II. A GENERAL DISCUSSION

T. Evslin

1. Introduction

The following is a general description of the logical properties of the SMART system. For operating instructions and specific descriptions of the system operation, see Sec. IV of this report. For a description of specific subroutines and algorithms, see the sections devoted to the appropriate chain link.

SMART is an experimental document indexing, correlating and retrieval system. English language documents are punched on cards and converted by the lookup programs to an internal form, called document vectors, which become part of the permanent document collection. Request vectors are formed in the same way and correlated with the documents. Document vectors which correlate with a request above a user-specified cutoff are taken to be answers to this request. The flexibility and importance of SMART lie in the different ways document and/or request vectors may be mechanically specified, weighted, or changed to provide the kind of retrieval which the user wishes to use or evaluate.

2. The LOOKUP and Phrase Detection

Documents and requests are introduced into the system in natural language form on punched cards. The only special SMART conventions are

those for document identification and for the simulation of punctuation not available on the IBM 026 keypunch. During the lookup, English texts become document vectors. The simplest form of a vector is one in which each English word is represented by a unique internal concept number. Such a simple system uses very little of the information actually contained in a document.

The first step in the sophistication of the lookup is to assign concept numbers to word stems, rather than to whole words. Obviously a document containing the word "mathematics" should correlate to some degree with a document containing the word "mathematical." The semantic value of both words is the same. Therefore, the first step in the SMART lookup is to split words into stems and suffixes. A list of suffixes is provided internally and rules for handling doubled consonants and terminal y's have been included.

The word stems are looked up in a stem dictionary called the thesaurus, and the suffixes are looked up in a suffix list. From the stem dictionary, each word stem is replaced by one or more concept numbers representing semantic values. The suffix lookup produces syntactic codes for each word.

Two types of stem dictionaries are available in the SMART system. One is handmade; the other produced mechanically. A handmade dictionary offers the advantage of allowing synonyms to be properly identified and assigned to the same concept number. It also allows the assignment of multiple concept numbers to word stems with several possible meanings.

However, a mechanically produced, or "null" dictionary, frees the system from the need for an expert lexicographer to produce a thesaurus for each collection in a new field. It permits the rapid preparation of a dictionary of significant concepts in any discipline as soon as the documents are available. A significant concept, for these purposes, is one which occurs in the collection either with a frequency above a user-specified cutoff, or which appears in a top group of most frequent concepts whose size is also determined by the user. Eliminated from the list of significant concepts are those concepts which are so common that their appearance in a document carries no semantic information. Words like "the," "if," and "or" obviously have no semantic value, although they are necessary for analysis. A list of these "common" words is supplied with the system and may be modified by the users. This list is generally conceded to be standard through all disciplines and subjects. (For a discussion of the production of hand dictionaries see Information Storage and Retrieval, Report No. ISR-7, Sec. III.)

A further sophistication of the lookup process to produce even more complete vectors is possible. If the co-occurrence of single words in documents is significant in the correlation process, then certainly the co-occurrence of entire phrases would be expected to be of even greater importance. For this reason, SMART includes two methods of phrase detection. Phrases, like single word stems, are detected in the text; and the concept number identified with each phrase is added to the vector representing the corresponding document.

Statistical phrases are defined as the co-occurrence of concepts in a sentence regardless of relative position or other relationship between

phrase components. Therefore, the statistical phrase finder would find that the sentences "Information retrieval is a new science" and "Information can be gained by the retrieval of the capsule" both contain the phrase "information retrieval." Preliminary experimental evidence shows that statistical phrase searching significantly improves the performance of the SMART system. There is theoretical speculation that the statistical definition of a phrase may be more valuable than a syntactic definition (in which components are syntactically related) for indexing purposes, since the co-occurrence of concepts in a sentence is likely to be significant whether or not the components are syntactically related.

Two kinds of statistical phrase dictionaries are available, which are somewhat analagous to the hand and mechanically produced concept dictionaries. The hand produced dictionary has the advantage that a phrase may be assigned to the same concept number as a synonymous phrase, or word stem. The mechanically produced phrase dictionary, called the cluster dictionary, has the same advantage as the null word stem thesaurus: it can be quickly and effortlessly produced for any collection without the need for a human expert in the topic field. This cluster dictionary is formed by correlating each sentence in the collection with each other sentence, and forming clusters out of those concepts which co-occur with significant frequency in the same sentence. Each cluster is then assigned an arbitrary cluster number. When a cluster, or a hand-defined phrase, is detected in a document or request, its concept number is added to the vector being formed for that request.

