performed by an intermediary searcher in translating a query from natural language into IRL. The automation of this intellectual process is difficult to carry out for several reasons. First, the information need (POIN) does not have clear boundaries and hence cannot be expressed precisely, even in natural language. Second, in terms of the search, the intermediary searcher has to guess the best combination of descriptors drawing on his or her own experience, intuition, and methodological recommendations instead of relying on formal rules. This process has traditionally been considered to be creative, and the efforts of the system developers are directed toward supplying the intermediary searcher with additional information about the system (such as the frequency of using descriptors), which, in their opinion, will improve the search quality. In creating automatic methods of query indexing, the researchers, understandably, do not even aspire to copy the intermediary searcher's intellectual activity. Typically, they proceed from what they think are pragmatic assumptions, which are often not formulated explicitly. Only a few algorithms for constructing disjunctive normal forms have been created so far. Next, we discuss the various approaches implemented in these algorithms.

In 1970, we suggested and published the first algorithm for the automatic construction of query formulations in Boolean form (Frants, et al.). Since 1969, the program implementing this algorithm (written in ALGOL) has been tested with encouraging results. In developing our algorithm, we used the following pragmatic assumptions.

Information about the user's POIN has to be presented to a system in a natural language (as mentioned previously, we refer to this representation as a search request). This assumption is based on the common notion that the natural language is the most convenient and simple form for the user to present his or her informational need. In other words, whenever the users express their POIN in the form most convenient to them, the system acquires the fullest and the most precise information about the POIN (see, for example, Barker, Veal, & Watt, 1972; and Schaffer, March, & Berndos, 1972). Yet different forms are possible in formulating search requests, which leads to the following assumption: the formulation of the search request has to be the most simple and convenient form for the user. How do we choose such a form? Search requests can be formulated either of two ways:

1. With somebody's help.
2. With no help at all.

In the first instance, users can be aided by both IR system experts (an intermediary searcher, for example) and/or outsiders (psychologists, for instance). Both methods (1 and 2) are used in practice, but the last one is more prevalent due to the fact that most experiments showed that this method is not only the simplest, but the most effective. Therefore, we consider it necessary to allow the user to formulate his or her own request.

Even in cases where the users formulate their own requests, they can do it orally, in written form, or in the form of marked documents. We define a marked document as *a document that the user determines to pertain to an information need that the user wants to express in natural language*. The set of marked documents used to formulate a search request is called a *marked set*. We now elaborate on how a request is formulated using marked documents.

Back in the early 1960s, researchers noticed that in trying to explain his or her POIN to the intermediary, the user often presented certain documents that were intended to give an idea of what the user wanted. This situation is analogous to formulating a request with the aid of marked documents. We suppose that all documents in the collection are in a natural language and that all documents, which may be presented by the user as marked ones, are also written in a natural language. In other words, within the framework of the created method, we assume that whenever the user expresses his or her POIN by a set of marked documents (written in a natural language), we are dealing with a search request.

Several experiments have shown that a query formulation based on marked documents seems to be more effective than a query formulation constructed using other forms of search requests (see, for example, McCash & Carmichael, 1970; and Moody & Kays, 1972). Researchers also noticed that whenever the user had some marked documents and knew that he or she could formulate requests using these documents, the user was more willing to take this opportunity than other opportunities. For that reason, the created algorithm (called *M-algorithm* in the published work) was first and foremost designed for search requests formulated in the form of marked documents. We say "first and foremost" because not every user can (or wants) to make a request in such a form. Therefore, the users had the opportunity to formulate their requests in any form convenient to them. This option was taken into account in developing the M-algorithm, which means that in the system such a request was viewed as one marked document representing the user's POIN.

In creating the M-algorithm, we were mainly concerned with finding a way to allow the selection of useful (in terms of search) descriptors and combining them successfully (again in terms of search) into disjunctive normal form by the logical operators AND and OR. To determine a descriptor in a search request, which would be important in a search, we set forth an idea sometimes called an *inverse documents frequency*. This idea is based on the assumption that the importance of the descriptor is proportional to the frequency of its occurrence in the document profiles of the marked documents and inversely proportional to its occurrence frequency in the document profiles of the collection of documents. In other words, we talk of a "counter" probability when the maximum probability of descriptor occurrences is desired in the document profile of the marked documents and when the minimum probability of its occurrence is desired in the document profiles of the entire collection of documents.

To calculate the value of descriptor's importance for the search, the fol-