

A blue, wavy, fabric-like background with the text "Chapter 9" and "When words are important" centered in white.

Chapter 9

When words are important

AQUAD provides several functions that might help you in your analysis when you wish to look at your text from a linguistic angle, that is, when the words in your data are important. As mentioned earlier (in chapter 5, section 5.3), searching for words in your data text can be used to achieve semi-automatic coding. This is one of the five functions you may use in AQUAD to examine your data's linguistic properties:

- Searching for words in the data to guide coding
- Making a word list and counting specific words or word parts in all or selected data documents
- Counting specific word suffixes of words in all or selected data documents (using the word list)
- Searching for words (in the word list) in all or selected data documents and printing them out with their surrounding text (key-word-in-context or "KWIC" lists)
- Searching for memos by means of key words

9.1 Searching for words in the data texts

This option was explained earlier in chapter 6. The key to remember is the button "*Keywords*" in the field labeled "*Search*" on the right margin of the window for one-step coding. You get access to one-step coding from the sub-menu "*Coding*" in the main menu. While you are reading your data texts, you may wish to activate a search for key words.

You click on the button "*Keywords*" and a particular window pops up in the upper right corner. Here you enter the critical word or sequence of words. See chapter 6, section 6.6 for more details and a screen shot.

You start the retrieval by pushing the "*Find*" button, then proceeding with the "*Next*" button from location to location where the key word(s) are hidden in your text. Or you click on the "*All*" button, which displays the results of a retrieval run in one glimpse. Let us assume we go step by step ("*Find*" and repeatedly "*Next*"): In case the word is found, the text display changes so that the line containing the critical word is highlighted. Maybe you are wondering about the result, because there are some specific rules to remember:

- The search is case-sensitive, that is, AQUAD differentiates between capital letters and small letters when retrieving key words in your data text (as opposed to your memos!). If you try to find, for example, the word friend, the program will not find "Friend" at the beginning of a sentence (and vice versa).
- The retrieval algorithm is unable to put hyphenated words together again. If you expect that you will use word searches frequently in your analyses, you should switch off any automatic hyphenation when transcribing the texts.
- AQUAD runs a "wild card" search, locating your string of characters wherever it occurs, including embedded in other words. If you try to find, for example, a string "he was", the program will also stop and highlight a line containing "The wasting of ..."

This last feature deserves some comments. Let us assume you want to retrieve all text segments where something is said about "advantage" or "disadvantage." If you enter simply "dvantage" as criterion for a word search, AQUAD will find everything you are looking for in one run – given that the words really appear in your texts. Or another problem: You are searching for "friend" and "friends," but you do not want to have reported segments where something is characterized as "friendly" or as "friendship." So add a blank after the critical word (represented by "_" in the following examples) and have AQUAD retrieve text segments containing "friend_" and then in a second run "friends_". Now you will find only these complete words, however, not if they appear at the end of a sentence, followed by a period instead of a blank! Therefore, you should use the opportunity to create a whole retrieval dictionary, that is a word catalog (more about this in the next section).

9.2 Searching for key words within their context

For some research tasks it may be helpful to get a fast overview of words in the data that you assume are indicators of critical concepts. In fact, you might even want to use this feature to begin defining your codes. It might be important to know in which data file these words appear and in which context. Word frequencies alone may be insufficient, because often the specific meaning of a word can only be understood from its context. Imagine, you have counted the word "high", but the context of *high* may be in one case "During this time I always felt at high tension ... ," and in another case "After my friend left, I really was high and dry." More information about this procedure (key-word-in-context or "KWIC" lists) can be found in Weber (1985).

The example below shows the result of a "KWIC" search for the word "see" in the example file poet001. There is a column labeled "K" (for keywords) between line numbers and text columns. Wherever AQUAD retrieves the key word,

- a small red dot is put into the appropriate cell of the "K" column, and
- within the text line the character "□" is placed immediately in front of the key word:

33	The house, in which she lived, was small and clean,		X	X	
34	but it looked desolate.		X		
• 35	Not a single tree was to be □seen, no flower,		X		
36	in front of the house stood a beehive,				
37	very usefull!		X		
38	A small potato field was there,				
39	very usefull!		X		
40	And there was a trench with wild plums.				
41	They were already withered and had berries,		X		
42	which constrict your mouth,		X		
43	if you try to taste them.				
• 44	"What I □see here is our age void of poetry,		X	X	X
45	the very image of it!" thought the young man.				
46	Anyway, this was an idea, a grain of gold,		X	X	
47	which he found at the wise woman's doorstep.				

Where and how do you start a search with key words? You find the access to this function in the option "Retrieval"; there is a function "Keywords" (just type "k"). It starts by opening a window where you can choose between

- entering key words by typing them *or*
- selecting the name of a (key-) word catalog from a list of available catalogs (in case there are any!).

If you decide to type your key word(s), these words have to be typed anew whenever you activate this function. If you are sure to use a list of words for several retrieval runs, you should create a word catalog.

To do so you choose – also within the option "Retrieval" – from "Word catalog" the option "write a word catalog." Then you define a name for saving the catalog you are going to write. Just enter a name of up to eight characters; the extension ".cwo" (catalog of words) is added automatically. Then a window is opened into which you can write as many key words as you find useful for your retrieval run. Please, remember to put only one word or one sequence of related words on a line. You should be already familiar with creating catalogs from writing your first file catalog (see chapter 3).