The user of the SMART system may request that syntactic phrases be detected. In order for a syntactic phrase to exist, its constituent concepts must occur in a defined grammatical relationship within the sentence. For example, the sentences "Information retrieval is a new science" and "Information may be obtained by the retieval of the capsule" both contain the statistical phrase "information retrieval," but only the first sentence contains "information retrieval" as a syntactic phrase. The sentence "Because the capsule contains top secret information retrieval is vital" does not contain the phrase "information retrieval" in a syntactic sense even though the components occur in adjacent positions. On the other hand, the sentence "Retrieval of information is the business of SMART" does contain the syntactic phrase.

The program for detecting syntactic phrases accurately is a highly complex algorithm and, compared to statistical phrase searching, is quite expensive to run. Basically, it takes those sentences which the statistical phrase finder has identified as containing all the constituents of some statistical phrase and feeds them through the Kuno Syntactic Analyzer (see Mathematical Linguistics and Automatic Translation, NSF-9). The syntactic codes associated with each word and previously detected from the word suffixes (see above) are introduced to the analyzer along with the actual sentence. Its output is a tree of graph of the sentence. Each node contains semantic and syntactic values; each branch a structural relationship. The match routines (described more fully in Sec. XII of this report) search the sentence trees for the occurrence of any one of the subtrees which are

in the phrase dictionary. If a subtree, such as the noun "retrieval" modified by the noun "information," is found to be wholly included in a sentence tree, the concept number associated with the subtree is added to the document vector. The dictionary of subtrees, or syntactic phrases, is prepared by hand.

The concept numbers which make up the document vectors may now represent four different kinds of information: word stems, statistical phrases, clusters, and syntactic phrases. Associated with each concept number is a weight representing originally the number of times the concept was detected in the document. The user, however, may specify that concept numbers from different sources be weighted differently. For example, he may weight statistical phrases 4.7 times higher than word stems, and clusters twice as high. He may decide that concepts which occur in a document title should be weighted several times higher than those in the body. One of the aims of the SMART system is to discover the effect of such weighting on the type and quality of the retrieved documents. In general, weighting phrases highly will result in a smaller number of retrieved documents, and a higher percentage of relevant ones for a given request. Weighting titles heavily tends to approximate a KWIC indexing system and retrieves a relatively higher percentage of irrelevant texts.

3. Further Modifications to the Document Vectors

A. Concept-concept Correlation

A human researcher who consults a library to find documents relevant to a given request would notice correlations beyond simple identity of words

and phrases. Given the request: "I would like information about solar systems," the researcher would, if he had any acquaintance with the field, select articles not only about the solar system, but also texts using the words "planetary" and "stellar." Concept-concept expansion is a method of adding related, although not necessarily synonymous, concepts to a document vector.

In order to perform concept-concept expansion, it is necessary to correlate each concept number against each other one and identify those which are related above a user-specified cutoff. The first step in this process is to invert the document concept matrix. In its original form (see above) it consists of rows called document vectors. Each vector consists of a list of the concept numbers associated with a given document and their weight in that document. In the inverted form each row is a concept vector containing for each concept the list of documents in which it occurs, and its weights in each document.

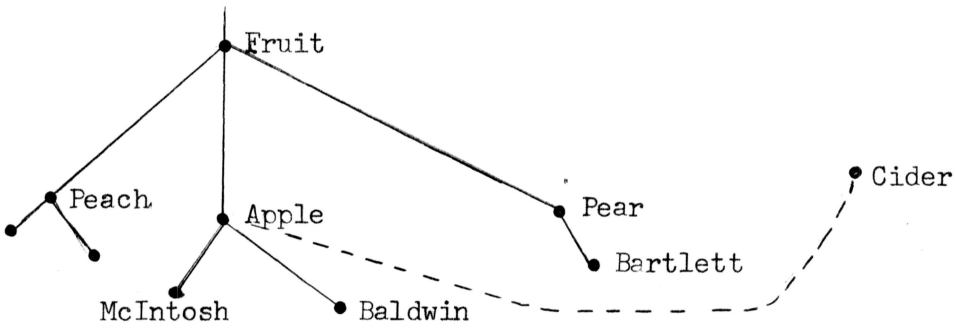
The concept vectors are correlated against each other according to one of two formulas specified by the user (see following for discussion of correlation methods). Concepts appearing in a similar set of documents with similar frequencies will correlate most highly. Concepts which correlate above a user-specified cutoff are assumed to be related. It is possible to iterate the concept-concept correlation as many times as desired, a process which, theoretically, should lead to a complete synonym list.

