $n_{wi} = \min\,(n_{ai},n_{bi})$.

Then the desired probability is

$$P^{s}_{i} = 2\sum_{j=1}^{n_{wi}} \;,\; \binom{n_{vi}}{j} 2^{-n_{vi}} = \sum_{j=1}^{n_{wi}} \frac{n_{vi}!}{j!\,(n_{vi}-j)!}\, 2^{-n_{vi}+1} \qquad [7]$$

This is cumulated directly:

$$n_a = \sum n_{ai} \quad \text{and} \quad n_b = \sum n_{bi}$$

$$n_v = n_a + n_b \; ; \; n_w = \min\,(n_a, n_b);$$

$$P^s = \sum_{j=ij!}^{n_w} \frac{n_v!}{(n_v-j)!} \; 2^{-n_v+1}$$

4. Program Operation

SIG is written in FORTRAN II, and runs under the Harvard PMS system on a 7094. It uses two subroutines: DBETA and DGAMMA, which compute the probabilities from Student's T distribution and a chi-square distribution.

The deck set-up to run SIG is as follows:

1. Job card
2. * XEQ
3. Program deck with subroutines
4. * DATA card
5. A card with three pieces of information:
   a. in columns 1-5, right adjusted, and integer specifying the input tape for the cards described under number 6-8. Usually this is 5 (card input);
   b. in columns 6-10, right adjusted, an integer specifying the number of comparisons to be made with this batch of data, called NPAIR;
   c. in columns 11-16, a number with decimal point, specifying the tolerance used in the sign test (in the description under 3, this is t). Usually 0.001.

6. A card with two pieces of information:

    a. in columns 1-5, right adjusted, an integer describing the number of runs in the immediately following data cards. This integer is called IRUN;

    b. in columns 6-10, right adjusted, an integer, specifying the number of requests in the following data deck. (called IREQ) (note: this card need not be on tape 5);

7. The following data deck, this deck, and the card preceding it, are punched by MORVAL and need not be changed. To obtain the deck from MORVAL, merely run MORVAL with the desired evaluation decks as input (to be converted to the input format for SIG) and the specification ORDER 1. This causes MORVAL to punch the desired deck, which can then be used as input to SIG.

    a. IRUN cards, each with an integer in columns 1-5, right adjusted, and the name of a run (twelve characters) in columns 11-22.

    b. IREQ cards, each with an integer in columns 1-5, right adjusted, and the name of a request (twelve characters) in columns 11-22.

    c. IRUN•IREQ cards, with the following information on each:

        i) in columns 1-3, right adjusted, the number of the request (as defined in b.)

        ii) in columns 4-6, right adjusted, the number of the run (as defined in a.)

        iii) in columns 7-12, the value of the rank recall for the specified request with the specified method. This is punched with decimal point (decimal point in column 8) and is the actual value of the rank recall.

        iv) in columns 13-18, the log precision, punched as in iii.

v) in columns 19-24, the normalized recall similarly.

vi) in columns 25-30, the normalized precision similarly.

vii) in columns 31-34, an integer, right adjusted, which when divided by 9999.0 gives the ordinate of the recall precision curve at the abscissa 0.1. No decimal point may be punched in this field.

viii) in columns 35-38,the ordinate of the recall precision curve at abscissa of 0.2, also multiplied by 9999.

ix) in columns 39-42, the point 0.3 similarly; and so on for 0.4 (cols 43-46), etc, until 1.0 (cols 67-70).

8. A card with either:

   a. blanks or zeros in columns 4-6, indicating the end of data decks and causing the program to return to the normal input tape if it is not currently reading it, and look for cards of type 9.

   b. A negative integer in columns 4-6, indicating that the program should look for more data, beginning with a card of type 6. This additional data <u>must</u> have exactly the same IREQ as the previous data and each request must be numbered as before (i.e. cards 7b must define the same request-number correspondence). Since the main system always alphabetizes the requests,this is not a serious restriction. Thus, additional <u>runs</u> may be introduced; but not additional <u>requests</u>. Normally, the additional data will be on the same tape as before; if it is not, the new tape number should be punched in columns 1-3, right adjusted of this card.

