Chapter 3

The cirt system: users' view

3.1. Overview

As far as the user is concerned, the system is a simple searching system based on sets of terms (no Boolean logic), allowing relevance feedback and manual query expansion. It will log on automatically; it will retrieve and display documents in rank order, displaying a limited set of fields on the screen but preserving all details for a print file.

The search logic is simple term weighting, with document ranking by total weight of matching terms. Initial weights are based on collection frequency; after some relevant documents have been found, the weights are revised using the relevance information. All weights are internal to the system, but may be inspected by the user. Several other aspects of the system which are normally internal to the system may be inspected and/or changed by a more sophisticated user; for example, the system normally looks for the top 15 ranked documents, but this parameter may be reset. Some facilities available on the host are useable through cirt: e.g. term truncation and field specification. Also the user may talk directly to the host.

The system is limited to requests with a maximum of eight terms. Subject to this limit, terms may be added during a search, but may only be deleted if a search has not already been made.

The system is at present geared to searching Medline on Data-Star. The dependence on Medline is not strong; the system has been run successfully on other databases. The dependence on Data-Star is, however, quite strong, and the programs would require substantial development before they could be run on any other host.

The user drives the system by means of a simple command language. The next few sections constitute a users' manual, and include details of the command language and its usage. The weighting scheme used is defined in section 3.7, and some experiences with the use of cirt are briefly discussed in section 3.8.

3.2. Data structures.

The program maintains two main data structures, the current query and the results of the searches done so far, as well as lists of documents which you have seen and those which you have classed as relevant.

The current query is a list of query terms together with their frequencies and weights. As each new term is entered it is sent to Data-Star to determine its frequency. Then a weight is calculated from the frequency and any relevance information available and it is entered in the list. As far as possible the query list is kept in decreasing weight order but once a term has been used in a search it is not possible to

move it. Hence new terms added after a search has been done will be inserted after the searched terms, but the unsearched terms will still be kept in decreasing weight order.

The searches done are stored in the form of a tree which is updated and added to each time a new search is done. Each node of the tree represents a combination of terms for which a search has been done. It is not necessary to know about the search tree in order to do searches but it is printed on the terminal as the search is done. This gives some idea about how the search is going.

3.3. The Output File

cirt produces an output file into which are copied documents and summaries of searches. If you leave the program normally (using the Quit command) you will be asked for a name for this file but if the program crashes any output file produced so far will be found in /tmp/cirt.<user> where <user> is your login username followed by a random 3 digit number.

3.4. Command Syntax

All commands are in the form

```
command arg1 arg2 ....
```

where the command should be one of the ones listed below and the valid arguments depend on the command. Most commands and some arguments can be abbreviated by leaving out letters which are given in lower case. eg if a command is specified as **NewWeights** it can be abbreviated as 'nw' or 'neww' or 'nweights' or entered in full as 'newweights' but it cannot be abbreviated to 'n' or 'ne'.

In common with many UNIX programs, lines which start with ! are submitted to UNIX as commands.

3.5. Doing a search

In this section is a summary of how to do search using cirt. The individual commands are described in more detail in section 6.

3.5.1. Invoking cirt

cirt is invoked by typing cirt. It should respond with its prompt which is

```
->
```

and means it is ready for a command.

3.5.2. Calling up Data-Star

The command

Call dstar[e]

will connect you to Data-Star and leave you to log in in the usual way. After logging in you should press CONTROL-P to return to command mode so that you can enter a query. Alternatively you can use the command

LOGIN dstar[e]

which will call up Data-Star and then log you in using the userid, password and data base name taken from the file .loginn in your HOME catalogue. If it manages to log you in then the **LOGIN** command will return you to command mode. If for some reason it can't log you in it will leave you talking to Data-Star so that you can complete the job yourself. If 'dstare' is used rather than 'dstar' the Euronet address of Data-Star will be used. If you can't get through on one then try the other.

3.5.3. Building a query

The command to add terms to a query is

ADDterm term1 term2 ...

Terms can be deleted with the command

DELete term1 term2 ...

but terms which have been used in a search cannot be deleted. The current query and a summary of the results of any searches can be listed using the command

List

A "term" can be an ordinary natural language word which will then be searched as given. Alternatively, it can be any acceptable Data-Star search, such as "psycholog$6" (meaning the right-truncated root "psycholog" together with at most 6 more letters) or "jones-j.au." (meaning the name J.Jones in the author field). If the single Data-Star search contains blanks, it must be enclosed in double quotes - e.g.