The user may indicate whether he wishes to have requests, texts, or both expanded by concept-concept association. In either case, the document

vectors to be expanded are read-in, and additions are made from the list of significantly related concepts. For example, if the vector of a certain text contains concept number seven, and concept seven has been found to be related to concept twenty, concept twenty will then be added to the vector. If concept twenty is already present in the vector, its weight will be increased. A user of the SMART system may specify a constant to be used multiplicatively to increase or decrease the weight assigned to concepts which are added by concept-concept expansion (to make them more or less important in determining final request-document correlation).

B. The Hierarchy

Another way of modifying the request and/or text vectors in order to change the nature of the answers received, is to use hierarchical expansion. A hierarchy of concept numbers is included in the SMART system. A segment of a sample hierarchy appears below.



To modify a document containing the word "apple," the user may specify movement in any one of four possible directions. He can expand upwards adding the concept number for "fruit" (the parent of "apple,") to the vector. He can expand laterally, adding "pear" and "peach," which are

the brothers of "apple." Downward expansion to sons adds "Baldwin" and "McIntosh." "Cider" is a cross-reference to "apple," a related concept whose relationship does not lend itself to hierarchical expression. Any combination of the four directions: up, down, sideways, and across, is available. The concept numbers added through these methods may be weighted as the user desires. The user may also choose to have the concepts determined by hierarchical expansion, replace the original vector instead of being appended to it.

4. Correlation

After documents and requests have been looked up in the thesaurus, and have been found, and after the vectors have been expanded and weighted according to the user's object-time specifications, the vectors must be correlated, and documents which are significantly related to the requests must be identified. The word "significant," here as elsewhere in this text, means with a correlation higher than a user-specified cutoff.

Each vector can be thought of as a set of ordered numbers, defining a point on the surface of an N-dimensional sphere, where N is the number of concepts included in the thesaurus. Correlation between two vectors is then a measure of the arc on the surface of the N-sphere which separates them. Two algorithms are available to the user for measuring this arc: cosine and overlap:

$$\text{COSINE}_{ab} = \frac{\sum_{i=1}^{N} a_i b_i}{\sqrt{\sum_{i=1}^{N} a_i^2 \sum_{i=1}^{N} b_i^2}}$$

$$\text{OVERLAP}_{ab} = \frac{\sum \text{Min}(a_i, b_i)}{\text{Min}\left(\sum a_i, \sum b_i\right)}$$

Consider for example the two vectors

$$\bar{a} = 6,0,5,0,0,1,2,$$

and

$$\bar{b} = 2,1,0,2,0,1,1,4.$$

This representation of the vectors implies that for document $\bar{a}$, concept number one has a weight of six, two does not appear, three has a weight of five, and so on. Similarly for document $\bar{b}$, where concept one has a weight of two, concept two a weight of one, and three does not appear. Their correlations by the two given methods are completed as follows:

$$\text{COSINE}_{ab} = \frac{(6-2+0-1+5-0+0-2+0-0+1-1+0-1+2-4)}{\sqrt{(36+0+25+0+0+1+0+4)(4+1+0+4+0+1+1+16)}} = 0.50,$$

$$\text{OVERLAP}_{ab} = \frac{2+0+0+0+0+1+0+2}{\text{Min}(11,14)} = 0.45.$$

These correlation methods have the advantage that all correlation values fall between one and zero. This restricted range is helpful to the user

who must pick a cutoff above which all correlations will be declared relevant.

The same two equations used for request-document correlation are available for the other two correlations performed by SMART: concept-concept correlation (see above) and document-document correlation (see below).

At this point a list of requests is available, and correlations of each document in the collection with each of these requests has been produced. These lists may now be thought of as a matrix, where each row is a request and each column is a document. At the intersection of each row and column is the correlation of a given document with a given request. The user may now print out the documents whose correlation with each request lies above the specific cutoff, or he may perform further operations on the request-document matrix.

5. Document-Document Correlation to Expand the Answer List

If a research worker were performing a document search, he might search his library card catalogue for all the sources listed under his primary topic. He might then go a step further, and search under topics suggested by his primary reading, or listed in the bibliographies of his original sources. Using SMART it is possible first to find a list of documents significantly related to a given request, then to go further and add to this list documents related to the original documents. This process is called document-document correlation. It takes place following request-document correlation.

Each document in the collection is correlated with each other document according to the algorithms explained above. A list of documents whose correlation is larger than a user-specified cutoff with each other document is then prepared. Using this list, a document is added to the answer vector for a given request, if it is significantly related to some other document which is already an answer to that request.

