AUTOMATED ASSISTANCE IN THE FORMULATION OF SEARCH STATEMENTS FOR BIBLIOGRAPHIC DATABASES

MICHAEL P. OAKES and MALCOLM J. TAYLOR*
Department of Computer Science, University of Liverpool, P.O. Box 147, Liverpool L69 3BX, U.K.

(Received 17 January 1997; accepted 27 May 1998)

Abstract—We report on the design and construction of features of an automated query system which will assist pharmacologists who are not information specialists to access the Derwent Drug File (DDF) pharmacological database. Our approach was to first elucidate those search skills of the search intermediary which might prove tractable to automation. Modules were then produced which assist in the three important subtasks of search statement generation, namely vocabulary selection, the choice of context indicators and query reformulation. Vocabulary selection is facilitated by approximate string matching, morphological analysis, browsing and menu searching. The context of the study, such as treatment or metabolism, is determined using a system of advisory menus. The task of query reformulation is performed using user feedback on retrieved documents, thesaurus relations between document index terms and term postings data. Use is made of diverse information sources, including electronic forms of printed search aids, a thesaurus and a medical dictionary. The system will be of use both to semicasual users and experienced intermediaries. Many of the ideas developed should prove transportable to domains other than pharmacology: the techniques for thesaurus manipulation are designed for use with any hierarchical thesaurus. © 1998 Elsevier Science Ltd. All rights reserved

1. INTRODUCTION AND BACKGROUND

The Derwent Drug File (DDF), previously called Ringdoc, contains over 1.2 million references compiled from over 1100 pharmaceutical journals, with over 50,000 new abstracts being prepared annually. The DDF is maintained by Derwent Publications (Derwent), and is available online to subscribers of the DDF literature service, through the DIALOG retrieval system or on CD-ROM. Currently, pharmacologists wishing to access the DDF database must express their information need verbally to professional search intermediaries, who translate this information need into the most appropriate DDF syntax search statement in order to access the database and retrieve the required documents. There is a need for automated systems which assist the nonexpert user in the formulation of such search statements. Studies have shown that computerised search intermediaries provide the advantage of always being available without appointments (Morrissey et al., 1986), and are available to the many potential semicasual users whose need does not warrant the service of a professional search intermediary (Paice, 1986). The large number of domain experts who are potential users of an automated intermediary system is illustrated by the fact that 87% of chemists who use computers at work were found to make use of textual bibliographic systems (Philip, 1996).

*Corresponding author. Tel.: +44-151-794-3682; Fax: +44-151-794-3715; E-mail: m.j.taylor@csc.liv.ac.uk.
1.1. Existing intelligent intermediary systems

One common feature of all intelligent intermediary systems described in the literature is that each system must possess knowledge. This knowledge may be of search tactics (Hawkins and Wagers, 1981), knowledge of the requester and the requester's information need (Tan et al., 1989), domain knowledge (Smith et al., 1989) or knowledge of the documents themselves (Simmons, 1987).

1.2. Intelligence: knowledge types

1.2.1. Knowledge of search tactics. One early intermediary system which possessed knowledge of search tactics was Paperchase (Horowitz and Bleich, 1981). Paperchase contained predefined search tactics to structure the search along a predefined path without an explicit series of commands input by the user, for example, based upon an analysis of the MeSH terms which are common to the greatest number of retrieved citations. The CITE system (Doszcocs, 1983) implements tactics for known item searches, and PSI (Shepherd and Watters, 1984) offers the user search strategy suggestions. KISIR (Sormunen, 1989) also provides search tactics online, including those for query reformulation, and provides user guidance and context sensitive help at each stage of their implementation. METACAT (Chen, 1992) employs inference heuristics based on known search strategies.

1.2.2. Knowledge about the user. A number of intelligent intermediary systems acquire knowledge about the user, either by building up a profile of the user which might include the user's level of search expertise and domain knowledge at the start of the session, or by iteratively refining a model of the user's expertise and information need based on the user's interaction with the system. Banwell advocates the elicitation of individual user profiles at the start of the search session (Banwell, 1989), while other authors advocate the classification of users into stereotypes with common interests and levels of expertise. The RADA system classifies its users into special interest groups (Tan et al., 1989). Stewart and Davies (1997) describe both statistical term-based and artificial intelligence approaches to user profiling, as well as their own notion of social filtering, where judgements made by one user about an information source are passed on for the benefit of other users.

Knowledge of the information need can also unfold during the process of user feedback. For example, interactive learning of the user's information need may be acquired from the user's judgement of documents retrieved at an intermediate stage of the search process, or from his or her choice of additional thesaurus terms to be added to the original query formulation (Aigrain and Longueville, 1994). Oddy's THOMAS system (Oddy, 1977) is concerned with modelling the user's information need through interactive dialogue rather than by query formulation as performed by the system described in this paper. Two systems which attempt to build up an accurate request model by analysing the query and interacting with the user are IOTA (Chiaramella and Defude, 1987) and FPR (Croft, 1987). User profiling may be less necessary for an in-house system where the users are a homogeneous group of experts than for a public access system which is usually available to a less expert, heterogeneous user population.

1.2.3. Domain knowledge. An example of the incorporation of domain knowledge into an intermediary system is EP-X (Smith et al., 1989) which uses its knowledge of environmental chemistry to communicate with the user about the meaning of words and topics. PLEXUS (Vickery, 1989) employs a clarification dialogue, asking such questions as “is X a plant?”. Rau’s SCISOR system (Rau, 1987) stores knowledge in the domain of corporate take-overs. An example of knowledge about documents is the logic-based text representation of Simmons (1987), which may be employed for question/answer as well as retrieval systems.
1.3. Intelligence: knowledge representation

The knowledge held by the search intermediary and subject knowledge can be represented in a number of ways. The EP-X system employs frames (Smith et al., 1989), while RADA employs a semantic network (Tan et al., 1989) and CANSEARCH (Pollitt, 1987) rules. In each case, the thesaurus applied in controlled indexing is an important source of subject knowledge (Pollitt, 1989b). Both McMath et al. (1989) and Bertrand-Gastaldy and Davidson (1986) describe the graphical on-screen representation of thesauri, and Pternick (1984) suggests a number of enhancements which can transform thesauri into true searching vocabularies. A thesaurus can be considered both as a semantic net and as a relational database (Jones, 1993). In Shoval’s system the knowledge base is represented as a semantic network (Shoval, 1985). Cohen and Kjeldsen’s GRANT system is based on a semantic net where the nodes represent areas of research, and the user’s research interests become associated with the agencies supporting work on related topics through constrained spreading activation (Cohen and Kjeldsen, 1987). SHRIF (Findler et al., 1992) employs a semantic net which consists of atomic nodes corresponding to individual requests and composite nodes corresponding to common requests. The rules in CANSEARCH are for the display of frames of concepts and terms, and also for search statement generation (Pollitt, 1987). RUBRIC (McCune et al., 1985) employs production rules to represent a hierarchy of retrieval topics with fuzzy context expressions and specific word phrases at the bottom. For example, “\( A \mid B \rightarrow \text{class C} \)” denotes that both A and B are members of class C, while “\( C \rightarrow D \)” means that C and D are synonyms. By chaining through its selection rules, RUBRIC is able to explain why a document was selected. Frames enable special purpose, fast deduction algorithms to extend explicitly held beliefs to a larger, virtual set of beliefs through the property of inheritance (Fikes and Kehler, 1985). COALSORT (Monarch and Carbonell, 1987), PLEXUS (Vickery, 1989) and EP-X (Smith et al., 1989) all represent topics as frames which support the interpretation of user entries and the generation of topic refinement suggestions.