ADDterm "labour or labor"

If right-truncation is used and more than 100 terms match, cirt will simply pass on the Data-Star error message generated.

3.5.4. Searching

The command to execute a search is simply

Search

The search is done using a backtracking algorithm and a summary of the results of the search is displayed as the search proceeds. For each individual search sent to Data-Star the number of documents retrieved

and their weight is displayed. This listing can be switched off using the command

SET -Verbose

but it is probably a good idea to leave it on as it gives a good idea of how the search is going. A search can be abandoned by pressing CONTROL-C and then can be finished by retyping the **Search** command.

cirt maintains an internal parameter called the search-size. When a search is done cirt guarantees to retrieve the search-size documents with highest weights. At present search-size defaults to 15 but it can be changed by using the command

SET SearchSize=n

The search procedure is set up in such a way that it never needs to resend any search statement that it has already sent to the Host. Hence if you type

Search

after just completing a search the program will simply display its search-tree on the terminal. If on the other hand a new search is done after executing some other commands then enough search statements will be sent to bring the search-tree up to date.

3.5.5. Printing documents on the terminal

The command to start printing documents on the terminal is

Print

The program will then display the retrieved documents on the terminal, starting with the one with highest weight which hasn't yet been seen. At present only the Author, Title and Abstract fields are displayed. At the end of each document cirt produces a ?? prompt in response to which you can make the following replies (each followed by carriage return).

p Print the document in the output file and then display the next document. All the paragraphs of the document are printed, not just the ones displayed at the terminal.

P The same as 'p' but instead of displaying the next document return to command mode (-> prompt).

r Print the next document and flag it as relevant to the query, then display the next document. Documents known as relevant can be used to recalculate the query term weights (see below).

R The same as 'r' but don't display the next document.

CR (That is just carriage-return) Display the next document but don't print the current one or flag it as relevant.

q (or Q) Return to command mode without printing the current document or flagging it as relevant.

Once a document has been seen it won't be displayed again whether or not it is printed or relevant. Normally only search-size documents will be displayed but cirt won't actualy stop displaying documents until it reaches one which has a weight which is strictly less than the search-size'th document.

3.5.6. Relevance feedback

After doing a search and looking at some documents, a few of which were deemed relevant, you can recalculate the weights using the command

NewWeights

3.5.7. Logging off from Data-Star

There is no command for leaving Data-Star. To logout tidily you should type the command:

Talk

which will connect you through to the host. Then you can type

..o [cont]

If you type the 'cont' your present state will be remembered for the next time you login.

3.6. The cirt Commands

In the next sections we describe each of the commands currently available with permissible arguments, options etc.

ADDterm [term [term ...]

Add term(s) to the query, and calculate weight(s).

BReak (no arguments)

Sends an interrupt to the host.

Call [host]

Attempts to call the host which should be the name of an X29 host which appears in the x25 directory. If you intended to do a retrieval search using the commands described below then host should be either dstar or dstare. If the call is successful you will be left talking to the host so that you can login, choose a

database, look at news bulletins etc. To get to command mode you should type CONTROL-P.

CLr (no arguments)

Clear down a network call. The call will be cleared and the cost, period of the call, and the number of 64 byte segments received and transmitted will be displayed. This command should be used if, for some reason, the network connection dies without clearing the call automatically. It is not the usual way of logging out from Data-Star (see Logging Off above).

DELete [term [term ...]

Delete term(s) from the query, provided you have not already done a **Search** since adding them.

LIMit (not yet available)

Submit a limiting query. All your searches will then be limited to the documents retrieved in this query.

List (no arguments)

List the current query and a summary of the search tree. The listing is pretty well self explanatory.

LOg [filename]

Start keeping a log of traffic between cirt and the host. If the filename is given then a file of that name is created for the log. If no filename is given a filename of 'netlog' is assumed. (Logging in 'netlog' is automatically initiated when cirt is first called.) If the filename is 'off' then any current log file is closed and logging is stopped. The log is stored in a binary form but it can be listed by using the (UNIX not cirt) command

listlog [filename]

Again the filename defaults to 'netlog' in the current directory.