6. Evaluation

Since SMART is an experimental information retrieval system, it lends itself to an evaluation procedure which tests the completeness and relevance of the answers produced by each of the analysis procedures. In order to do this it is necessary to compare the list of request-document correlation received as answers to a search request with a list of anticipated document ranks prepared by hand. Several algorithms are available and a user can easily add his own.

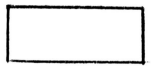
The two basic evaluation measures for an information retrieval system are "precision" and "recall" respectively. Recall is defined as the number of relevant documents retrieved over the total number of relevant documents in the collection. Precision is defined as the number of relevant documents retrieved over the total number of documents retrieved. These two measures tend to vary inversely for a given information retrieval method. For example, if the request-document cutoff is lowered, recall will deteriorate as more irrelevant documents are returned. Basically, a combined recall and precision measure must be used to evaluate the relative effectiveness of different methods of information retrieval. The evaluation

system is described fully in Sec. XVII of this report.

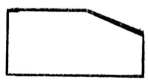
7. Graphical System Summary

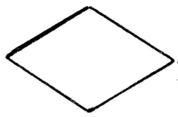
The flowchart which follows is in no sense a system flowchart. It does, however, explain the basic logic, as well as some of the logically important input and output process. A formal flowchart appears in the next part.

Legend:


action always performed by system

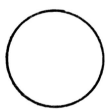

optional function of system

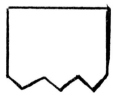

card input via magnetic tape at object time, or card output

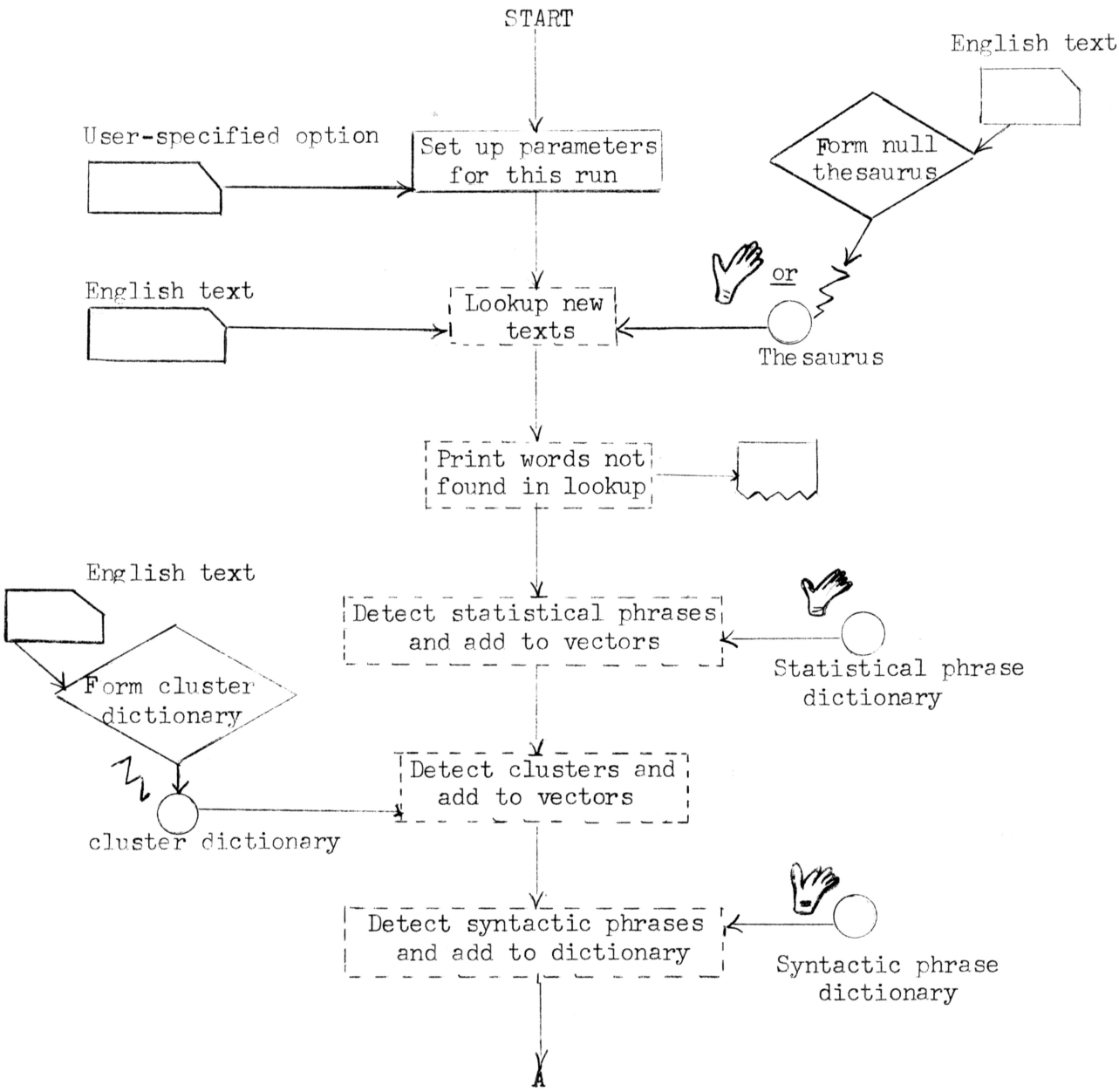

preprocess function


prepared by hand


prepared by machine

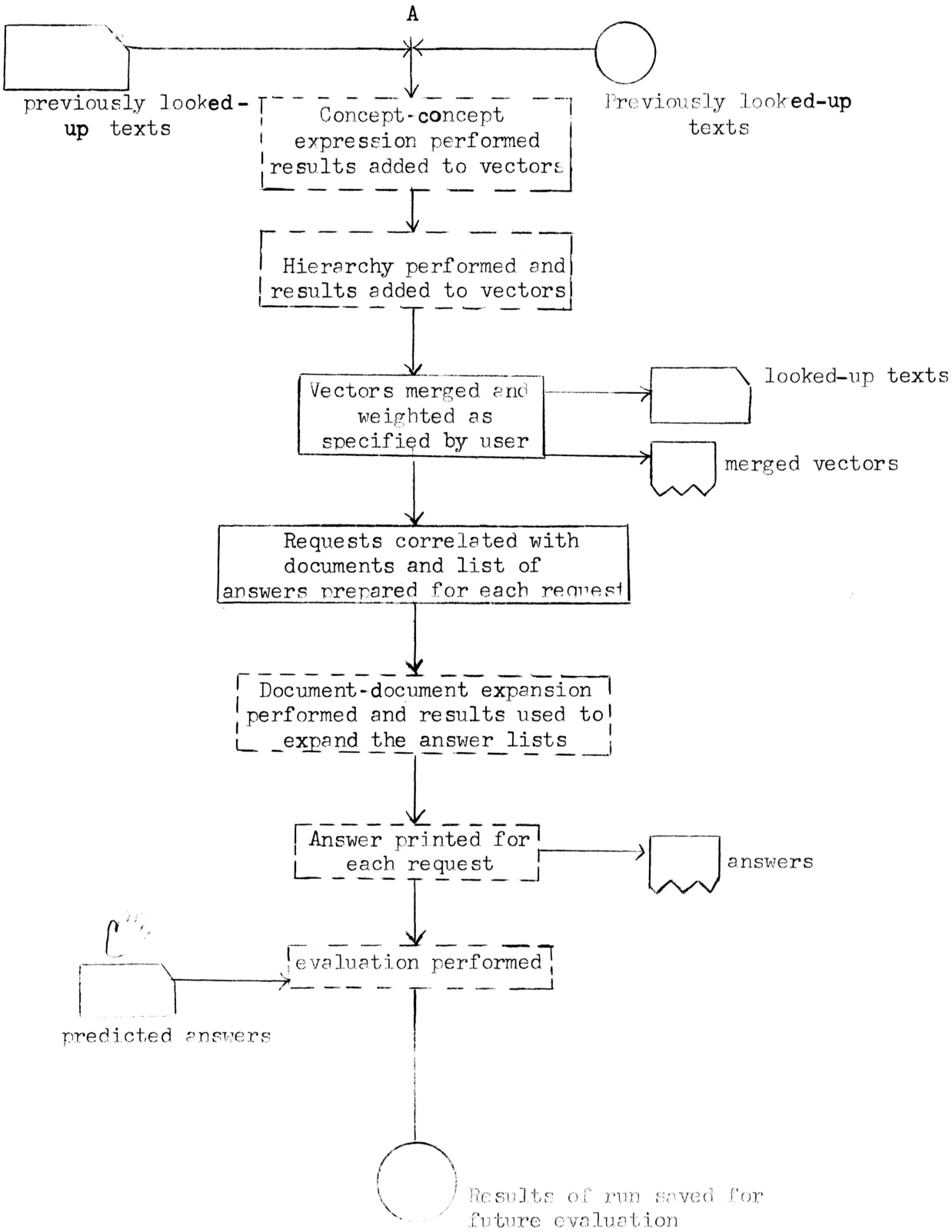

tape input at object


printed output



Smart Logical Functions

Flowchart 1



Flowchart 1 (continued)