Beale et al. (1989) have produced the ADAM system, which employs a neural network for the task of user profiling. Wettler and Rapp (1990) have produced a connectionist model which assumes that the associative strength between terms depends on their probability of co-occurrence in the same documents. This model was extended to enable term selection to be simulated by an associative lexical net (ferber et al., 1995). Larson’s CHESHIRE system (Larson, 1992) which employs classification clustering is able to support browsing by hypertext and allows free text input. Engels et al. (1992) and Rochfeld and Negros (1992) describe the use of entity-relation models in knowledge representation. The PLEXUS system (Vickery and Brooks, 1987) also employs a form of entity-relation model to describe the semantic categories of keywords and their relations.

Apart from considering various modes of internal knowledge representation, which enable the making of inferences, this paper considers the presentation of system knowledge to the user. For example, the use of menus to display the thesaurus has been described by Pollitt (1989a). The OAK system (Meadow et al., 1995) encapsulates knowledge of search statement formulation in a mainly menu-based interface language, while the HIBROWSE system uses a series of menus which appear in multiple windows to perform the related task of SQL statement generation for querying databases (Pollitt et al., 1994).

In the creation of an automated intermediary system, modules should be kept separate for independent modification, testing and replacement (Chiaramella and Chevallet, 1992). The communication between modules is an issue considered in this paper.

The literature thus describes a variety of structures for representing the knowledge required by automated intermediary systems. Semantic nets enable inference by the spreading activation mechanism, and the inheritance of properties pertaining to related
nodes. The DDF thesaurus is equivalent to a semantic net, and the application of both
these inference modes are investigated in this paper. Frames also allow the
representation of relations between concepts. This paper considers the filling of frame
slots to represent the stepwise development of the DDF query.

Another mode of knowledge representation is in the form of production rules.
However, according to Brooks’ criteria (Brooks, 1987) for the tractability of a given
domain to expert system techniques, the domain should be homogeneous and consist of
a limited number of entities and relations. In contrast, the task of an intelligent
information retrieval system involves a number of functions, including users and their
problems, documents and their descriptions and retrieval heuristics. Hence the domain
is not narrow and homogeneous, and involves many objects and their relations. In
keeping with Paice’s notion (Paice, 1986) of an intermediary system which employs few
rules and does not make lengthy chains of inference, this paper does not describe the
production of a traditional expert system.

1.4. Role of the search intermediary

The user of information retrieval systems such as DDF has to rely upon a search
intermediary as an intermediary between himself and the system. The search
intermediary will be familiar with thesauri, Boolean retrieval strategies, truncated terms,
synonyms, acronyms, term codes and query input formats such as terminal command
languages. He or she will know the full range of searchable fields and be familiar with
the appropriate search aids (Vernimb, 1987). Professional intermediaries use four main
types of knowledge, namely knowledge of how particular databases are constructed,
knowledge about the domain being searched, knowledge of the user and knowledge of
experienced search intermediaries use metaknowledge, which consists of factual
knowledge (demonstrated by the choice of nouns and adjectives made during the search
process), experiential knowledge (the ability to relate a present event to previous search
experience) and process knowledge (the development and progression of a given search
statement solution).

The intermediary has good communication skills for question analysis and the ability
to formulate the question into the concepts which best describe it. The search
intermediary does not see the question as simply a string of jargon terms to be entered
indiscriminately, but maps an understanding of the topic of interest or the user’s
information need onto an overall search plan or strategy, which may include the use of
qualifiers to denote the context of terms or the relations between them. The search
intermediary reacts appropriately to answers different from those expected, and shows
flexibility and originality in redesigning the search statement. The intermediary will
know when further effort in such query formulation will probably be unrewarding
(Vernimb, 1987).

A series of interviews was held with Dr M. Randall, a professional search
intermediary at Pfizer Pharmaceuticals, in order to ascertain the main subtasks involved
with query formulation, and then to devise a general schema whereby an automated
intermediary system might mirror these subtasks. In each case, the initial standard
request form was retained, all stages of the query formulation process were recorded
automatically by the DIALOG retrieval system, and all comments made by the search
intermediary and the observer’s own thoughts at each juncture were written down. It
was found that the task of the intermediary system consists of three parts, and that the
following subtasks should be considered:

1. The expert intermediary system must first accept terms chosen by the user to express
   the information need. At Pfizer, the pharmacologist requiring the search routinely
   writes the initial query on a standard request form, and has the option of supplying
a list of specific related keywords. Thus the first task of the system is to assist in term selection, by mapping the user selected terms onto the relevant DDF keywords, unless they are in a stop list. In general, there are two extremes in query formulation. If the user has supplied very little information, then several broader and alternative terms must be found. If the user has supplied many initial keywords, the list must typically be pruned. Although any pharmacological term will probably retrieve some documents of relevance from the DDF database, the use of assigned keywords will make the search more specific and efficient.

2. Once the necessary set of keywords has been ascertained, suitable contextual information should be appended to each keyword by the appropriate assignment of logical operators and delimiters. Keywords are separated into a series of “sentences”, within which all the keywords are contextually linked. The LINK operator (L) is used to combine drug names with associated keyword concepts such as diseases or organisms into sentences, and the AND operator is used to combine sentences with each other. The search statement PYRIDINE(L)BIOSYNTH. will retrieve abstracts concerned specifically with the biosynthesis of pyridine, while PYRIDINE AND BIOSYNTH. will retrieve abstracts which discuss both pyridine and biosynthesis, but not necessarily the biosynthesis of pyridine.

In addition to the correct selection and linkage of keywords, it is also necessary to append a relevant DDF delimiter or qualifier to each term to denote its context. These are used to delimit precisely the field of reference of scientific terms which are already specific, rather than to overcome the problem of word sense ambiguity (Jones, 1993). For all keywords other than drugs or diseases, the qualifier FT (further term) is routinely appended. However, drugs and diseases may take any one of several qualifiers according to the context of the query. For example, if a drug is used in the context of treatment, the TR qualifier is appended to the drug name. The range of drug and disease qualifiers employed by the DDF is given in Table 1.

Both these indexing features are illustrated by reference to the query Find references where quinidine is interacting with verapamil. The relevant DDF syntax statement is DI = QUINIDINE (L) DI = VERAPMIL, where DI denotes a drug interaction, and the LINK operator (L) denotes that quinidine and verapamil interact with each other, whereas the use of the AND operator would have retrieved papers where these drugs react with any other drug, not necessarily with each other.