9. NPAIR cards (see 5b) each containing a run name in columns 1-12 and another run name in columns 13-24. These must be spelled the same way as the names were spelled on the cards of type 5a. These cards specify the pairs of methods to be compared.

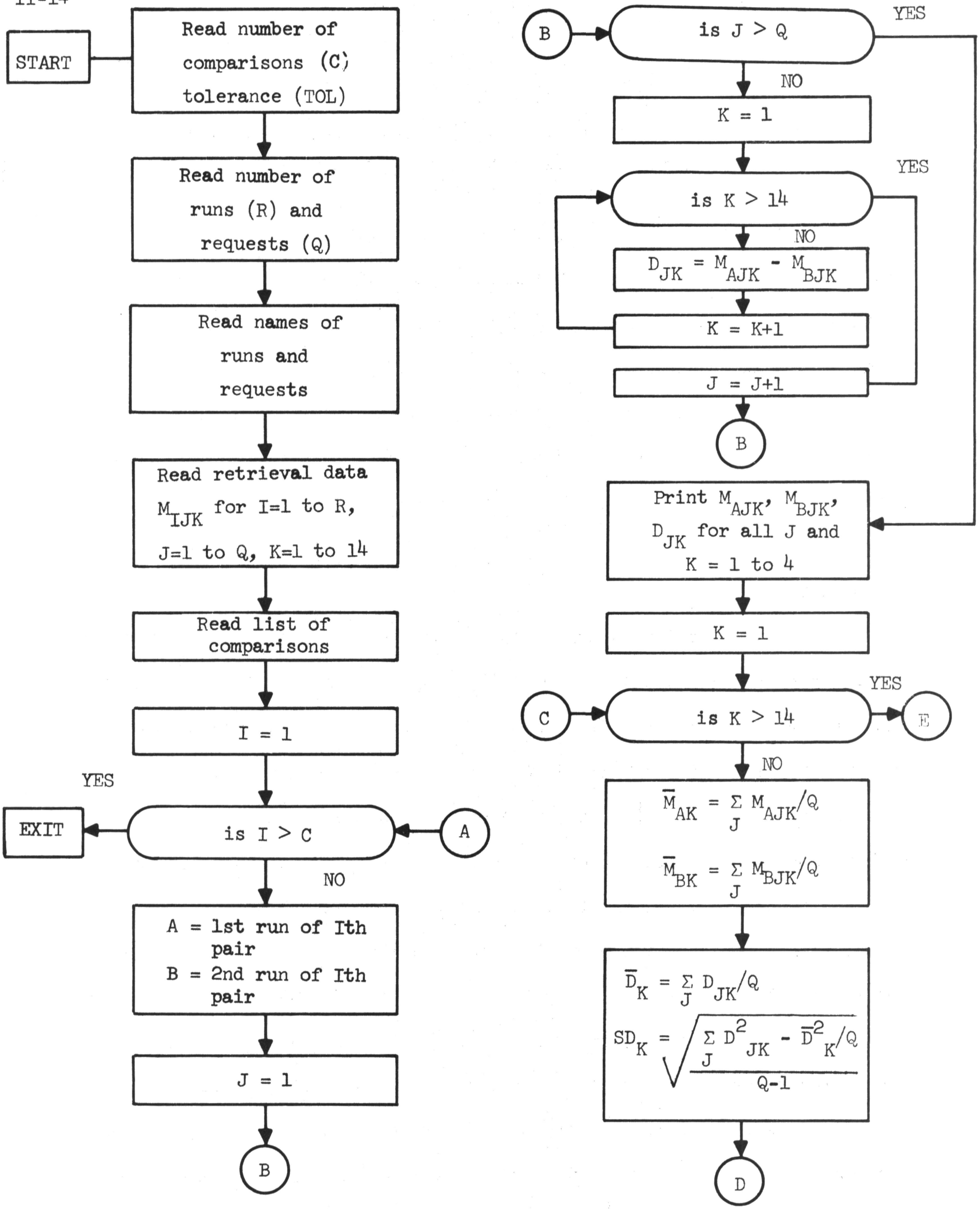
10. The SIG program now proceeds to process the desired comparisons. Each comparison is given the full statistical treatment. Any method may be compared with any other, any number of times.

