Semi-automatic indexing of archival lists and word-frequencies in such lists

by

Arie ARAD

States Archives - JERUSALEM

Archival indexing problems

Archival institutions absorb large quantities of items which have to be made accessible to the public, the administration and the scientific communities. The disproportion between the demand for detailed finding-aids and the means to prepare such aids for voluminous collections is a common problem, as well as an old one. Arrangement by provenance and the use of "Administrative logic" for retrieval is one, accepted, way of overcomming the problem. For "in-depth" retrieval, however, indexing of archival material has, yet, no substitute.

Conventional indexing is a time-consuming and demanding task. This, together with the scarcity of means and properly trained personnel make many archives lag in indexing and have unsatisfactory retrieval. Since the main part of the indexing and have unsatisfactory retrieval. Since the main part of the indexing process concerns reading, understanding, abstracting and even deducing - it seems that no alternative to human indexing can be forseen in the near future.

Computer techniques combined with human work can ease, somewhat, the problem. The aim of this paper is to describe a method to create provisional, useful, indexes with a relatively low expense of time and manpower. It should be stressed that, apart form other limitations, the method is not offered as a substitute for conventional indexing but as a provisional tool providing some "breathing spell" in the race between the intake of material and its proper processing.

Automatic indexing

A well-known method for automatic-indexing was developed by H.P. Luhn[1], based on a work by May Fallon, in I.B.M. in the years 1957-1958. This method, known as KWIC (Key Words In Context), is based on the idea that a title of an item expresses its contents[2] and that some words in the title are "catchwords", i.e. they may serve as retrieval cues.

The logical process of such automatic indexing is basically simple : each word of the title serves as an index-entry (to which the rest of the title is appended) - unless it is an insignificant word, such as an article or a preposition. To filter out insignificant words a "stop-list" is compiled. Each title word is compared with the stop-list, all words which are not in the stop-list go on to create index-entries while the others are ignored.

The advantages of the method are its simplicity, the provision of context (by having all the titles appended) and the relatively low expense of human effort.

There are several disadvantages to the method - some inherent and some of a technical nature. The main inherent disadvantage of the method is its dependence on the given titles : The resulting index depends, totally, on the quality of the titles; it will describe the item only as well as the titles reflect their contents and in the same phraseology used in the titles. Items with long titles will, usually,

receive more index entries then short titles, regardless of the contents and the importance of the item.

Technical problems concern the creation of compound, or multi-word, entries, inversion and permutation of such entries etc. Another interesting set of problems is the compilation and maintenance of stop-lists and the necessity to by-pass, occasionally, the stop-list.

Archival lists

The main characteristic of archival lists and finding aids is their heterogenity. This paper deals with a certain type of lists known as *transfer-lists* or *deposition-lists*. In many modern administrations archival material is deposited in the archives accompanied by a list of the deposited items, which is prepared by the depositing agency. A list prepared at the provenance has the advantage of being made by people who are acquainted with the material. For the archives it also means to deal with ready-made lists, which saves a considerable amount of labour.

Not all transfer-lists are suitable for automatic processing. Large series of individual-files, some types of case-files and sequential-series-files can not be processed (and in most cases should not be thus processed). If, however, a set of subject files, or in many instances, case files - is listed properly - such lists may serve for automatic indexing.

Unlike books, or articles, whose creation is a one-time process, files are formed by accumulation over a period of time. Titles are given to files at the beginning of the accumulation and indicate the intended, or predicted contents only. The relation between the title of a file and its contents depends on the quality of the filing procedure and on the quality of the filing-scheme. Idealy files should be checked, and their titles modified, if necessary, when transfer-lists are prepared, unfortunately, this is not always the case. It follows that the inherent dependence on the quality of titles imposes stricter limitations on archival material than on scientific papers.

Having noted all those reservations, archivists still encounter many lists that have a quality sufficiently good to justify automatic indexing.

Semi-automatic archival indexing

The author has developed, for use in the Israel State Archives, a set of programmes for semi-automatic indexing of archival-lists -based on the KWIC method. The use of the prefix "semi" deserves an explanation. While, in principle the KWIC automatic indexing should work for file titles as well as for any set of titles, some problems made it necessary to pre-edit the titles manually before processing them automatically.

The first set of problems is the necessity to create, and manipulate, compound key-words. Three elementary forms of combinations were treated : simple combination, inverted combination, and

two-way (permuted) combination.