The task of appending contextual information can be performed by a combination of (a) detecting inherent contextual information in a given keyword, e.g. pharmacokinetics implies that the context is drug-metabolism, or (b) eliciting contextual information by dialogue with the user.

3. Query reformulation. In cases where a syntactically correct DDF search statement yields either too many or too few documents, the search must be either broadened or narrowed. In cases where the user has produced little information, the user will generally be satisfied with about 10 documents. On the other hand, when much initial information has been provided, the requester may be prepared to scan as many as 100 retrieved abstracts. Query reformulation is also necessary if the set of retrieved documents does not fully satisfy the user’s information need. Techniques for query reformulation include (a) substitution of keywords by the higher or lower

<table>
<thead>
<tr>
<th>Qualifier</th>
<th>Meaning</th>
<th>Applicability</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI</td>
<td>drug interaction</td>
<td>drugs only</td>
</tr>
<tr>
<td>DM</td>
<td>drug metabolism</td>
<td>drugs only</td>
</tr>
<tr>
<td>OC</td>
<td>other context</td>
<td>drugs and diseases</td>
</tr>
<tr>
<td>RC</td>
<td>reference compound</td>
<td>drugs only</td>
</tr>
<tr>
<td>ST</td>
<td>side effect/toxicity</td>
<td>drugs and diseases</td>
</tr>
<tr>
<td>TR</td>
<td>treatment</td>
<td>drugs and diseases</td>
</tr>
</tbody>
</table>
terms in the DDF thesaurus, (b) the substitution of logical connectives between the search terms, and (c) user feedback regarding the dropping of relatively unimportant terms (where the terms are combined by the AND operator) to broaden the search and the addition of further terms in order to make the search more specific. The information requester may be present as the search is carried out and may indicate to the search intermediary when the search output seems satisfactory.

Five actual DDF searches, considered by Dr Randall to be typical of those performed at Pfizer, were analysed. One aim was to determine the extent to which the optimal search strategy as defined by Dr Randall could have been inferred from information contained within the DDF instruction bulletins, and to make this information available online and easy to use by nonexpert users. Dr Randall reported that the DDF instruction bulletins were typically consulted for about 40% of searches, either merely as a double check (about 20% of searches) or to look up information unknown to the professional intermediary (also about 20% of searches).

The DDF thesaurus (Ringdoc, 1989) is available in printed form and provides a list of about 15,000 keywords together with synonyms and parent terms. The four thesaurus appendices A to D provide respectively (a) definitions of nonobvious terms, where everyday terms such as healthy volunteers must be described as HUMAN, (b) definitions and searching hints for chemical substructure terms, (c) a hierarchical classification of generic terms and (d) a listing of all specific keywords collated under each generic term.

Consideration of the five case studies revealed recurring themes which gave rise to ideas which were later used in the construction of the automated intermediary system. For example, similarity between the character structures of the names of chemicals can reveal structural similarity between the chemicals themselves, as exemplified by such suffixes as -oxicam and prefixes as isox-. This suggests the feasibility of grouping related terms using approximate string matching techniques which measure the similarity of their constituent characters. The DDF thesaurus (Ringdoc, 1989) shows that many drug names offer a choice of parent terms according to whether structure or function is more important. This suggests the potential usefulness of an on-screen thesaurus browser, similar to the strictly hierarchical MenUSE system developed by Pollitt (1989a), but allowing alternative modes of traversal allowed by the option of more than one parent term. Since the thesaurus codes reveal the generic nature of the terms in each subtree, (for example, all terms with codes commencing with 1.2 are drugs), entry to the thesaurus browser at points other than the root was postulated. Synonyms are often discovered during the query reformulation process, for example documents retrieved in response to piroxicam contain the index term feldene. A means of monitoring such occurrences was built into our query reformulation module. The use by the requester of such acronyms such as BBB for blood brain barrier suggested the need for an acronym look-up table.

1.5. Building processes into a system

Upon consideration of the interviews with the search intermediary, we feel that a complete expert intermediary system should consist of the processes enumerated in Fig. 1. The terms primary search equation (PSE) and final search equation (FSE) are adapted from Chiaramella and Defude (1987). The optimally refined version of the FSE will be referred to as the optimal search statement (OSS).

The configuration of our intermediary system is shown in Fig. 2. The thesaurus is the main source of knowledge about term relations and the domain. It has been transformed off-line into a system of codes which reflect the position of the term in the thesaurus hierarchy, necessary for the automatic generation of the thesaurus browser described in Section 2.2.2 and the query reformulation process described in Section 4.
The thesaurus browser allows the user to browse the thesaurus contents. The term conflation module encompasses a variety of techniques for the identification of term variants, including both the measures of character similarity described in Section 2.1.1 and morphemic analysis as described in Section 2.1.2 to determine semantic similarity.
The term definition browser is an automatically generated hypertext system, described in Section 2.2.1, which allows users to browse through the entries in the DDF book of term definitions. The advisory menus, in conjunction with the user, enable the stepwise creation of the search statement as described in Section 3, while the system of query templates described in Section 3 converts common query types directly into DDF syntax. System knowledge is derived from various sources, in particular the search intermediary and documents produced by the database manufacturers, namely the Derwent Instruction Bulletins (Ringdoc, 1986).

2. USER INPUT PROCESSING

2.1. Vocabulary selection

The intermediary system must first accept user input in natural language. User input is regarded as being composed of keywords and noise. The first task of the system is thus to assist in term selection by mapping user-selected terms onto relevant DDF
terms unless they happen to be in a stoplist. Although any pharmacological term will probably retrieve some documents of relevance from the DDF database, the use of a controlled vocabulary or indexed keywords will make the search more specific and efficient (Cimino, 1995). Terms relevant to the initial user input term may be related orthographically, according to similarities in character structure, as described in Section 2.1.1, or related semantically, where terms are similar in meaning, as described in Section 2.1.2.

2.1.1. Character-based term matching techniques. Hall and Dowling (1980) describe a number of approximate string matching techniques which measure the similarity between the character structures of words, and thus enable the conflation or grouping of similar and equivalent words. This can enable a wider catchment area of alternative grammatical forms, spellings and abbreviations to be mapped onto a narrower set of system recognised keywords, but also users may be led to new unanticipated terms. User filtering is generally necessary to identify the required keywords from the list of terms produced by a given string matching technique. Regarding groups of terms as a single class will reduce the lexicon in size and ensure that all records indexed by that class of terms will be retrieved by any member of that class. Character-based similarity is of interest when considering the vocabulary employed in pharmaceutical databases, since the existence of common domain specific prefixes and suffixes such as di-, benz- and -ate means that orthographically similar terms are often related in meaning.

With term truncation, all terms with the same N initial characters are considered equivalent. Truncation is a standard feature of the DIALOG retrieval system, where for example the syntax react? will retrieve references to reacts and reaction. In our experiments, using Jardine and van Rijsbergen’s E measure (Jardine and Van Rijsbergen, 1971), the optimum truncation length was found to be 5 (Oakes and Taylor, 1992). Longer truncation lengths might result in related terms not being picked up, while shorter truncation lengths might result in nonrelated families of words such as mice and micelle becoming confused.