After finishing all comparisons, the program erases its memory and looks for a new data deck, beginning again with a card of type 5. It continues to read data decks and process them until it runs out of input cards (there is no special ending sentinel).

SIG can perform about 25 comparisons per minute, assuming each comparison to involve about 35 requests. A flow chart of SIG is shown in Fig. 1.

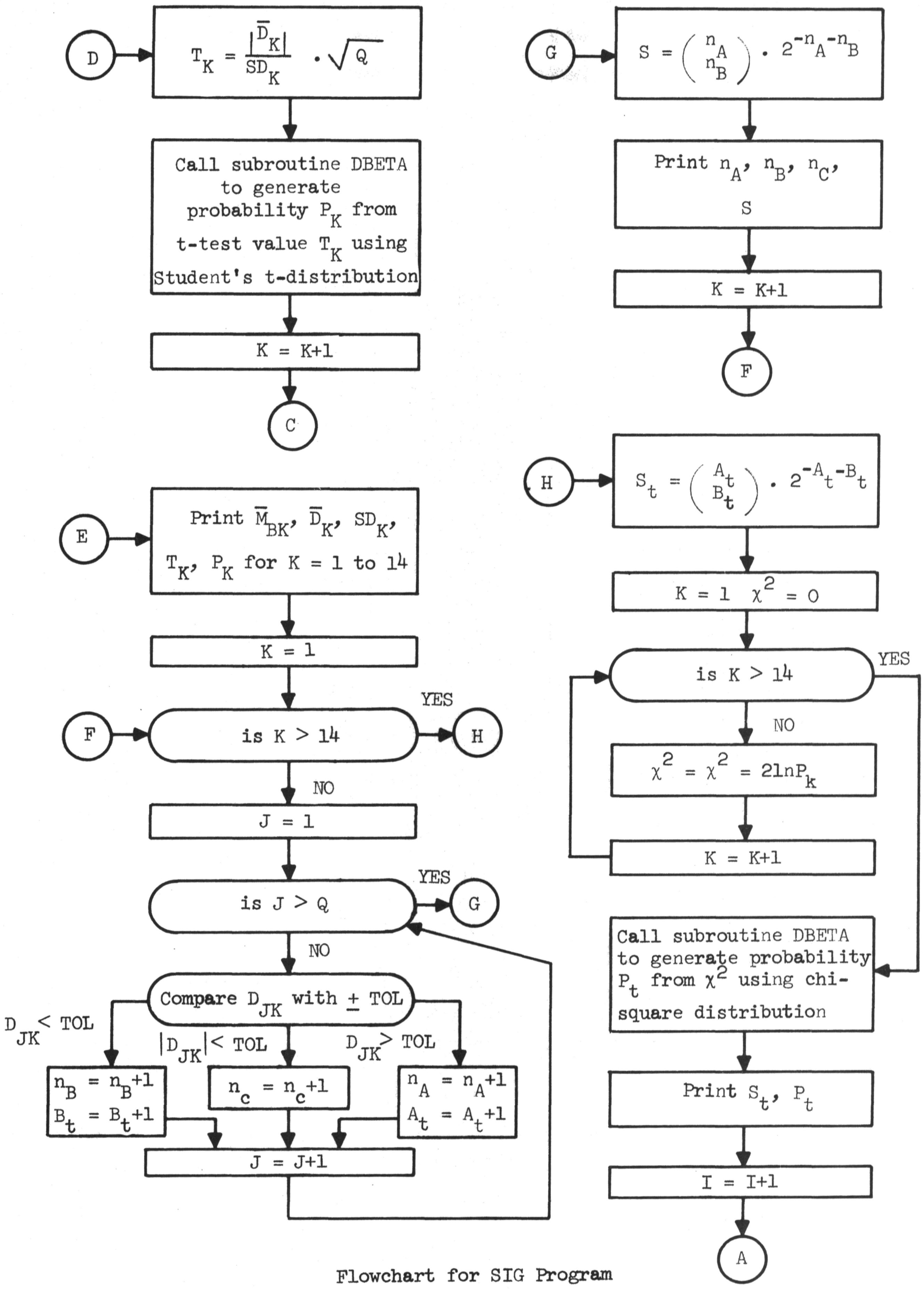
5. Output

For each comparison, SIG produces a page or two of output containing all the statistical data and some of the raw data (the remainder of the raw data is given later). (See Figs. 2-5). This output list begins with a line "COMPARING METHOD A(...) WITH METHOD B (...)" which defines A and B for this comparison. The next section of output has one horizontal line of numbers for each request. Read across from left to right, each line has: the name of the request, the value of the rank recall for this request and method A (method A as defined in the page heading), the value of the rank recall on method B for this request, the difference of the first two numbers (rank recall with method A minus rank recall with method B) and then the same three values for each of the next three measures (log precision, normalized recall, normalized precision). Thus, in the nomenclature used earlier, for the j^{th} horizontal line, the column heading "REQUEST NAME" contains the name of the j^{th} request; the column headed "RANK RECALL" and


START
Read number of comparisons (C) tolerance (TOL)
Read number of runs (R) and requests (Q)
Read names of runs and requests
Read retrieval data M_{IJK} for I=1 to R, J=1 to Q, K=1 to 14
Read list of comparisons
I = 1
is I > C
YES
EXIT
A
NO
A = 1st run of Ith pair
B = 2nd run of Ith pair
J = 1
B
B
is J > Q
YES
NO
K = 1
is K > 14
YES
NO
$D_{JK} = M_{AJK} - M_{BJK}$
K = K+1
J = J+1
B
Print M_{AJK}, M_{BJK}, D_{JK} for all J and K = 1 to 4
K = 1
C
is K > 14
YES
E
NO
$\overline{M}_{AK} = \sum_J M_{AJK}/Q$
$\overline{M}_{BK} = \sum_J M_{BJK}/Q$