Simple combination means the stringing together of two or more words, in their order of occurance in the title to form a single key-word. This is often necessary for names of institutions or companies, e.g. "BAYSIDE LAND DEVELOPMENT CO.", for compound geographical names "PETAH TIQVA" or for combinations of both - "PETAH TIQVA MUNICIPALITY". Simple combination may be necessary for subject whose meaning can not be simply deduced from their components e.g. "STATE DOMAIN", "POLICE POST", etc. and also when there is a combination of nouns and adjectives - "FOREIGN NATIONALS".

Since the programme recognizes as a word any set of characters between consecutive spaces (blanks), the simple combination is achieved by filling the space with a punctuation mark - usually, though not always, a hyphen (-), e.g. "FOREIGN-NATIONALS".

Inverted combination is needed for the same reason as a simple combination but when the order of occurance of words in the title is not the desired order for the index. This is often the case with names of persons, e.g. "WINSTON CHURCHILL" which we wish to appear in the index as CHURCHILL.WINSTON. In the example we have the case "KAFR SABT" which should be inverted as "KAFR" is the Arabic word for "Village". The programme recognizes the number sign (#) as a mark for inversion and "KAFR#SABT" will appear in the index as "SABT.KAFR". Another case where inverted combination is called for is the case where the most significant word for retrieval is not the first in the subject phrase, e.g. "PREVENTION-OF#FRAGMENTATION" will appear in the index as : "FRAGMENTATION.PREVENTION-OF".

The two-way, or permuted, combination is a combination of two phrases(1), each of which should serve as a retrieval cue. The programme prepares two such entries with each of the phrases first. The recognized symbol for this is the eaqual sign (=). Thus, "POSTS=TELEGRAPHS" will appear as "POSTS-TELEGRAPHS" and also as "TELEGRAPHS-POSTS". Another example is "DISPOSAL-OF=STATE-DOMAIN" which will appear *also* as : ::STATE-DOMAIN-DISPOSAL-OF".

The programme allows for combinations of - with = and of - with #.

The last feature of the processing of index entries is the possibility to remove prefixes by means of the slash (/), thus "RE/CLASSIFICATION" would appear in the index as : "CLASSIFICATION". This feature, while not vitally important for English is of prime importance to languages such as Hebrew and Arabic where the article and many prepositions are prefixed to the words. The slash can be combined with any permissible combination of the former symbols, as long as it is the first of them.

Combination and inversion of words by itself was found to be insufficient. In principle each word,

or a combination of words, should be checked against the stop-list. In many cases it is necessary to be able to ignore words even when they do not appear in the stop-list, otherwise the stop list would be overloaded with rare words, insignificant names, codes, numbers, etc. Inflated stop-lists are wasteful both in memory volume and processing time. The exclamation mark (!) prefixed to the word makes the programme ignore it and go on to the next word. This, naturally makes it possible to get rid (in the index) of whole phrases or parts of them, e.g. : ! (FILES-144-149), or : ! BLOCK-6382-PARCEL-75 etc.

There is also need to by-pass the stop-list for the opposite reason, that is - to include in the index a word which is significant in a certain case though usually it is not. In our example the word "VILLAGES" is included in the stop-list. If, however, we had a file which deals with some general problems of villages we might wish to include this word in that specific case. The programme recognizes the plus mark (+) prefixed to a word to by-pass the stop-list and include that word in the index.

The plus can be used for a secondary purpose : Since the programme finds the + at an early stage of the processing of each word, and since the task of checking the stop-list is eliminated in that case, a careful use of the plus (e.g. for all relevant place-names or personal-names) would reduce considerably the processing time - although it does not change the results in the index.

To sum-up the programme's features :

- (or any punctuation mark) combines words.
creates an inverted combination of words.
= creates two permutations of a phrase, direct and inverted.
! makes a word be ignored - regardless of stop-list.
+ makes a word be included in the index - regardless of stop-list.

Once the editing symbols have been entered with the text the programme SWITE prepares the index entries which are kept in a separate file. In this file the index entry contains (translated to string) also the dates and so, the sorting produces a chronological arrangement within alphabetically identical entries. The entry contains also an identification of the file description (title, dates, original notation, call-numbers) so that the description can be appended at the output stage.

The last stage of the processing is sorting the index-entries and printing the alphabetical part of the entry together with the complete description.

Compilation and maintenance of the stop-list

It has already been mentioned that stop-lists should be kept as short as possible in order to save volume and reduce processing time. On the other hand - the stop-list should be comprehensive enough to filter out effectively the unnecessary words.