Stemming rules such as those of Porter (1980) facilitate the removal of common suffixes, designed to reduce grammatical variants of a word to the base form of the equivalence class. For example, computing might be reduced to comput. Prefix rules can also be created to deal with such terms as triphosphate. In our experiments stemming rules tended to produce high precision but poor recall (Oakes and Taylor, 1992), the most effective set of stemming rules encountered according to the method of Paice (1996) being those of Paice and Husk (Paice, 1990).

Some of the shortcomings of stemming rules can be overcome by the use of character-based association measure verified by user feedback. Adamson and Boreham (1974) consider the number of matching bigrams or pairs of consecutive characters in pairs of character strings, and then calculate their similarity using Dice’s similarity coefficient (Dice, 1945), which is twice the number of matches divided by the total number of bigrams in both words. The resulting value is 1 for an exact match, and 0 when there is no match at all. When comparing pediatric and paediatric, American and British variants of the same word, the bigram sets PE-ED-DI-IA-AT-TR-RI-IC and PA-AE-ED-DI-IA-AT-TR-RI-IC are obtained and Dice’s score is 14/17 = 0.82. Bigrams are preferred to monograms which give no information about the order in which characters occur, and so would allow anagrams to be deemed equivalent. In our experiments, trigrams were found to be marginally more effective than bigrams.

Of the methods examined for the conflation or grouping of terms which occur in Appendix A of the DDF thesaurus, namely truncation, Dice’s similarity coefficient and five sets of stemming rules, simple truncation was found to be the most effective according to the E measure (Oakes and Taylor, 1992).

As part of the construction of a general user input free text processor, we have implemented a dictionary for the partial matching of compound search terms, where for example the input term prostaglandin will retrieve the related compound term prostaglandin-antagonists. This dictionary was produced by scanning the DDF
thesaurus, storing all hyphenated terms and allowing their constituent single terms to act as keys.

All of the above techniques have been combined into a series of modules written in Pascal designed to map user input terms onto keywords recognised by the system. The list of keywords thus built up constitutes an interface to the term definition browser which will be described in Section 2.2.

2.1.2. Semantic term matching techniques. In order to retrieve useful alternative terms, which are not closely orthographically related, we have incorporated synonym and acronym dictionaries into our system. The synonym dictionary was derived automatically from the DDF thesaurus, where entries such as *obsidan use propanolol* indicate that one term is preferred to another. The acronym dictionary was compiled from a list of entries suggested by DDF subscribers, such as HPLC for high pressure liquid chromatography.

Further instances of semantically related terms which are not necessarily orthographically related arise, particularly in the domain of medicine, due to the use of terms of Latin or Greek derivation alongside their English equivalents. Instances of such semantically-related terms can be found by morphemic analysis, where terms are analysed into their smallest indivisible meaningful units or *morphs* (Leech, 1975). The set of morphs representing a given grammatical element (e.g. *thio-, sulph- and sulf-* which all mean sulphur) is known as a morpheme, which can be represented by a single canonical lexical form. Morphemic analysis is an extension of stemming rules, where the primitive concepts within a term are found by recognition of its constituent lexical morphs. For example, the term *cardiopathy* consists of the morphs *cardio-* meaning heart and *-pathy* meaning disease.

Two domain specific help systems have been produced: (1) a system which acts as a translator of medical terms of Greek or Latin derivation with their everyday English equivalents and (2) a system which enables morphemic analysis to be performed on each term in the organic chemicals subtree of the DDF thesaurus.

The morph and morpheme dictionary for the translator of medical terms, where each recognised morph is represented alongside its canonical medical form, was derived from the Oxford Concise Medical Dictionary (Martin, 1990), which provides a translation of over 500 Greek and Latin term fragments. Whenever the translator is presented with a medical term, the character sequences within that term are matched against the morph and morpheme dictionary. Whenever a match is found, the corresponding morpheme is displayed. For example, *achromatopsia*, the inability to perceive colour, is translated as *a* = not, *without*; *chromat* = colour; *opsia* = condition of vision. The translator is useful for a user who is unfamiliar with medical terms of Greek and Latin derivation, since it provides translations of these without the need to consult a complete dictionary of medical terms.

Morphemic analysis was also performed on all terms in the organic chemicals subtree of the DDF thesaurus, so that each term was assigned a set of morphemes identified in that term. In addition, each child term in the hierarchy inherited the codes of its parent term. The various sources of information employed in the derivation of this morpheme dictionary are given in (Oakes and Taylor, 1994). User input terms are also assigned a set of morphemes to their constituent morphs. For example, the input term *aminocyclitol* would be analysed into the morphs *amino-*, *-cycl-* and *-itol* which are assigned the morphemes [amino] [cyclic] and [alcohol]. This set of morphemes is matched against each of the stored morpheme sets in the database, such as those derived from the term *peptide*, *cyclic*, namely [carboxyl], [amino] and [cyclic]. Dice’s similarity coefficient for the degree of match between the two sets of morphemes is 0.67. All stored terms with an above threshold similarity to the input term are ranked and displayed for user approval or rejection. In addition, an explanation of each output term is given in terms of its constituent morphemes.

Both systems allow explanations of unfamiliar terms to be provided rapidly, by matching the terms against a relatively small set of morphs rather than requiring an entire medical or chemical dictionary to be stored online. The 1252 chemical terms
found in the organic chemicals section of the DDF thesaurus can largely be represented by fewer than 100 primitive codes, and terms not found in that section can still be input by the user and interpreted according to the presence of recognised lexical morphs. Both the medical and organic chemical morpheme dictionaries discussed in this section are available on request from the authors of this paper.

2.2. Browsing

Browsing is an interactive and adaptive searching strategy, during which, retrieved items are examined in order to improve the success of further stages of the search. One aspect of browsing is the technique of examining materials retrieved at an intermediate stage of the search in order to come to a clearer understanding of the original information need. The system described in this paper incorporates two browsers, one which browses those records held in Appendix A of the DDF thesaurus, described in Section 2.2.1, and the other which enables browsing of the thesaurus itself, described in Section 2.2.2.

2.2.1. The term definition browser. Appendix A of the DDF thesaurus provides definitions of terms whose meanings are not self evident nor to be found in standard reference books, and thus forms a potentially valuable source of additional terms for the enhancement of DDF search statements. In most indexing systems, the primary key (e.g. a subscriber’s surname in a telephone directory) is well known to the user. However, the records in Appendix A are there precisely because their title terms are not intuitively obvious. In such cases, access to a relevant record may be facilitated by indexing the records according to the terms contained in all fields of each record. An example of an Appendix A record is given in Fig. 3. The top two fields, the title and explanatory text, are present in every record. The other three fields, containing related and more-specific terms and a suggested codeless scanning strategy, may or may not be present.

