tion one succeeds in determining the frequency or relative frequency of occurrence of any set of these descriptors in the same collection, then the algorithm constructed on the basis of the preceding proposition will be sufficiently effective. At any rate, the attempts made in the early 1970s to calculate the probability of the joint occurrence of descriptors in the collection did not yield satisfactory results; the attempts were based on the occurrence frequency of individual descriptors and the joint probability was computed using known formulas for calculating probabilities for independent events. (Linguists are well aware that the use of terms in the text is not independent.) Therefore, in constructing the algorithm we proceeded using the following assumption.

The set of descriptors may form a subrequest only if the value of Ψ_i for every descriptor of that set falls within a certain interval of values and if the set consists of the required number of descriptors. The required number of descriptors included in the set strictly depends on the lower bound of that interval of values within which the values of Ψ_i have fallen (the interval chosen is such that its lower bound is the maximum of what is possible for this set of descriptors), and the frequency of the occurrence of the set in the documents of the marked set is not less than a certain value.

The algorithm created on the basis of this proposition proved more effective than the M-algorithm. Dillon et al. used a very similar approach in two papers published in the early 1980s (Dillon & Desper, 1980; Dillon, Ulmsmeider, & Desper, 1983). They also made use of the marked set as information about the user's POIN, and the descriptor's significance for the search is calculated on the basis of its occurrence frequency in the marked set and in the entire collection. The descriptors for which the calculated value exceeds a certain established threshold are used as independent subrequests and those that did not reach the threshold are partitioned into zones and are used to form subrequests of two, three, and more descriptors, depending on the number of zones and the values chosen for these zones. Still, the algorithm proposed by Dillon and colleagues has a few substantial distinctions.

Like Rickman, Dillon and colleagues considered the automatic construction of query formulations only in the process of feedback, and in a marked set they utilized all user-assessed documents of the previous output (both pertinent and nonpertinent to the user's POIN). On the one hand, these nonpertinent documents are used for determining descriptors included in the query formulation with the NOT (AND NOT) operator; on the other hand (which is the most interesting point), they are used for calculating the descriptors' significance for the search. This is done as follows. First the descriptor's significance for the search (in our opinion, the authors aptly call it "prevalence") is calculated using only the pertinent documents (positive prevalence), and then the "insignificance" of the same descriptor is calculated using the nonpertinent documents (negative prevalence). Then the "genuine" significance (simply prevalence) is calculated for the given descriptor by subtracting negative prevalence from positive prevalence. This, naturally, is only a general idea for calculating the descriptor's prevalence in the search. We now show more explicitly how the authors practically calculate prevalence. For this we will use the notation we introduced previously and will introduce a few new notations for calculating negative prevalence, namely r_i^-—the occurrence frequency of the i-th descriptor in the document profile of nonpertinent documents—and n^-—the number of nonpertinent documents in the marked set. We will assume first that the i-th descriptor occurs only in pertinent documents. This means that negative prevalence will equal zero and then Ψ_i (prevalence) equals positive prevalence, that is, in this case Ψ_i is calculated only using pertinent documents. For this there is the following formula:

$$\Psi_i = \frac{r_i \cdot 1}{n \cdot R_i}.$$

In many ways this calculation is similar to that used in the M-algorithm, the only difference being that the authors disregard the size of the search collection. The authors, incidentally, use different symbols and a different form of writing. In addition, they normalize the result, but the essence of the result does not change. While preserving our notation, we will rewrite the formula as suggested by Dillon and colleagues:

$$\Psi_i = \frac{r_i/n}{\log R_i}.$$

Let us assume now that the output contains only nonpertinent documents (the marked set consists only of nonpertinent documents). In this case, positive prevalence for every descriptor will equal zero, while Ψ_i will equal negative prevalence:

$$-\Psi_i = \frac{r_i^-/n^-}{\log R_i}.$$

If the i-th descriptor occurs both in the pertinent and nonpertinent documents, Ψ_i is calculated this way:

$$(\pm)\Psi_i = [(r_i/n) - (r_i^-/n^-)]/\log R_i.$$

In our opinion, it is exactly such use of nonpertinent documents that presents certain interest, because it probably helps in a number of cases to more fully take into account the use of a descriptor in the search.

The next distinction lies in partitioning descriptors into zones. Dillon and colleagues suggested forming zones, not only for the descriptors whose value Ψ_i is positive, but also for the descriptors with negative values. Figure 7.4 illustrates this type of partitioning.

All zones range from 1 to −1, because the result of calculation of the descriptor search prevalence is normalized. With this type of partitioning, the