$\overline{D}_K = \sum_J D_{JK}/Q$
$SD_K = \sqrt{\dfrac{\sum_J D^2_{JK} - \overline{D}^2_K/Q}{Q-1}}$
D

Flowchart for SIG Program

Fig. 1


D
$T_K = \frac{|\overline{D}_K|}{SD_K} \cdot \sqrt{Q}$
Call subroutine DBETA to generate probability P_K from t-test value T_K using Student's t-distribution
K = K+1
C
G
$S = \binom{n_A}{n_B} \cdot 2^{-n_A-n_B}$
Print n_A, n_B, n_C, S
K = K+1
F
E
Print $\overline{M}_{BK}$, $\overline{D}_K$, SD_K, T_K, P_K for K = 1 to 14
K = 1
F
is K > 14
YES
H
NO
J = 1
is J > Q
YES
G
NO
Compare D_{JK} with ± TOL
D_{JK} < TOL
$|D_{JK}|$ < TOL
D_{JK} > TOL
$n_B = n_B+1$
$B_t = B_t+1$
$n_c = n_c+1$
$n_A = n_A+1$
$A_t = A_t+1$
J = J+1
H
$S_t = \binom{A_t}{B_t} \cdot 2^{-A_t-B_t}$
K = 1 $\chi^2 = 0$
is K > 14
YES
NO
$\chi^2 = \chi^2 = 2\ln P_k$
K = K+1
Call subroutine DBETA to generate probability P_t from χ^2 using chi-square distribution
Print S_t, P_t
I = I+1
A

Flowchart for SIG Program

Fig. 1(c)

COMPARING METHOD A (NULL CONCON) WITH METHOD B (HARRIS THREE)						
Request Name	Rank Recall			Log Precision		
	A	B	Diff	A	B	Diff
Automata Phr	0.5238	0.9649	-0.4411	0.7126	0.9881	-0.2755
Comp Systems	0.0725	0.1228	-0.0503	0.3783	0.4806	-0.1023
Comps-Assemb	0.3714	0.7428	-0.3714	0.8542	0.9453	-0.0911
Core Memory	0.0691	0.1064	-0.0373	0.3157	0.3695	-0.0538
Differntl Eq	0.5298	0.7574	-0.2276	0.8620	0.9219	-0.0599
Error Contrl	0.1460	0.1875	-0.0415	0.5342	0.5972	-0.0630
M10-Counters	0.8182	0.7347	0.0835	0.8682	0.8599	0.0083
M2 Transmit	0.0522	0.0963	-0.0441	0.2819	0.4698	-0.1879
M3-Inform	0.1968	0.3134	-0.1166	0.6300	0.7666	-0.1366
M8-Storage	0.0375	0.2763	-0.2388	0.2670	0.4666	-0.1996
Missile Trak	1.0000	0.7500	0.2500	1.0000	0.6309	0.3691
Morse Code	1.0000	1.0000	0.	1.0000	1.0000	0.
Pattern Recg	1.0000	1.0000	0.	1.0000	1.0000	0.
Random Numbs	0.0517	0.2000	-0.1483	0.1750	0.3408	-0.1658
Solstat Circ	0.2766	0.3402	-0.0636	0.6921	0.7912	-0.0991
Switch Funds	0.3529	0.4444	-0.0915	0.7416	0.8005	-0.0589
Thin Films	0.2157	0.8462	-0.6305	0.6294	0.9242	-0.2948