We have previously described our method of automatically generating a term definition browser for Appendix A of the DDF thesaurus from electronically stored records provided by Derwent Information (Oakes and Taylor, 1991). The browser accepts a list of user selected terms, and aims to retrieve pertinent Appendix A records by matching the set of supplied terms with the index terms which identify the Appendix A records. If matching of user input terms against terms in the explanatory text fields is selected, up to three records will be displayed, being those records whose text fields have the greatest (nonzero) computed similarity to the set of user input terms according to Dice’s similarity coefficient.

Once an initial set of records has been accessed, further records can be retrieved by means of explicit links stipulated by Derwent which act as pointers to “related” or “more specific records”, or by implicit links generated by overlapping sets of terms in the explanatory text of two records. The term definition browser is thus designed to allow the user to check the suitability of both the chosen and system-selected terms, and to discover related terms which may be of use.

2.2.2. The thesaurus browser. A DDF thesaurus browser was generated from appendices C and D of the DDF thesaurus, following the work of Pollitt who placed the terms of the MeSH thesaurus (Medical Subject Headings, 1986) in a single hierarchical

ANALGESICS
Drugs which relieve pain
*Related LOCAL-ANESTHETICS OPIOIDS
*More Specific NARCOTICS
*C/S /IT,FT ANALGESIC OR AMINOPYRINE OR ANTIPYRINE

Fig. 3. The structure of an Appendix A record.
menu-based browser called MenUSE (Pollitt, 1989a), where the user is led through a hierarchy of menus to the desired search term, and then is returned to the top level so that further terms may be found. The thesaurus browser thus facilitates the partial solution of DDF search statement formulation by enabling the selection of obvious DDF keywords. Such a system of menu-based term selection can act both as a "memory jogger" and to circumvent the problem of "you don't know what you want until you've found it". Menus do not involve any typing in of search terms by the user, which helps overcome the problem of alternative spellings and nomenclatures. They also prevent typing errors on the part of the user which may occur when attempting to input lengthy medical or pharmaceutical terms.

In order to create the hierarchy of menus, it was necessary to impose a coding hierarchy, such as that which already exists in MeSH, where the code number for each term denotes its position in the hierarchy of terms. Appendix C of the DDF thesaurus gives code numbers for the terms at the top two levels, e.g. (1) biological activities, (1.1) pesticides and plant hormones. This coding system was extended to include the third level terms e.g. (1.1.1.) acaricides and the lowest level terms given in Appendix D e.g. (1.1.1.1) aldicarb. In this way it was possible to generate interconnected menus covering all terms in appendices C and D, by employing such rules as a term with code N must be the parent of a term with the code N.1. The menus for terms appearing more than once in the thesaurus offer a choice of parent terms for moving upwards in the hierarchy. For example, the chemical term aminocyclitol will offer both monovalent N-containing compounds and alicycles as alternative parent terms, since it fits into both categories.

2.3. The advisory menus and the thesaurus browser

The thesaurus browser alone performs the task of vocabulary selection. Pollitt (1989a) has shown that a thesaurus browser alone may solve the problem of formulating MEDLINE search statements. However, the DDF indexing system is richer, with the LINK operator and the use of some nonobvious keywords, and thus a hybrid approach is required, where the thesaurus browser interacts with a system of advisory menus. The advisory menus perform the tasks of (a) user guidance, (b) context determination and the use of qualifiers, (c) nonobvious keyword selection, (d) logical operator selection and (e) control of the thesaurus browser, by restricting the search space to the required subtree and level of specificity. Thus the advisory menus have a dual role, both in vocabulary selection and context determination. The advisory menus were constructed manually, but employ only information which is available in the DDF search menu handbook (Ringdoc, 1986), which is designed as an aid to search intermediaries and is a guide to the stepwise derivation of syntactically legal DDF search statements.

2.3.1. The advisory menus. In a DDF search statement, keywords are separated into a series of sentences in which all of the keywords are conceptually linked. A typical sentence contains the name of a significant drug together with all keywords relating to that drug. The advisory menu system poses questions in the way a human intermediary would, enabling single term additions to be made sequentially to one of three sublists by analogy with the formation of linked lists in a computer program. These sublists are built up for each main aspect of the search equation, pertaining to drugs, organisms and diseases respectively. Each member of a sublist except a named drug may be the head of its own list, representing OR-ed concepts, if the user selects more than one option from the menu at a single stage of the interaction. Upon completion of the drugs sublist, it is linked to the organisms sublist. When the organisms sublist is complete, it is linked onto the diseases sublist. In this way a single list is built up with a drug keyword at its head. Thus one run through the advisory menus will form a linked
list corresponding to one sentence of the search statement. A sample interaction with
the advisory menus is shown in Fig. 4.

This shows a single sublist query, with a drug name at the head, obtained from the
thesaurus browser. Drug-related concepts form the remainder of the list. The obvious
keyword BIOASSAY is obtained from the thesaurus browser, while the nonobvious
keyword QUANT, is obtained from the advisory menus. All qualifiers are inferred by
the advisory menus.

2.3.2 Integration of the thesaurus browser and the advisory menus. Since the DDF is a
pharmacological database, it is assumed that every query posed to the advisory menus
must somehow be drug related. Thus the user must initially select either the name of a
specific drug or a group of related compounds, otherwise the system will assume that
the query pertains to all drugs. The advisory menus and the thesaurus browser are inter-
connected, enabling traversal from one to the other when necessary. In the combined
system, the relevant section of the thesaurus browser is called from the advisory menu
section whenever a wide choice of inputs is possible, for example when the selection of
a specific drug name or group of drugs is required.

The hierarchical organisation of the thesaurus browser in such cases should provide
the necessary guidance in the location of such terms. A return to the advisory menu
section allows natural language questions to be posed, which guide the user to those
DDF specific terms which are difficult to find using a thesaurus browser, covering such
factors as the synthesis or analysis of the selected drug, group of drugs or drugs in
general. The responses to these questions, selected from displayed lists of possible
responses, will result in the linkage of relevant additional keywords. The advisory menu
section also makes provision for the selection of qualifiers, and is responsible for the
correct combination of search terms, whether obtained from the thesaurus browser or
the advisory menus themselves.

The menu-driven process then ascertains the names of any organisms considered in
the query, where the top level categories are healthy humans, clinical patients,
laboratory animals, animal models, isolation of compounds from animals, plants and
microorganisms. Additional terms to be linked to the organism name are picked up,
such as PEDIATRICS if the study concerns human subjects within the relevant age
group. Finally the names of any associated diseases and associated information will be
obtained from the user (Oakes and Taylor, 1993).

3. QUERY TEMPLATES

With the assistance of Derwent Information Ltd., a questionnaire was sent out to
subscribers of the DDF database. One objective of this questionnaire was to identify
the commonest types of queries presented to the DDF database. The text of this
question was What are the most common types of query that you answer using DDF?
Please estimate the frequency of each type as a percentage of total DDF queries.

Six specified possibilities were suggested by Derwent Information, and the
respondents were allowed to suggest additional query types of their own. It was
explained to the respondents that we wished to derive statistics concerning the relative
occurrence of each query type. Our intention was to categorise these past query types
according to semantic type, as was done in the OAR (open architecture for reasoning)
project according to the user’s point of view (OAR, 1990), to assist in the selection of
commonly required input templates which map onto frames with valued slots. In this
way a system of query templates could be designed for the direct conversion of
common query types into DDF syntax.

