	A	B	C	D	E	F	G	H	I	J	K	L	M
1	x		x	x			x		x	x	x		x
2		x	x		x	x	x		x		x	x	
3	x	x	x	x		x			x	x	x		x
4	x		x				x	x			x	x	
5		x	x		x	x	x		x		x	x	x
6		x	x	x		x	x				x	x	x

Figure 7.5
Term-document matrix.

The first step in the algorithm is to translate the marked documents into the descriptor language using the same method of indexing that was used for the entire collection of documents in the IR system. Obviously, this is only needed when these documents are not already present in the system. Otherwise, we can use the existing document profiles. The next step is to create a term-document matrix (Lancaster, 1968). When creating such a matrix we use the marked documents. An example of a term matrix is given in Figure 7.5.

The column headings of this matrix come from the set that is the union of the sets of all descriptors appearing in the document profiles of the marked documents. Such a set will be called a *relevant neighborhood*. In our example, the set {A, B, C, . . . , M} is the relevant neighborhood for the search request represented by the documents 1, 2, . . . , 6. The relevant neighborhood is used to construct a query formulation. To decide which descriptors are going to be used and how to combine them will require formal criteria. The choice of these criteria is very important because they will determine the quality of the search.

First, we will determine the importance of each descriptor from the relevant neighborhood for the search of documents similar to the marked set. Actually, to determine the importance of a descriptor we use the same criterion that was introduced in 1970 in developing the M-algorithm. However, because we only mentioned it earlier without providing a detailed explanation, we would like to do so now.

When analyzing the term-document matrix in Figure 7.5, it is clear that the occurrence frequency of each descriptor from the relevant neighborhood varies from 1 to 6. For example, descriptors C and K appear in every pertinent document, whereas descriptor H appears only once. It would seem that we can conclude that descriptors C and K are more characteristic for the marked set under consideration (and hence for a search request) than is descriptor H. But that would be true only if all descriptors from a dictionary have the same frequency of occurrence in the entire collection of documents (more precisely, in

the document profiles in the collection). But it is well known that such is not the case. Because we are interested in the importance of the term from the search point of view (and not its semantical importance in a search request), we use the following assumption: The most important (essential or characteristic) descriptors are the descriptors whose occurrence frequency in the marked set is maximal and in the entire collection is minimal.

The following formula (already known to us) realizes this assumption.

$$\Psi_i = \frac{r_i \cdot N}{n \cdot R_i}.$$

The value of Ψ_i determines the importance of the corresponding descriptor in the search process, and the descriptors that are more important have larger values for their Ψ_i. This is illustrated by the following example.

If in a collection of 1000 documents (this collection includes the marked set of six documents) the descriptor D (Figure 7.5) occurs 4 times and in our matrix the same descriptor occurs 3 times (in 3 pertinent documents out of 6), then it is reasonable to assume that the remaining document from the collection containing descriptor D is "similar" to our marked documents. In other words the system will consider it as likely to be relevant. The value Ψ_D for descriptor D is

$$\Psi_D = \frac{r_D \cdot N}{n \cdot R_D} = \frac{3 \cdot 1000}{6 \cdot 4} = 125.$$

On the other hand, if the descriptor C (which appears in every document in the marked set—i.e., it shows up six times) appears in 500 documents in the collection of 1000 documents, then is unlikely that the hundreds of documents that contain the descriptor C will be relevant. The value of Ψ_C for the descriptor C is 2.

The examples given show that when a descriptor weight (value Ψ) is greater than some predetermined value, then this descriptor can be used during search without combining it with other descriptors using the operator AND.

So the algorithm using the suggested criterion computes the "importance"—from the point of view of the search—of each descriptor in the relevant neighborhood. Now we are ready to discuss the process of query formulation construction in Boolean form. We mentioned earlier that a query formulation is convenient to represent in disjunctive normal form. The algorithm's goal is to select the best subrequests from the point of view of the search, regardless of how many descriptors are contained in each subrequest.

The construction of a query formulation begins with determining subrequests consisting of one descriptor. A descriptor from the relevant neighborhood will constitute a separate subrequest if the value Ψ_i for this descriptor (see the preceding definition of Ψ_i) is greater than a predetermined bound L, that is, $L < \Psi_i$. After these descriptors are determined by the algorithm, they become