The simplest, and most pragmatic way of compiling and maintaining a stop-list is by trial and error : In the first stage a certain amount of text is processed while the stop-list is empty and therefore all words will appear as index-entries. The indexer has, then, to choose the stop-list words out of these index according to his knowledge, experience and intuition. The text is now enlarged and processed with the initial stop-list and the process is repeated again and again. With each processing of additional text the stop list grows but the number of additional words should decrease with each run.

Although a stop-list may never reach "saturation" it can be held rather small - depending on the uniformity of the processed material. The proper use of the exclamation (!) in this specific programme contributes to keeping the stop-list short(4).

Practice shows that a stop-list established for certain kinds of texts is not necessarily the best list for different kinds of texts. More specifically : Each record group of archival material deserves its own stop-list to have reasonable results of automatic indexing.

For this, and other reasons, it seems desirable to approach the problem of stop-lists from a theoretical point of view. One possible approach is by examination of word occurance frequencies in the indexed material, in our case - in archival transfer-lists.

Word frequencies and the stop-list

The text used for the experimental example (as well as for the earlier examples of indexing) is a list of files of the series L (Lands), at the Chief-Secretary's Office of the Mandate Government of Palestine - Record-Group 2 in the Israel State Archives.

Zipf(5) has formulated a general description of the statistical structure of language : If R - the rank of a word is its place in the ordered set of words in a text as arranged by decreasing frequencies, and F is the frequency of occurance then, for large sample of language the relation : F.R = Constant is a good approximation of the distribution. Zipf explanation of the formula, as well as later explanations(6) are not completely satisfactory and are out of the scope of this paper. All explanations, however, have to do with the greater significance of the rarer words. An interesting fact is that the Zipf formula holds for many languages (including non European languages such as Hebrew(7)). Zipf formula holds also for sub-languages.

In this paper the term "sub-language" is used in a specific sense which calls for an explanation. It is well known that different communities use language differently(8) - according to their different needs. Archival groups, or record groups are products of different administrative agencies - each having its own aims, its own practices and hence - its own sub-language. This sub-language is also a sub-set of what we might call the "administrative language" which is, in itself, a sub language of the language of the community to which the administration belongs. The language of a list of files is even more limited, since file-titles are almost invariably noun-phrases and not statements or sentences.

So, we deal with a very limited sub-language. Despite being that limited even the language of a list of files of a certain record-group conforms with the Zipf distribution formula. The correspondence is apparent even for a very small sample(9).

For us, the more important aspect of the frequency distribution is to find out whether the more frequent words can serve to compile a stop-list, which is what we expect intuitively.

A sample of 2006 words out of the list of files, mentioned in the beginning of this paragraph, and the frequency list of the first 47 most frequent words (ranks 1-47) was examined. Those 47 words cover almost 69 % of the whole text. Out of the 47 words only 5 were judged unfit for the stop-list, the first of the unfit words had the rank 10 and the next had the ranks : 21, 34, 38 and 46. Only one non-stop-list word belonged to the set which covered up to 50% of the text.

It seems, therefore that the answer to the question posed above is positive : The more frequent words can serve to determine, with due care, a basic stop-list. The stop-list which was prepared for the example, in the empirical method described in the former paragraph, contains two words which were not encountered in the first 47 words of the frequency-list. The rest of the words in that stop-list occupy the following ranks in the frequency-list : 1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 14, 17, 20, 26, 27, 43. That means that 12 out of 18 words (67 %) are within the 50 % coverage - see graph 1.

The author has written a simple programme for preparing frequency-lists.

The next question was - what should be the size of the text-sample to be processed in order to provide a satisfactory preliminary stop-list. While, naturally, there is no exact, objective, answer to this - it might be of interest to describe the experiment carried-out, which gives some indication of a reasonably possible answer.

The samples of words from the list of files were prepared in five stages. At each stage a larger sample was processed and each sample contained the former one. Samples were drown from groups of consecutive files, the groups, however, were spread in regular interval throughout the list(10).

In order to estimate how well the sample represents the text in general (and, therefore, how well the most frequent words would serve as a basic stop-list), the following properties were checked :

a) Vocabulary density was defined as the ratio of the number of the different words to the total number of the words in the sample. Graph 2 and table 1 show the relation between sample size and vocabulary density.

b) From the frequency lists the most frequent words, whose combined coverage amounted to 50 % of the text (i.e. 50 % of the total number of occurances of all the words of the sample) were compared. The lists were compared with respect to the *order* of the words between each

consecutive sample. This was done by computing the Spearman rank-order-correlations(11). See table 1.

c) Pearson moment-product correlation between the relative frequencies of the first 18 words (of sample n. 5 - the biggest one) in all samples were computed. See table 1.