1Derwent Information Ltd., Derwent House, 14 Great Queen Street, London WC2B 5DF.
The replies to this question were collated, and the results are shown in Table 2. In total, the templates suggested by Derwent (the top 6) covered 61.7% of cases reported by the questionnaire respondents, and a further 15.7% of cases could be covered by the other templates which were suggested by the respondents themselves. Other, less commonly requested queries also specified by the respondents but not tabulated constituted a further 10.2% of the total, while the remainder of the queries were unspecified, being merely grouped under the heading “other”.

AM: Select one of the following:
   1. A Specific Named Drug.
   2. Group of Drugs with a Particular Activity.
   3. Drug Activities in General.

User : 2
AM : Select a Drug Higher Term.
{The user is now sent to the Thesaurus Browser drug subtree, and retrieves the term ANABOLICS.}
AM : Current List :
   FT = ANABOLICS
{Drug higher terms are routinely assigned the qualifier FT = Further Term.}
AM : Select the Appropriate Context of the Drug Under Study :
   1. Pharmacology.
      :
   5. Analysis of the Drug.
      :
   12. Any or All Contexts.

User : 5
{The user is now sent to the Thesaurus Browser analysis subtree, and retrieves the term BIOASSAY.)
AM : Current List :
   FT = ANABOLICS
   (L) FT = BIOASSAY
{FT is routinely assigned if the term is not a drug or a disease.}
AM : Select one of the Following :
   1. Quantitative Determinations Only.
   2. Qualitative Determinations Only.
   3. All Determinations.

User : 1
{The Advisory Menus now add the non-obvious keyword QUANT. .}
AM : Current List :
   FT = ANABOLICS
   (L) FT = BIOASSAY
   (L) FT = QUANT.
{No further keywords selected.}

Fig. 4. Edited transcript of a session with the advisory menus (AM).
The responses given in Table 2, column 1 were shown to Dr M. Randall at Pfizer Ltd., and he provided a generic translation into DDF syntax whenever this was deemed necessary. Since these common query types were found to be generalisable, for example in cases where the generic term disease can be replaced by any specific disease name without altering the syntax of either the original query or the corresponding search statement, a set of templates could be created for the direct translation of each query type.

The original plan was to create templates corresponding to these common query types and to arrange them under a top level menu in order of decreasing usefulness. However, it was later decided to group certain query types together if they were considered to be closely related. For example, just one template was created to cover the two very closely related concepts side effects/toxicity and toxicity of a single drug. The frequency of use of each template shown in Table 2 is the sum of the two closely related query types catered for by the same template in such cases. A single menu entry was created for group of drugs in the treatment of a specific disease and single drug name in the treatment of a specific disease since these two queries differ only in the level of specificity of the drug term. A number of other templates were also extended to include classes of drugs, even if the class of drugs form of the query was not in the original list of 23 common query types.

Once the set of menu options and the set of relevant translations had been decided upon, it was possible to incorporate these into a system of query templates. Table 2 shows the top level menu selections (in lower case) alongside the translations for each entry given by Dr Randall (in upper case).

Each of the query types given in Table 2 provides one selectable option from the top level menu of the query templates module. Having made a selection of query type, the user by interaction with the program must provide specific instances of all generic terms contained within the corresponding DDF syntax query template, such as DRUG-CLASS or DISEASE. The choice of keyword in each case is constrained both by the fact that the keyword may only be retrieved from the relevant subtree of the DDF thesaurus, such as the drugs subtree, and the level of specificity of the keyword, where specific drug and disease terms are found at level 4 of the term hierarchy and their generic terms are found at level 3 of the DDF thesaurus. AUTHOR is elicited from the user in the format OTHER A.N., a FREE-TEXT term is input directly by the user, and a DERWENT-KEYWORD is retrieved from any part of the thesaurus at levels 3 and 4. All other components of the translated queries given in Table 2 are retained as literals.

Whenever the user is required to provide a specific instance of a generic term, the option is given of inputting a term directly, and this term will be accepted only if the term originates from the required subtree at the required level of specificity. Alternatively, the user has the option of calling the thesaurus browser before selecting a term. The query templates module controls the part of the thesaurus which may be browsed and the level(s) of specificity in the same way that the advisory menus control the thesaurus browser.

For example, the user might select option 8 from the top-level menu, Analysis of a drug. The template for this is OC = DRUG (L) FT = ANALYSIS. The user will therefore be required to select a single drug name such as ASPIRIN. The final output, after replacement of DRUG with ASPIRIN, will be OC = ASPIRIN (L) FT = ANALYSIS, which is a valid DDF search statement.

4. QUERY REFORMULATION

Query reformulation is the iterative process of refining the final search equation (FSE) into the optimal search equation (OSE), which were introduced in Fig. 1. Query
Table 2. Distinct query types and their relative frequencies

<table>
<thead>
<tr>
<th>Query Type</th>
<th>Frequency</th>
<th>FT = DRUG/DE</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Single named drug (34.6%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(2) Class of drugs (13.1%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(3) Work by a named author in a specific field (4.1%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(4) Interactions between two specified drugs (3.8%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(5) Interactions between two drugs only one of which is specified (2.7%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(6) Two named drugs in a combination preparation (3.4%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(7) Drug(s) in the treatment of a specific disease (1.9%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(8) Analysis of a drug (1.5%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(9) Pharmaceutical preparations of a drug (1.5%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10) Side effects or toxicity of drug(s) (2.1%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(11) Pharmacetics of drug(s) (1.4%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(12) Pharmacological effects of drug(s) in humans (0.4%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(13) Pharmacological effects of drug(s) in animals (0.4%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(14) Pharmacology of drug(s) (1.6%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(15) Physical data for drug(s) (0.8%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(16) Metabolism of a drug (0.6%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(17) Synthesis of a drug (0.4%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(18) Route of administration of a drug (0.4%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
reformulation is necessary in cases where a syntactically correct query results in the retrieval of too few or too many documents, or documents which do not adequately reflect the information need. This refinement process depends on the ability of the user to view the documents most likely to be retrieved by intermediate query formulations. The use of user feedback regarding the suitability or otherwise of retrieved documents to modify the intermediary system’s model of the information need is relevance feedback. In order for the system to select a set of documents likely to be of interest to the user for relevance feedback, a means of document ranking is required. Reformulating a query automatically with terms taken from documents that are most similar to the query has been described as a reliable method of finding further relevant documents (Wilkinson, 1997).

In this section a query reformulation module designed for integration into a complete intermediary system is described. The input to this module is the primary search equation (PSE), which is a set of DDF keywords selected by the user. The module accepts qualified keywords as input, although these do not occur in the PSE as defined. The output of this module is the optimal search equation (OSE), which is the syntactically legal search statement consisting of DDF keywords, qualified if necessary, and suitable logical operators which most closely describes the user’s information need. The final search equation (FSE), a syntactically legal, but not necessarily optimal search statement is never explicitly expressed during this query reformulation process.