In order to find the independent word "old" at the beginning of sentences, at the end of sentences, somewhere within sentences, but not as part of other words (for instance, "gold"), a word catalog should contain at least the following entries. In this catalog, the underlining characters "_" represent spaces in front of and after the key word – do not type this character instead of a blank (using the space bar on your keyboard) when creating a catalog!

```
_old_  
Old_  
_old,  
_old;  
_old.  
_old!  
_old?
```

Besides, the rules already mentioned in section 1 are valid here, too. Remember above all, that the search is case-sensitive! Once you have created a word catalog, you can use it again and again by simply selecting its name via the function "*Load a word catalog*".

A particularly interesting possibility to apply word catalogs should be mentioned: You can use a word catalog to define a dictionary of synonyms or a complete list of all concepts defining a particular field of meaning. Thus, you get an opportunity for semantic retrieval of critical text segments. Let us assume you want to find all text segments where your interview subjects made remarks about their social relations. You could type a word catalog containing all relevant elements of this field of meaning, for instance, friend(s), acquaintance, family, mother, father, brother(s), sister(s), partner, wife, husband, etc.

Beyond just retrieving important text segments without reading all the texts again, you may apply your findings in a time-saving strategy: Why not use them for *semi-automatic coding* (see also chapter 6, section 6.6)? Mark the unit of meaning that contains your keyword by clicking into the appropriate cell in the column "Line" and highlight the data segment – at the same time you will get the window for code entries on the screen additionally. If your text segment should comprise more than one line, keep the left mouse button pressed and move the cursor down the lines within this column as far as necessary. If you are not yet familiar with entering codes, read more about how to enter codes in one-step coding in chapter 6, section 6.5.1.

9.3 Searching for keywords in memos

The details are described in chapter 8, section 8.3. Keywords become relevant for the retrieval of memos, if you decide to use the "*Browse*" option.

9.4 Counting words

Leaving aside the controversy about using any quantitative means in a study where the data consist of narrative text, for some research purposes it may be useful to find out how frequently certain critical words appear. The count could tell the researcher, for instance, about the emphasis various people have placed on certain matters. Of course, any issue can be expressed with different words, and the researcher may want to conduct consecutive searches for several words all referring to the same issue. Further discussion about word frequencies can be found in Vorderer and Groeben (1987).

Before you can count the words, applying “*Count words*” from the option “*Retrieval*”, you have to create a catalog of words to be counted or to enter the name of an appropriate word catalog. The rules and conventions described above (see section 9.2) for keywords within the text are **invalid** here. When we count words, they are isolated from their context and counted just as they appear in the text. Therefore, it is a good idea to create *word catalogs for counting* by assembling a “*Select words from text*” (or several catalogs from several texts and “*Merge word catalogs*” afterwards).

The result of counting words is shown in a table with two columns: The left column shows the words in alphabetical order, followed by a number representing each word's frequency in the text; the right column shows the words ordered by frequency, with the number of appearances in front of each word. The following table shows part of the result of counting some words (listed in the sample catalog “negative.cwo” on your hard drive); the comparison of part 1 and 2 shows a big difference – to evaluate its relevance, we have to return to the text:

```
KEYWORDS in poet001.atx
|||===== poet001 =====|||
(control code "%do not count" was applied)

poet001: Alphabetical Order -----      Frequency order -----
desolate                1          1 desolate
emptied                 1          1 emptied
no                      3          1 void
not                     6          1 withered
small                   2          1 wobbly
void                    1          2 small
withered                1          3 no
wobbly                  1          6 not

Total:                  16
Different words:        8

|||===== poet002 =====|||

poet002: Alphabetical Order -----      Frequency order -----
not                     1          1 not
withered                1          1 withered

Total:                  2
Different words:        2
```

9.5 Counting suffixes

For specific questions, as for example when identifying cognitive styles from verbal productions of a person (see Günther, 1987), it may be useful to have a search and count algorithm which only considers suffixes (elements that are commonly added to the ends of words, such as "-ly", "-ed", "-ion", or "-ism"). You must make another word list in which you enter all the suffixes you wish to count. After that, you choose "count suffixes", which provides you with frequency results as well. If you search suffixes, do not forget to take into consideration the singular and plural forms of nouns.

9.6 How to exclude parts of a text from word analyses

In chapter 5 is described how parts of a text can be excluded from analysis. This possibility is really necessary when you analyze your texts on the level of single words. Otherwise, for example all questions or remarks of the interviewer would also be counted or retrieved – but you want to find out something about the interviewees only.

Here we show a tiny text example from a study on teacher-student interactions, which demonstrates how everything mentioned by a teacher during a particular interaction sequence is blocked for analysis. To all text segments containing the teacher's utterances the control code "\$do not count" is attached. Thus they become excluded for AQUAD. Utterances of students are marked not marked by this code, thus they are open for analyses:

```
54 ...
55 TCH How many balls can we put           $do not count    55 - 56
56   on the scale additionally?
57 STD Hm?
58 TCH How many more balls could we compare? $do not count    58 - 59
59   How many balls on each side? ...
60 STD Two -- three -- and four, or one ball ...
61 TCH And what is optimal?                 $do not count    61 - 61
62 STD Four
63 TCH Why?                                 $do not count    63 - 63
64 ...
```

Due to AQUAD's master code file, these codes are just a mouse click away from your code files; you do not have to type them again and again. If you want to remove them, this can also be done very easily by deleting them. For more details see chapter 6 on coding.