Computation of Recall and Precision Differences

for

Individual Requests

Fig. 2

COMPARING METHOD A (NULL CONCON) WITH METHOD B (HARRIS THREE)						
Evaluation Measure	A	B	Diff	STD.DEV.	T-Test	Prob.
Rank Recall	0.3950	0.5225	-0.1276	2.07E-01	2.54E 00	0.0219
Log Precision	0.6437	0.7267	-0.0830	0.47E-01	2.33E 00	0.0334
Normalized Recall	0.9233	0.9675	-0.0442	5.35E-02	3.41E 00	0.0036
Normalized Precision	0.7419	0.8639	-0.1219	1.20E-01	4.19E 00	0.0007
Quasi-Cleverdon 0.1	0.7385	0.9735	-0.2351	2.88E-01	3.37E 00	0.0039
" " 0.2	0.6544	0.8973	-0.2428	2.82E-01	3.55E 00	0.0026
" " 0.3	0.5844	0.8245	-0.2401	2.51E-01	3.95E 00	0.0011
" " 0.4	0.5326	0.7551	-0.2226	2.39E-01	3.84E 00	0.0014
" " 0.5	0.5187	0.7146	-0.1959	2.00E-01	4.04E 00	0.0009
" " 0.6	0.5035	0.6499	-0.1464	1.59E-01	3.79E 00	0.0016
" " 0.7	0.4452	0.6012	-0.1561	1.79E-01	3.59E 00	0.0024
" " 0.8	0.4091	0.5514	-0.1423	2.24E-01	2.62E 00	0.0184
" " 0.9	0.3794	0.4973	-0.1179	2.29E-01	2.12E 00	0.0499
" " 1.0	0.3106	0.4118	-0.1012	2.44E-01	1.71E 00	0.1070

T-test Computations for 14 Different Recall and Precision Measures

Fig. 3

SIGN TESTS OF PAIRED COMPARISONS						
RNK REC	LOG PRE	NOR REC	NOR PRE	Q-C 0.1	Q-C 0.2	Q-C 0.3
A B = 2 13 2 0.0074	A B = 2 13 2 0.0074	A B = 2 13 2 0.0074	A B = 2 13 2 0.0074	A B = 0 9 8 0.0039	A B = 0 11 6 0.0010	A B = 1 12 4 0.0034
Q-C 0.4	Q-C 0.5	Q-C 0.6	Q-C 0.7	Q-C 0.8	Q-C 0.9	Q-C 1.0
A B = 1 11 5 0.0063	A B = 0 13 4 0.0002	A B = 3 11 3 0.0574	A B = 2 12 3 0.0129	A B = 3 12 2 0.0352	A B = 4 11 2 0.1185	A B = 4 11 2 0.1185

Sign Test Computations for 14 Different Recall and Precision Measures

Fig. 4

"A" contains m_{1jA} ; the column under RANK RECALL headed B contains m_{ijB} ; the column under RANK RECALL headed DIFF contains d_{1j} . The next three columns contain m_{2jA} , m_{2jB} , and d_{2j} . Note that 2 refers to the log precision, as defined earlier. The remainder of the line contains m_{3jA} , m_{3jB} , d_{3j} , m_{4jA} , m_{4jB} , and d_{4j} , in that order.