The PSE or original set of query terms is analysed, and hierarchy codes are assigned to each term. Qualifiers are considered to constitute a fifth level of the DDF hierarchy, where for example if the hierarchy code for “STREPTOKINASE” is 1.3.13.32, then “DI = STREPTOKINASE” (denoting streptokinase in the context of a drug interaction) will be regarded as a child term of “STREPTOKINASE” and be assigned a hierarchy code of 1.3.13.32.1. A check is made to ensure that each qualified query term has a permissible qualifier. In the same manner, as the stored document file is read in, each document term will be assigned a code number corresponding to its position in the hierarchy, as was done in the generation of the thesaurus browser described in Section 2.3. More than one hierarchy code is assigned to terms which appear more than once in the DDF thesaurus.

The query reformulation process then commences by iteratively displaying documents calculated to be relevant to the current set of query terms, recording user approval or otherwise of these documents, then suggesting appropriate terms to the user for addition to, substitution with or deletion from the current query set. This process is repeated until a complete iteration has occurred in which the user has made no amendment to the query set. Finally, the system in conjunction with the user assigns logical operands to connect the individual query terms.

The semantic distance of each document from the query set is evaluated using the DISTANCE heuristic (McMath et al., 1989), which depends on the number of steps in the thesaurus between query and document terms, thus enabling document ranking. Other heuristics which estimate the conceptual distance between thesaurus terms are described by Lee et al. (1994) and Shinnou (1996). At each iteration a constant number of titles of documents deemed to be closest to the query set are displayed individually to the user in order of closeness to the query, to enable the user to pass relevance judgement on these displayed documents.

The possible query term refinements made possible by this query reformulation module are listed below.

1. A relevance score is maintained for each term in the thesaurus. If the document last offered to the user is accepted, all terms in that document will have their relevance scores incremented. Any term with above threshold relevance is automatically proffered to the user for possible inclusion in the query set unless it is related in the hierarchy to the query term. If this term is rejected, its relevance returns to 0.
2. If the document last offered to the user has been rejected, all terms in that document have their relevance score decremented. If the relevance score is below the negative threshold, and the term is in the query set, the user is asked whether that term should be deleted from the query set.

3. If the last shown document has been accepted by the user, and any term in that document is related in the thesaurus to any query term (i.e., is a parent or child of a current query term, as determined by the relation between the hierarchy codes of each term) then the user is given the option to replace the query term with the related document term. If this option is rejected, the user is given the option to add the related document term to the current query set without deleting any query term.

4. Finally, if the last displayed document has been rejected by the user, and any of the query terms are related to any term in that document, then the user is given the option to delete that query term from the query set.

In order to assist the user in deciding which changes in the query set might be advantageous, the user is provided with an estimation of the number of documents which will be retrieved (a) if the query set remains unchanged and (b) a given change is made to the query set. This will assist the user in deciding which changes to the query set will bring the number of retrieved documents closer to the desired number.

All terms which are related in the thesaurus, with identical top two level hierarchy codes, are assumed to be OR-ed together. For example, ASPIRIN (code 1.2.1.55) and PARACETAMOL (1.2.4.82) share the top two level codes of 1.2, showing that they both belong to the same (analgesics) subtree of the thesaurus and should be combined with the OR operator. The formula used to estimate the number of documents likely to be retrieved, or postings, when two terms are OR-ed together is

\[
\text{postings}(A \text{ OR } B) = (a + b) - (ab/t),
\]

where \(a\) is the number of postings associated with term A, \(b\) is the number of postings associated with term B and \(t\) is the total number of documents in the database — this term is included to account for those documents containing both terms. To estimate the postings for an OR-ed set consisting of more than two terms, the first two terms are OR-ed first, then subsequent terms are sequentially OR-ed with the existing OR-ed set.

In order to estimate the number of documents likely to be retrieved when two sets of terms are either LINK-ed or AND-ed together, the formula employed is

\[
\text{postings}(A \text{ LINK } B) = \text{postings}(A \text{ AND } B) = (ab/t).
\]

The operator assignment procedure is called just once at the end of the query reformulation process. The procedure takes the final list of query terms and calculates which OR-ed sets exist, according to which terms have common hierarchy codes at the top two levels. If all the terms in the final query constitute a single OR-ed set, then the operator assignment process is complete. If more than one OR-ed set is found, but only one of these is drug-related, then all other OR-ed sets are automatically LINK-ed to the drug-related OR-ed set. If more than one drug related OR-ed set is found, the user must select which of the nondrug OR-ed sets pertain to each drug-related OR-ed set. A nondrug OR-ed set may be assigned to more than one drug-related OR-ed set, but each drug-related OR-ed set may occur only once in the final search statement.

The actual procedure for allocating related OR-ed sets whenever two or more drug-related OR-sets are present is as follows:

1. A drug-related OR-ed set is identified.
2. All members of that set are displayed.
3. All nondrug OR-ed sets are displayed in turn.
4. The user must select which of the displayed nondrug OR-ed sets pertain to the drug OR-ed set currently under consideration. The user responses are stored in an OR-ed set connectivity matrix. The rows of this matrix correspond to drug-related OR-ed
sets, while the columns correspond to nondrug related OR-ed sets. Each drug-related OR-ed set should become LINK-ed to each nondrug OR-ed set, whenever there is a value of 1 in the relevant row and column in the matrix. A value of 0 shows that a particular drug-related OR-ed set does not pertain to a particular nondrug OR-ed set.

5. Repeat until no further drug OR-ed sets are found.
6. Separate drug related blocks of terms are to be AND-ed together to form the FSE, which is then displayed.

5. EVALUATION METHODS

Results for an evaluation of the system are not yet available. Ideally the system would be evaluated in a “live” setting, using real users with real information needs (Hancock-Beaulieu et al., 1995). Cornell (1995) advocates the use of two types of queries: (a) selective, where just one or two references are required to answer a specific question, and (b) comprehensive, where the user requires everything that has been published on a topic since a certain date. Ways of initiating a search are either to find all documents similar to one that has been found to be useful, or to start from interviews with the patron as routinely takes place at Pfizer. In order to evaluate the overall system, we could compare the documents retrieved in response to a selection of queries (a) by a search intermediary using traditional information sources, (b) by a search intermediary making use of the system described in this paper, (c) the end user unaided, and (d) an end user with access to the automated intermediary system.

To evaluate the relevance or otherwise of the documents retrieved, we can use the criteria of Froelich (1994), namely topicality, perceived validity by the user, and novelty. Given relevance judgements on the retrieved documents, recall and precision can be calculated. According to Su (1994), recall is more important than precision to users. However, recall is much more difficult to calculate than precision, because we need to know the number of documents in the entire collection which would have been considered relevant, had they been retrieved. Other criteria to evaluate the search could be efficiency (as estimated by search session time), utility (such as whether the information retrieved opened up a further research avenue) or user satisfaction, including factors such as the user’s confidence that all the information that there exists on a topic has indeed been retrieved (Su, 1994).

