order to obtain the best output from the system, we need to enforce certain requirements (on the value of the parameter) on another indexing process. This means, in turn, that not only the indexing results but also the very processes of indexing documents and queries are interrelated. We also know the parameter responsible for this interrelation, namely, the average size of descriptor sets obtained as a result of the indexing processes. Consider now which process should be adjusted to another.

From general considerations, the result of the document indexing (an unordered set of descriptors) is simpler than that of the query indexing (a set of descriptor sets represented in the normal disjunctive form). We are, however, interested in the very processes of obtaining the mentioned results. Which process is simpler and which is more complex depends on the principles and implementation approaches that the developers of concrete systems have accepted. This means that in one system the query formulation construction could be more complex, whereas the document profile construction could be more complex in another system. Because in this case we consider Boolean IR systems using manual methods of the query formulation construction (the majority of real systems) and in many systems query formulations are constructed by users, the developers of such systems try to adjust the document indexing process to the real value of the previously mentioned parameter (the number of descriptors combined by operand AND) of the query formulation. The manual construction of the query formulation normally provides two or three descriptors unified by the logical AND, sometimes three or four descriptors, and very seldom five or more. This parameter is taken into account by the system developers (in most cases intuitively) and determines the accepted average number of terms in the document profile. Let us illustrate real values of this parameter by the examples of systems we have worked with. For instance, the average length of sets included in the query formulation of IR systems providing services for the users in the area of information science is 2.7 descriptors, whereas the average size of the document profile is 6.3 descriptors. The same values for the IR system intended for the computer science area are 3.4 descriptors for the query formulation and 7.4 descriptors for the document profile. Of course, we do not consider these values optimal, but they illustrate the order of values used in functioning IR systems. In any case, the parameter values used in practice are quite close to those given here, although they are chosen empirically. The authors are not aware of any strict methods for calculating these values, even for the size of the document profile.

Let us return to automatic indexing. It is worth noting that, according to some researchers, the average size of the document profile is about 30 descriptors, even when only abstracts are indexed. For instance, when we performed automatic indexing of 1276 abstracts (for the nitrogen industry) with an average of 184 words and using a descriptor dictionary with as many as 979 descriptors, the resulting document profiles contained, on average, 34.8 descriptors. Now we can understand why many IR system developers consider document profiles obtained by means of automatic indexing too large. The point is that such document profiles do not match existing query formulations and introduce a certain imbalance into the system, breaking certain harmony that developers intuitively find. Moreover, even in trying to adapt to this type of document indexing, it is not so easy to change the existing manual methods of the query formulation construction. This is particularly true when we deal with users. It is unrealistic to ask them to provide query formulations consisting of, say, 11 descriptors combined by operand AND. This is the main reason that automatic document indexing has not found a wide application in functioning systems. However, if the query indexing is also performed automatically and if the query set sizes can be easily changed (i.e., if the system might be easily adjusted), then the automatic document indexing algorithm will be completely vindicated and will be attractive for all systems where documents are indexed. Such algorithms have been developed and we consider them in Chapter 7, which deals with the automatic indexing of queries.

When considering the very problem of indexing, we have noted that there are systems in which document texts are considered document profiles. Consequently, no indexing is performed in such systems. Note, however, that in such systems documents are represented by abstracts. In some cases stop words are removed from the texts of abstracts, which virtually constitutes a simplified form of indexing. Therefore, at present the automatic indexing of documents is still a rather important process (see, for example, Fidel, 1994; Milstead, 1994; and Soergel, 1994) that, together with the automatic indexing of queries, can substantially improve retrieval results.

6.8

Conclusion

This chapter described the first steps in the construction of a fully automated IR system. The basis of any algorithms for the automatic translation of a document from a natural language into IRL, or the automatic indexing of documents, is the well-known idea of word-for-word translation that was borrowed from a branch of artificial intelligence called "machine translation." Despite the simplicity of the idea, its practical realization faces many obstacles connected with the properties of natural language. Different approaches to dealing with these obstacles, such as stemming algorithms and stop lists, are analyzed in this chapter and, as an example, one of the possible algorithms (which is used in the functioning system) is described.

The different methods for constructing document profiles described in this chapter give an idea of how to develop methods for automatic indexing in the future. One of the indexing approaches emphasized—free text search—was