The next section of printed output contains 14 horizontal lines, one for each evaluation measure. Each line, reading from left to right, gives the name of the evaluation measure, its average value (over the requests) on method A, its average value on method B, and the average of the difference between this measure on method A and method B for each request. This last figure, the average of the differences, should be equal to the difference of the averages within the truncation error of the printout (± 0.0001). The next column gives the standard deviation of the differences from their average, and the following columns give the number computed for the t-test and the resulting probability (for the two-tailed test). (Fig. 3) Using the previous nomenclature again in the i^{th} line, the column headed "EVALUATION MEASURE" gives the name of the i^{th} evaluation measure; the column headed A gives M_{iA} ; the column headed B gives M_{iB} ; the column headed DIFF gives D_i; the column headed STD. DEV. gives SD_i; the column headed T-Test gives T_i; and the last column gives P_i .

The next section presented is the sign test data (Fig. 4). Each of the 14 measures is listed across the page, and under each measure name there are 3 headings, A, B, and "=". Under each of these three characters is an integer. The integer under "A" is the number of requests whose performance, for the specified measure, was better on method A than on method B by more than the tolerance (n_{ai}). Similarly, the number under "B" is the number of requests which did better on method B by at least the tolerance, n_{bi}. The number under the "=" sign is, of course, the number of requests which had the same performance value on both methods to within the tolerance (n_{ci}). There is also a decimal fraction under each measure which represents the probability of obtaining this large an imbalance between A and B with random assignment of requests to the two preferences (p_i^s). Thus, if the numbers under A and B are 0 and 5 respectively, the fraction will be 0.0625, since the probability of obtaining a 0-5 division of five requests at random is 2/32 or 1/16. If the numbers under A and B respectively are 13 and 14, the fraction will be 1.0000 since the probability of such an imbalance is 1, or certainty (one cannot have less imbalance between 27 items than to divide them 13 and 14).

After the 14 sign tests, the program lists the cumulated tests (Fig. 5). The cumulated t-test values are given first; the combined significance probability is given, and then the chi-square. The combination algorithm tends to ignore insignificant tests and combine significant tests strongly, so that if there are more than a few significant tests, it will most likely give 0.0000 (less than 10^{-4}) as the combined significance number. The value of the chi-square is also given. The combined sign test is straightforward; the numbers under the fourteen distinct sign tests are added and the same computation carried through.

This statistical data is repeated for each comparison. At the end of all the comparison for one data deck, the program prints an appendix giving the Quasi-Cleverdon curves for each request and each method, since this information would not otherwise be available.

```
Combined significance values                            }
  for superiority of B over A                           }  T-test
  is 0.0000 (Total Chi-Square = 1.67E 02)               }
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Total comparisons favoring A =  26                      }
Total comparisons favoring B = 165                      }  Sign-test
Total comparisons not caring =  47                      }
Significance test            =   0.0000                 }
```

Combined Significance Output for 14 Measures
(T-test and Sign-test)

Fig. 5

References

[1] M. E. Lesk, The SMART Automatic Text Processing and Document Retrieval System, Report No. ISR-8 to the National Science Foundation, Section II, Harvard Computation Laboratory, December 1964.

M. E. Lesk, Operating Instructions for the SMART Text Processing and Document Retrieval System, Report No. ISR-11 to the National Science Foundation, Section II, Harvard Computation Laboratory, June 1966.

[2] M. E. Lesk, Evaluation of Retrieval Results in the Extended SMART System, Report No. ISR-9 to the National Science Foundation, Section XVII, Harvard Computation Laboratory, August 1965.

[3] M. E. Lesk, A Program to Evaluate the Structure Search Process (MORVAL), Report No. ISR-9 to the National Science Foundation, Section XVIII, Harvard Computation Laboratory, August 1965.

[4] R. A. Fisher, Statistical Methods for Research Workers, 12th edition, Hafner Publishing Co., New York, 1954, page 119.

[5] We gratefully acknowledge the assistance of Miss Mary Hyde, of the DATA-TEXT project (Professor A. S. Couch, Department of Social Relations, Harvard University) in providing us with programs for the t-distribution and the chi-square distribution.

[6] R. A. Fisher, Statistical Methods for Research Workers, 12th edition, Hafner Publishing Co., New York, 1954, page 99.

[7] R. A. Fisher, Statistical Methods for Research Workers, 12th edition, Hafner Publishing Co., New York, 1954, page 63.