The main techniques for evaluating user responses to the system are the use of transaction logs, pre- and postsearch questionnaires and postsearch group discussions (Meadow et al., 1995). Jones et al. (1995) used transaction logging to monitor the behaviour of online thesaurus users, to test the effects of thesaurus-based query enhancement. The query reformulation module described in this paper already captures information about users’ judgements of documents, since this information is required for the selection of new search terms and documents. The other components of the system can easily be modified to echo all user inputs to a file, thus providing transaction logs of interactions. A logging facility can incorporate a time stamp (Hancock-Beaulieu, 1995). Hancock-Beaulieu et al. used data obtained by online questionnaires. The presearch questionnaire sought information about the purpose of the search. The post search questionnaire covered such issues as navigation, the query expansion and term selection processes, problems and suggestions for improvements. Subjective opinions of the user about the various aspects of the search process can be plotted on a 7-point Likert scale (Su, 1994).

The term definition browser can be evaluated using the method of Dimitroff and Wolfram (1995). Their method was to record the search times and number of thesaurus nodes visited for five specified search queries. Using test queries from the Derwent training manuals, where everyday language must be transformed into nonobvious
keywords, it would be possible to record how many of these keywords were found within a reasonable time limit by the search intermediary and the end user, working with and without the automated system.

6. SEARCH STATEMENT GENERATION FOR THE WORLD WIDE WEB

There has been much recent interest in the use of information retrieval techniques being used to access information on the World Wide Web (WWW). Furner (1997) cites assistance of the user in the formulation of a query as one of the key elements of a Web search service. SOURCERER (Rodgers, 1995) is an example of a thesaurus-based system being used to access material on the Internet, amenable to enhancement by the thesaurus-related facilities described in this paper. The LISTSERV software allows searching and retrieval by E-mail of discussion list archives. The approximate string matching and semantic term matching techniques described in this paper would help match users’ queries to the contents of messages. In fact, LISTSERV already has a SOUNDEX-like phonetic search option (McMurdo, 1995). The VERONICA software enables Boolean keyword searches within the menu titles of the WWW’s hierarchically-organised gopher servers. The advisory menus could assist in this task by assisting in the formulation of Boolean queries and restricting the range of the menu search. The Lycos search engine matches user queries against its the contents of its database by weighted keyword match (McMurdo, 1995); this process could be enhanced by query reformulation. The query reformulation techniques could also enhance the “more like this” option on the Excite search engine, or the “improve your search” links on the Open Text Index, both of which seek to obtain further documents based on positive user relevance judgements on documents retrieved earlier in the search (Lindop et al., 1997).

7. CONCLUSIONS

The intermediary system described in this paper embodies intelligence, in that it consists of a knowledge base which accesses both the DDF information sources and additional database-specific knowledge provided by a human search intermediary. These modules make up a system with “a knowledge base and inferential capabilities that can be used to establish connections between a request and a set of documents” (Croft, 1987). The system described in this paper is concerned with the conversion of information requests into DDF search statements, which can then be used to retrieve documents from the database itself. The main knowledge base in this system consists of the DDF thesaurus and knowledge derived from it, but the system also stores knowledge provided by the professional search intermediary, the DDF printed search aids and information derivable from them, and employs some external data sources such a chemistry data book (Earl and Wilford, 1991). The user is not required to provide additional domain knowledge, other than to accept or reject proffered domain-specific keywords. The techniques developed for processing the DDF printed search aids should be applicable to the printed information sources pertaining to other databases. The computerised search aids described in this paper are not controlled by typical expert systems, but rather along the lines of the less complex, potentially more practical intermediary systems envisaged by Paice (1986), which do not make lengthy chains of inference, but rather require user feedback after just one or two inference steps.

Section 2.1.1 describes the conflation or grouping of terms according to similarities in their character structures. The advantage of these techniques is that they are domain- and language-independent, assuming that terms related in their character structures are
also related in meaning in all domains. There is no need to construct an online
dictionary which gives the meanings of terms and term fragments as is necessary for the
semantic conflation of terms. In Section 2.1.2, techniques were also employed for the
conflation of terms according to their semantic similarity, in cases where terms may be
similar in meaning, even though they do not resemble each other in their character
structures. Morphemic analysis techniques were developed as an extension of stemming
rules, which enabled the creation of modules for the automatic translation of medical
terms of Greek or Latin derivation and the conflation of related chemical terms. All the
term conflation techniques described in Section 2.1.2 are domain independent, with the
exception of morphemic analysis for which new morpheme dictionaries must be hand
produced for transportation to other domains, or perhaps automatically derived from
machine readable dictionaries.

A browser was created to facilitate access to those records held in Appendix A of the
DDF thesaurus, which contains convenient window-sized records providing definitions
of nonobvious search terms. The browsable links in the system are both explicit, being
pre-assigned by Derwent Information, and implicit, being based on a coefficient of
similarity between the text content of the records. Browsers similar to the term
definition browser can be created using the techniques developed here, irrespective of
the domain, as long as the records are in the required format, with titles, explanatory
test and fields denoting predefined links with other records.

The procedures for generating the hierarchy codes required by the thesaurus browser
described in Section 2.2.2 can be employed to create new code sets corresponding to
alternative thesauri, enabling the automatic generation of new thesaurus browsers and
the query reformulation engine to operate in conjunction with alternative thesauri. The
advisory menus were specifically created for use with the DDF database, and
replacement of these would require the lengthy process of hand-coding the searching
advice produced by other database manufacturers. However, the techniques described
for intercommunication between the advisory menus and the thesaurus browser are
more generally applicable. The use of the advisory menus which link together keywords
in context leads into the second main phase of query formulation, that of context
determination. Selection of the correct context for common query types is also enabled
by the creation of a set of query templates. The set of query templates described in
Section 3 is domain specific, but these can be replaced with alternative templates
corresponding to other queries, either in the same domain or designed for other
databases.

In order to automate the third main task of the intermediary, that of query
reformulation, a module was created which iteratively modifies the user’s query by the
addition, deletion and substitution of related thesaurus terms in response to the user’s
assessment of retrieved documents. The method described in Section 4 will enable the
creation of a similar query reformulation module for any database indexed by a
hierarchical thesaurus.

In summary, techniques have been employed to assist in the three main tasks of the
search intermediary, namely vocabulary selection, context determination and query
reformulation. The techniques described for thesaurus manipulation are designed for
use with any hierarchical thesaurus. Although general techniques have been employed
in the construction of each module, each component employs search materials produced
by Derwent, and thus the work described in this paper will be of direct benefit to the
manufacturers of the DDF. The work described in this paper should be of interest to
companies such as Pfizer which subscribe to the DDF and other scientific bibliographic
databases, since the system contains both features which will enable the semi-casual
user to access the DDF directly, and also will be of use to experienced intermediaries
since information held in the printed search aids is made easily accessible.

Acknowledgements—We would like to acknowledge the support of Derwent Information Ltd. who provided
printed copies and electronic versions of the DDF thesaurus and appendices. We are also grateful for the
support of Pfizer Pharmaceuticals, in particular Dr M. Randall, in advising in the role of the search intermediary.

REFERENCES


