least valuable descriptor (from the remaining descriptors) is subtracted, then the descriptor earlier subtracted and the descriptor subtracted again make up a pair combined by the AND operator. For this pair, the number of documents that it can find (P_{ij}) is calculated, and this pair is added to the remaining single descriptors for a new calculation of estret. Calculation is performed by adding the occurrence frequencies' descriptors plus the number of documents that can be retrieved by the added pair. Then this is all repeated unless the calculated number of documents (estret) becomes equal to $T/2 \leq$ estret $\leq T$. In this case, all the remaining descriptors and all the added pairs are combined by the operator OR and the query formulation is finally considered constructed.

Note that whenever estret $> T$ and there are no more single descriptors to be subtracted, it is the pair with the least weight that is subtracted. If this process is repeated, then the pairs subtracted are used to form triples, which are added to the remaining pairs for a new calculation. In this case, with $T/2 \leq$ estret $\leq T$, it is only the pairs (remaining) and triples that are combined by the OR operator into the final query formulation. This means that the query formulation consists only for pairs, triples, and so forth.

Experimentally, the algorithm was tested in the IR systems MEDLARS and INSPEC (Salton, Buckley, & Fox, 1983). During the experiment, results of the search performed on the basis of automatically constructed query formulations were compared with results obtained with query formulations constructed manually. In MEDLARS, 30 queries were used in a collection of 1033 documents in biomedicine. Five different versions of the search were tested; that is, five different values of T were used. In other words, in the first search (version), the 20 best documents were searched for all 30 queries; in the second search, 30 documents were searched, and then 50, 100, and 200 documents. In three versions out of five, the results of algorithm operation proved better as compared to the manual query formulation. In the INSPEC system (12,684 documents and 77 queries), four versions of search were used, that is, T was 20, 30, 50, and 100 documents, respectively. In all four cases, in this system the results of algorithm operation proved worse than those for manual operation.

We have considered the basic approaches to automatic query indexing. In so doing, we discussed, within the framework of each approach, the application of descriptors when using a more traditional form of search request as well as a marked set of documents. But where did descriptors originate? Practically all authors imply (if this is not explicitly discussed) that before the algorithm for the automatic construction of query formulations is used, the search request, formulated in any form, must be indexed with the aid of a document indexing algorithm. So the use of an automatic document indexing algorithm, for example, the one described in Chapter 6, in all cases must be a preliminary step before using an automatic query indexing algorithm—a step that makes it possible to determine the initial ("working") set of descriptors for each query. To put it differently, unordered sets of descriptors, obtained using an automatic

document indexing algorithm, provide building blocks for constructing query formulations.

The main disadvantage of all proposed query indexing algorithms is the lack of any explanations for the use of thresholds; that is, apart from their selection not being automatic it is not even substantiated. This state of affairs exists despite the fact that the choice of thresholds substantially affects the final results of operation for both the algorithm and the system as a whole. The important thresholds include the number of zones and their ranges in the algorithms of the early 1970s (Frants, et al., 1970; Voiskunskii & Frants, 1971; 1974), as well as in the algorithm set forth by Dillon et al. (Dillon & Desper, 1980; Dillon, et al., 1983), the values of T, M, q-count, and excessive frequency in the algorithm set forth by Salton et al. (Salton, Buckley, et al., 1983; Salton, Fox, et al., 1983; Salton, et al., 1985). This is why when developing an algorithm we tried to concentrate on finding a way to eliminate this disadvantage. By the mid-1980s, we successfully found a simple and effective enough method to automatically choose thresholds that made it feasible to create an advanced algorithm for the automatic construction of query formulations in disjunctive normal forms. It was successfully tested in several experiments, and the algorithm was published in 1991 (Frants & Shapiro, 1991). We present it in the following section.

7.4 Algorithm for the Automatic Construction of Query Formulations in Boolean Form

It is easy to imagine a situation when in some functioning IR system the query formulation is constructed by the system experts from the user's search request given in the form of a marked set of documents. After reading these documents, the expert forms his or her interpretation of the user's information need (POIN) and constructs the query formulation using his or her own knowledge and experience. It is clear that if we could replace a human expert with an automatic expert (i.e., a computer program), which after "seeing" the marked documents would find all similar documents, our problem of creating an automatic expert would be solved. So our algorithm should allow the user to "show" the computer the pertinent documents and provide the user with an output consisting of similar documents. The suggested algorithm will construct the query formulation, which will be used to search for the appropriate documents.

We want to remind the reader that the query formulation constructed by the algorithm has to be in a Boolean form (specifically, disjunctive normal form). This means that we need to solve two problems. First, we must find the descriptors that will be used for the search, and, second, we must combine the descriptors using the logical operators AND or OR.