might increase proportionally to the number of versions. Hence, we assume that only one version of document indexing is used, although the following discussion could be extended to multiversion indexing.

It is more practical to use the multiversion approach in constructing query formulations. There are several reasons why this is the case. First, as a rule, the collection of search requests in the system is many times smaller than the collection of documents. Second, the requests in many cases are shorter than the documents. Third, it is usually not necessary to keep the query formulations in the memory of the system.

Thus for creating an IR system capable of optimal search we will require the presence in this system of some set of algorithms for constructing query formulations. Clearly it must also contain a mechanism that permits a choice of the best available algorithm for each concrete request. To develop such a mechanism it is necessary to know how to evaluate the algorithms effectively (with a reasonable amount of effort) in order to determine which of them leads to the best search results for a given search request. Thus creation of a method of evaluating a search is a necessary condition for the creation of an IR system realizing optimal search. Next we describe a criterion for evaluating different alternatives.

9.8 Criterion for Selection of the Best System's State

In choosing the best among available system's states we will consider the outputs obtained by different query formulations. Let us assume that, on the basis of one search request, we construct a set of several query formulations (using different algorithms and/or parameters of these algorithms) and obtain a corresponding set of outputs as the result of the search in the same collection of documents. For each output, n will denote the number of documents in the output and r will denote the number of pertinent documents in the output. Then the best query formulation will be the one that provides the output with the highest value of r^2/n. Clearly, we need a justification for this criterion. The detailed description of this criterion and its justification was given in Voiskunskii (1982; 1987). Moreover, it will be discussed in great detail in Chapter 10. However, for the benefit of the reader we briefly discuss it now.

In comparing the quality of query formulations we will consider the quality of their corresponding outputs; that is, higher quality output indicates better query formulation. The question of designing and choosing characteristics to describe the quality of the output has been studied extensively (Bollmann, 1978; Cleverdon, 1970; Cooper, 1973; Kraft & Bookstein, 1978; Lancaster, 1979; Raghavan, Bollmann, & Jung, 1989; van Rijsbergen, 1979; Sparck-Jones, 1978). In these studies special attention was paid to characteristics that were the most convenient from a practical point of view. Such characteristics are typically

based on standard measures of output quality: recall, denoted by R, and precision, denoted by P. For example, one of the commonly used characteristics is $R + P$. Larger values of $R + P$ indicate better output. This and other characteristics described in the literature require the computation of the number of pertinent documents in the entire collection of documents, denoted by c. For example, $R + P = r/c + r/n$.

The use in a search process of any characteristic that would require the user to determine all the pertinent documents in the entire collection of documents for every user's request is completely unacceptable because it is equivalent to the user conducting a manual search in the collection. Therefore, in developing an automatic method for determining the system's best state (on the basis of the best output) it is necessary to find a characteristic that would not require such a time-consuming operation as finding pertinent documents in a large collection of documents. Such characteristic, $\sqrt{R \cdot P}$, was proposed in 1982 (Voiskunskii, 1982). This characteristic allows one to compare the quality of outputs without computing c. The use of this characteristic seems to be justifiable because the function $\sqrt{R \cdot P}$ is monotonic in both R and P. However, it is possible to provide another argument that makes the use of $\sqrt{R \cdot P}$ even more desirable.

Let us assume that for a given search request it is known which documents in the collection are pertinent, and there is an output obtained as the result of the search using this request (more precisely, a corresponding query formulation). Let R and P be the recall and precision levels for this output. Now consider two vectors:

$$K = (k_1, k_2, \ldots, k_N) \text{ and } V = (v_1, v_2, \ldots, v_N),$$

where N is the number of documents in the collection; $k_i = 1$ if the i-th document in the collection is pertinent and $k_i = 0$ otherwise; and $v_i = 1$ if the i-th document is found during the search and $v_i = 0$ otherwise. Notice that these vectors represent the results of evaluating the collection of documents: K represents the user's evaluation of the documents' relevance to his or her information need, and V represents the system's evaluation of documents' relevance to the user's search request. Clearly, in the case of ideal output, vectors K and V would coincide. In most cases, however, the two vectors are going to be different and the quality of the output is determined by how "close" K and V are. The natural measure of "closeness" between two vectors, which is used in many fields (including information retrieval), is the cosine of the angle between the two vectors. We will use this measure to evaluate the quality of the output.

Let ϕ be the angle between vectors K and V defined in a standard way. We will now show that $\cos \phi = \sqrt{R \cdot P}$. The number of 1s in vector K is equal to c (number of pertinent documents), hence,

$$\sum_{i=1}^{N} (k_i)^2 = c.$$