

KorpusDK Cheat Sheet

Anders K. Madsen

v. 1.0 rev. 3 — March 1, 2009

License: [Creative Commons BY-NC-SA](#)

KorpusDK uses a query language called CQP (Corpus Query Processor) that has a special syntax allowing for very narrow and complex queries. CQP shares a lot of its syntax with *regular expressions*^a known from programming languages such as Perl, Ruby, PHP, Python etc. CQP and regular expressions have quite a steep learning curve, but once learned they turn out to be incredibly useful for digging through texts.

This cheat sheet is meant as a crash course in CQP and is by no means exhaustive.

Searching KorpusDK using CQP is done by selecting *Formel søgning* (formal search) in KorpusDK.

^aSearch for “[regular expressions cheat sheet](#)” on Google.

1 Specific words and word forms

Searching for specific words (or parts of words) is done with the keywords `lemma`, `word` and `ortho`.

lemma: Given the uninflected stem of a word, the `lemma` keyword will search for all inflected and uninflected forms of the word.

word: Searches the corpus for the given form ignoring case and diacritics (and apparently, to some extent, form).

ortho: Finds exactly the given form in the corpus — both case and diacritics are taken into account.

To search for all (inflected/uninflected) forms of *gå*:

(1) `[lemma="gå"]`

To search for the specific form (case insensitive and not paying attention to diacritics):

(2) `[word="alle"]`

To search for only the specified form (case sensitive and paying attention to diacritics):

(3) `[ortho="Allé"]`

To find all occurrences of “gå amok” (including “går amok” and “gik amok”):

(4) `[lemma="gå"] [word="amok"]`

2 Parts of Speech

When searching for specific parts of speech, the `pos` keyword is used.

Possible values for the <code>pos</code> keyword			
N	noun	INDP	other pronoun
V	verb	KC	coordinating conj.
ADJ	adjective	KS	subordinating conj.
ADV	adverb (see KorpusDK)	IN	interjection
PROP	proper names	INFM	infinitive marker
PRP	preposition	NUM	numeral
DET	determiner	PREF	prefix
ART	article	USPEC	unspecified
PERS	personal pronoun		

Find all adjectives:

(5) `[pos="ADJ"]`

Find all nouns preceded by an adjective:

(6) `[pos="ADJ"] [pos="N"]`

The `pos` keyword can even be combined with any of `lemma`, `word` and `ortho` in order to narrow the search down, so to find all occurrences of the *skade* occurring as a verb:

(7) `[lemma="skade" & pos="V"]`

Or to find only the present tense of *skade* — *skader*:

(8) `[word="skader" & pos="V"]`

3 Wildcards, optionality and other stunts

Any text search is useless without wildcards and fortunately CQP has great support for those. In CQP wildcards usually consists of a match expression (specifying *what* you want to find) immediately followed by a quantifier (specifying *how much* of it you want to find).

Match expressions	
.	Matches any character.
[abc]	Matches any of <i>a</i> , <i>b</i> or <i>c</i> .
[^abc]	Matches any character that is not <i>a</i> , <i>b</i> or <i>c</i> .
[0-9]	Matches any character between 0 and 9.
p	Matches <i>p</i> .
(en et)	Matches <i>en</i> or <i>et</i> .
Quantifiers	
?	Matches 0 or 1 of the preceding expression.
*	Matches 0 or more of the preceding expression.
+	Matches 1 or more of the preceding expression.
{2,5}	Matches 2 to 5 of the preceding expression.
{2,}	Matches 2 or more of the preceding expression.
{3}	Matches exactly 3 of the preceding expression.

Common combinations are:

- .* : Any number (including 0) of any characters.
- .+ : 1 or more of any characters.
- s? : 0 or 1 of *s*.

To find all words starting with *miljø* followed by 0 or more letters (ie. including the form “*miljø*”):

(9) `[word="miljø.*"]`

To find all words starting with *miljø* followed by at least one letter (excluding the form “*miljø*”):

(10) `[word="miljø.+"]`

To find all words ending in *-en* eller *-et*:

(11) `[word=".(en|et)"]`

The quantifiers even work outside the keyword values, so it is possible to make search expressions like “a determiner followed by 2 or more adjectives followed by a noun”, which in CQP looks like this:

(12) `[pos="DET"] ([pos="ADJ"]{2,}) [pos="N"]`

One may — for various reasons — wish to modify ex. 12 to only allow for a single optional adjective, in which case the ? quantifier should be used:

(13) `[pos="DET"] ([pos="ADJ"]?) [pos="N"]`

Ex. 12 can be modified even further to only search for a determiner followed by a single optional adjective ending in *-sk* followed by a noun:

(14) `[pos="DET"] ([word=".+sk" & pos="ADJ"]?) [pos="N"]`

4 Misc.

Find all occurrences of *af* followed by 1–3 words of any kind followed by *søster*:

(15) `[word="af"] ([]{1,3}) [word="søster"]`

Find all words starting with *p* and ending in *s* with anything but *i* or *u* in between:

(16) `[word="p[^iu]s"]`

Find all words containing a consonant cluster of 5 or more:

(17) `[word=".*[^aeiouyæøå0-9\-\.] {5,}.*"]`

Find all words with a cluster of exactly 5 vowels:

(18) `[word=".*[aeiouyæøå]{5}.*"]`

For more details, see [the documentation on KorpusDK](#).