LOGIN [+r] [+m] dstar[e]

This command calls up Data-Star and then logs you in. Your login name, password and data-base name are taken from the file '.loginn' in your HOME directory. If the option +m is given the message-of-the-day will be printed. If the +r option is used, and a restart is available, you will be given one; otherwise the option will be ignored. If for some reason the login fails but the call succeeds you will be left talking to Data-Star so that you can finish the job yourself.

NewWeights (no arguments)

New query weights are calculated using relevance feedback information obtained during a previous **Print** command. The program must have been told about some known relevant documents before you can use this command.

Print (no arguments)

Start the printing of documents at the terminal. Only the Title, Author and Abstract paragraphs are displayed on the terminal although the whole document is fetched from Data-Star. At the end of each document the program will print ?? and wait for a reply. The possible replies and their effects are listed in the following table.

Responses to the **Print** ?? prompt

Response	document sent to output file	document flagged as relevant	next document displayed
r	yes	yes	yes
R	yes	yes	no
p	yes	no	yes
P	yes	no	no
<CR>	no	no	yes
q or Q	no	no	no

All the paragraphs of a document are sent to the output file, not just those displayed on the terminal. Those documents flagged as relevant are remembered and used by the NewWeights command to calculate relevance feedback weights.

Quit (no arguments)

Leave the program and return to UNIX. Any call that is connected is automatically cleared. Before cirt finally quits it will print

Output file name -

to which you can reply with any valid UNIX file name or just carriage-return in which case the output file will be destroyed.

RESET (no arguments)

Clears all the internal data structures including the current query and the search-tree. **RESET** should be used before starting a completely new query.

Search (no arguments)

Use the current query to do a weighted search of the database. The search is done from the bottom up using a backtracking algorithm and the search is displayed on the terminal as it continues.

SET [SearchSize=n] [Universe=n] [(+|-)Verbose] [(+|-)Debug]

Used to set various internal parameters and switches. If no arguments are given the current values of the options are listed. **SearchSize** is the theoretical number of documents that the **Search** command will retrieve in its ranked list of best documents. It is the minimum number of documents that can be viewed using **Print** before another **Search** needs to be done. At present it defaults to 15. **Universe** is the assumed total size of the database. It is used to calculate the query-term weights but its value is not very critical. The default value is 600,000 which is about right for medline. **+Verbose** is the default option. The principal result of using the option **-Verbose** is to switch of the printing of the search-tree when doing a search. Setting the **+Debug** switch will result in the displaying of a lot of unintelligible material and is not recommended.

Talk (no arguments)

This command connects your terminal directly through to the Host enabling you to look at the news files etc. It is also the only way of logging off. **Talk** is called automatically after a successful **Call** and after a **LogIn** which succeeds in calling the Host but fails to login. To return to command mode you should press CONTROL-P.

3.7. The weighting scheme

The basic formula for collection-frequency weights (Sparck Jones, 1972) is:

$$w = \log(N/n)$$

where N is the number of documents in the collection and n is the number to which the particular term is assigned. The formula for relevance weights (Robertson and Sparck Jones, 1976) is:

$$w = \log \frac{(r+0.5)(N-R-n+r+0.5)}{(R-r+0.5)(n-r+0.5)}$$

where R is the number of known relevant documents and r is the number relevant to which the term is assigned. The +0.5 in each component is included for estimation reasons.

These two weights are associated (Croft and Harper, 1979). N is asssumed by the system to be fixed, and has been taken to be 600,000, about the size of medline (it can be reset by the searcher). However, there is some reason to suggest that a smaller value should be used (Harper, 1980).

3.8. Experiences

Although no formal experiments have been done with the system, it is appropriate to indicate some general impressions derived from informal testing.

As regards the technicalities, it is clear that the 8 term limit could be a major restricting factor, which might however be mitigated given other facilities. One obvious improvement would be to allow term deletion after searching, so that new terms could be substituted for bad ones. A second would be to allow new terms (e.g. synonyms) to be added in by ORing with existing terms.

Synonyms in fact raise quite general problems for such weighting schemes. If several synonyms (or near-synonyms) are included as separate terms in the list, then their combined weights may swamp other concepts represented by fewer synonyms. It has been suggested by some early trials that this may be a significant problem in achieving effective retrieval. One way to alleviate this problem would be, as indicated above, ORing the synonyms as a single search term. A second alleviating facility would be that of marking as non-relevant whole groups of documents that were retrieved by inappropriate combinations of terms, without displaying them individually. This latter suggestion would, however, have some theoretical drawbacks.