A remark about the high value of the pearson correlation between sample 1 and 2 is necessary : This high value is a result of the fact that a number of words were missing in both lists (or missing in one list and had a very low frequency in the other) so that the correlation takes into account 3 pairs of zeroes and 3 pairs (.00 - .01) which makes the correlation higher.

Without going too deeply into the analysis, and comparing the data, it seems quite clear that according to the different tests the frequency list, in its first part (i.e. the most frequent) becomes quite stable, in this specific case, at a sample size of 1500. Therefore a sample of 2000 words seems sufficient to base upon it a stop-list in this case.

This seems to be a surprisingly small sample. Moreover - the first 40 different words, out of a vocabulary of 412 words (less than 10% of the vocabulary), cover 66% of the text (see graph 2). It seems that the explanation for that is the specifity of the sub-language of the archival list. The language of the file-list is limited in subject matter since it is a product of an administrative action of a limited scope. The language is limited grammatically since it consists, almost exclusively, of noun-phrases, therefore it is very poor in verbs which could, in a conventional text, inflate the stop-list.

As a consequence it is recommended that the first stage in the compilation of a stop-list would be a compilation of a frequency list of a suitable sample. After choosing the elementary list from the frequency list one can go on to the next stages in the pragmatical approach discussed above.

General and specific stop-lists

Examining further the stop-list we can distinguish two kinds of words in the list. The first kind, which we may refer to as "general" - are the words we would expect to appear in the stop list regardless of the specific record-group or its subject-matter. This would naturally include the articles and the prepositions. The other, "specific", words arise from the specific subject matter of the processed record group. The division into "general" and "specific" is based on subjective judgement. The general list is general only within the framework of the archives. A word which is a specific stop-list word for one record group may be a relevant index word for another record group.

Table 2 describes the number of general and specific words in the various samples (above the 50% line). For comparison a primitive measure of similarity between lists was adopted : the similarity between two lists was defined as twice the number of the common words divided by the total number of words in the two lists. The data of sample 5 is also compared with the data of sample 1, to show

the overall change.

The measure of similarity will be 1 for two identical lists, it will be 0 for two lists which have no common words. Two lists, which have the same number of words and a half of each is identical will have a similarity of 0.5.

Two phenomena are apparent : the list of general words is more stable throughout the samples, also its overall stability (from first to last sample) is considerably higher. See table 2.

The practical result of the preceding discussion is that in order to compile stop-lists for the indexing of archival lists we should begin by keeping a separate "general-stop-list" which should be based on a sample chosen from a large number of *different* record groups. Then, for each record-group, we should compile a "specific-stop-list". By merging the two lists we get a workable basic list which we can update through the "empirical", or pragmatic process.

The next development should be to make a frequency-list programme, such as more sophisticated by comparing each word with the "general-stop-list" and ignore all "general" words while compiling the frequency list.

Since it seems that in many cases we need relatively small samples - this method has a practical value and not merely a theoretical one.

NOTES

(1) Luhn, H.P. - pioneer of information science -.

(2) Literary or metaphorical titles are not discussed here.

(3) In principle one might require more than a single permutation but experience so far, shows it to be unnecessary or, at least, extremely rare.

(4) In our example only 18 words were necessary (not counting punctuation marks).

(5) Zipf, G.K., Human behaviour and the principle of least effort. Addison Wesley, 1949.

(6) e.g. Mandelbrot, B., On the theory of word frequencies and related markovian models of discourse. In structure of language and its mathematical aspects pp. 190-219.

(7) Irmai, S., Formulae determining word frequencies in Hebrew and other languages. Stencilled draft, Haifa, 1977.

(8) Within the same language is we use the term usually, i.e. English, French etc.

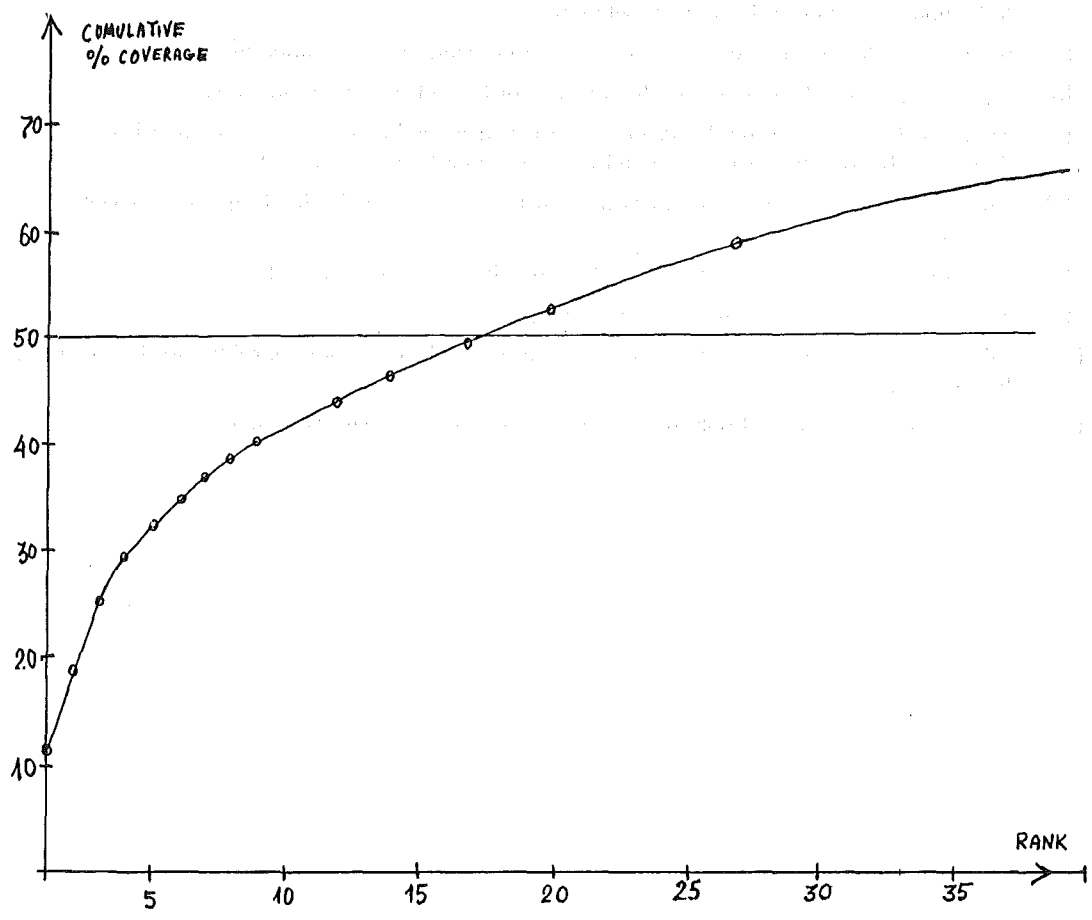
(9) The sample was "cleaned" from personal and place-names etc.

(10) Files processed for the index were chosen from pages which did *not* contribute to the frequency list samples.

(11) Correlation was computed only between words common to each two sample.

Graph 1

Cumulative coverage in percent vs. Rank



Circles denote words actually used in the example stop-list.

Graph 2

Density of vocabulary vs. Sample-size

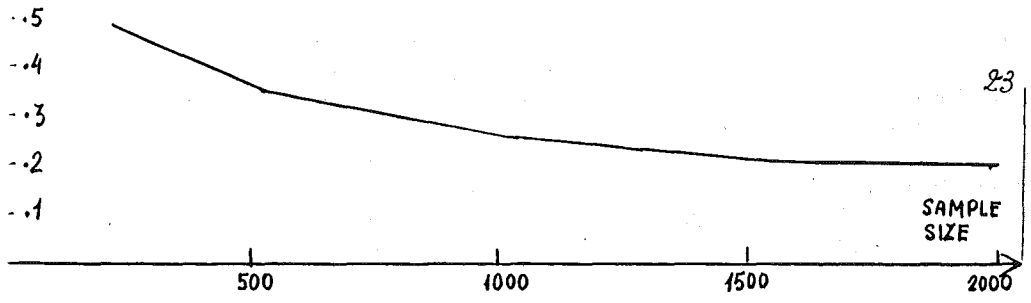


Table 1

General Data and Correlations of the five samples

sample n°	1	2	3	4	5
sample size	221	526	1042	1562	2006
vocabulary size	106	185	276	351	412
vocabulary density	.48	.35	.26	.22	.21
number of words giving 50% coverage	16	18	15	17	18
R_s (Spearman correlation)	.730	.846	.918	.934	
R_p (Pearson correlation)	.972	.961	.994	.998	

Table 2

Specific and General words in the stop-list

sample n°	1	2	3	4	5	1
sample size	221	526	1042	1562	2006	221
n° of general words	9	7	9	8	10	9
n° of specific words	4	5	6	7	7	4
common general words	7	7	8	8	8	
common specific words	3	3	5	5	2	
similarity - general words	.88	.88	.94	.89	.84	
similarity - specific words	.67	.54	.77	.71	.36	