Brieven als Buit-2; application manual

Table of contents

Introduction

This manual describes the corpus exploitation environment for the *Brieven als Buit-2* ('Letters as loot 2') corpus. The corpus application is developed by the INT. The backend of the application is the BlackLab Lucene based search engine developed for corpora with token-based annotation (http://inl.github.io/BlackLab/). The web-based frontend is a further development of the corpus-frontend application developed by INT (https://github.com/INL/corpus-frontend) in CLARIN and CLARIAH projects. Its design is inspired by the first version of the OpenSoNaR user interface by Tilburg and Radboud University (https://github.com/Taalmonsters/WhiteLab2.0).

Information about the corpus

Brieven als Buit-2 is a spin-off of the research programme *Brieven als Buit / Letters as Loot. Towards a non-standard view on the history of Dutch* (see www.brievenalsbuit.nl). This programme, initiated and directed by prof. dr. Marijke van der Wal (Leiden University) and funded by the Netherlands Organisation for Scientific Research (NWO), successfully ran from 1 September 2008 till 1 September 2013.

Just as the earlier *Brieven als Buit* internet application, the present *Brieven als Buit-2* comprises Dutch letters which were taken as loot by privateers and confiscated by the High Court of Admiralty during the wars fought between The Netherlands and England from the second half of the 17th to the early 19th centuries. The National Archives (Kew, UK), the current location of the confiscated documents, offered the opportunity to photograph selected letters, all kept in the High Court of Admiralty (HCA) Archives.

The letters were sent home by sailors and others from abroad but also vice versa by those staying behind who needed to keep in touch with their loved ones. These confiscated letters of men, women and even children represent priceless material for historical linguists. They allow us to gain access to the everyday Dutch of the past, the colloquial Dutch of people from the middle and lower classes.

All texts in *Brieven als Buit-2* were tokenised, but unlike the earlier *Brieven als Buit* corpus, the tokens are not tagged with part of speech and are not lemmatized. It is therefore only possible to search for word forms.

This first online accessible version of the corpus *Brieven als Buit-2* was released on 26 February 2021.

GiGaNT Lexicon service

Contrary to the earlier *Brieven als Buit* internet application, *Brieven als Buit-2* has not yet been annotated with part of speech and lemma. To make the corpus more accessible, suggestions for query expansion are given, using the INT lexicon service with the historical computational lexicon GiGaNT-HILEX.

The current version of GiGaNT-HILEX in the lexicon service contains the lexicon modules based on the *Dictionary of the Dutch Language* and the *Dictionary of Middle Dutch*.

If you want to make use of this service, please contact Katrien Depuydt (katrien.depuydt@ivdnt.org).

Metadata categories

The *Brieven als Buit* corpus has been enriched with an elaborate set of metadata categories. These metadata will all be described below. In the corpus application it is possible to limit a search by filtering on metadata categories.

Letter

A letter from the *Brieven als Buit-2* corpus.

Year

The year(s) in which the letter(s) was (were) written, as evidenced by the date of the letter.

Text type

The type of text to which the letter belongs (business, private, combination). Compared to the original internet application, *Brieven als Buit-2* contains more business letters or letters of a mixed private-business character.

Autograph

In the case of autographs (autograph), it has been established that the sender actually wrote the letter. In the case of non-autographs (non-autograph) the sender did not write the letter himself, but had it done by someone else. In uncertain cases it was not completely certain whereas in unknown cases it could not be determined at all whether the letter is an autograph or not.

Signature

The signature of the box containing letters in the archives of the High Court of Admiralty (HCA) in the National Archives in Kew, United Kingdom. A signature always starts with the letters HCA followed by a series of numbers, for example HCA 30-223.

Sender

The person who sent the letter. Please note that this is not always the actual scribe of the letter (see under Autograph).

Name

The name of the sender.

Gender

The gender of the sender (female, male, unknown).

Class

Four social layers are distinguished, based on the stratification that is common among historians (see Willem Frijhoff & Marijke Spies, 1650. *Bevochten eendracht*. Den Haag: Sdu, 1999, pp. 188-190). The four social segments of the application are: low class, middle-low class, middle-high class and high class.

The low class includes, for example, seafarers from the lowest ranks, servants, soldiers and the poor (11 documents). The middle-low class includes small shopkeepers, small farmers, seafarers from lower ranks and craftsmen (78 documents). The middle-high class consists of, for example, small businessmen, wealthy farmers, master craftsmen, captains and lower ranking officers such as mates (333 documents). The high class includes wealthy merchants, ship owners, academics, senior officials and senior officers in the army and navy (277 documents). It should be noted that high class does not refer to the highest social class of nobility and non-noble ruling classes. That upper class does not occur in the corpus. There are 687 documents of which it is not known to which class the sender belonged.

Age

Three age groups are distinguished, namely <30 (under 30), 30-50 (30 to 50 years) and >50 (over 50). There are 623 letters for which the age of the sender has not been determined (unknown).

Region of residence

This designation refers to the region where a sender grew up or where he or she spent most of his or her life. The region of residence is usually a Dutch province such as Zeeland or Zuid-Holland. The corpus contains many letters from the Caribbean, sent by people who temporarily or for a longer period of time stayed in the Caribbean region, but who originally came from Zeeland or Zuid-Holland. The region of residence is then Zeeland or Zuid-Holland and not, for example, Curaçao. This characteristic is important for linguistic research into dialect differences.

Note that for Noord-Holland, a distinction is made between Noord-Holland, Amsterdam - a city that was a metropolis at the time - and Noord-Holland (excluding Amsterdam).

Relationship to addressee

The relationship, personal and/or professional, that the sender has with the addressee, such as acquaintance, employer, friend, grandson, mother, nephew/cousin or sister.

Addressee

The person to whom the letter was sent.

Name

The name of the consignee.

Place

The place to which a letter was sent, e.g. Enkhuizen or Middelburg.

Country

The country to which a letter has been sent. Note that contemporary names are used, for example Sri Lanka (and not Ceylon) and Saint Kitts (and not St. Christopher).

Region

Within the Dutch language area, the term region refers to a province or dialect area, such as Noord-Brabant, West-Vlaanderen and Zeeland. Outside the Dutch language area, it indicates a geographical region: Azië (Asia), Caraïbisch gebied (Caribbean), Noord-Europa (Northern Europe), West-Afrika (West Africa) and Zuid-Europa (Southern Europe).

Ship

The ship to which a letter was sent, e.g. the *Spiegel*.

Sent from

The four place designations that are distinguished in this tab (Place, Country, Region, Ship) have already been described above at Addressee. Please note that the names of places, countries, regions and ships are different from those found at Addressee.

Application user manual

Getting started

Here are a few examples of what you can do with the corpus application (the links will take you to the application):

- To search for a word literally in the form you specify, use Simple Search or the attribute Word in Extended Search:
  - Simple Search for Word *scepe*
  - Extended Search for Word *dochter*
- To search for words satisfying a certain pattern, use *wildcards* in Simple Search or Extended Search, or *regular expressions* in Expert Search
  - words starting with *ver* and ending with *len* in Simple Search and Extended Search
  - lemmata starting with *ver* and ending in *eren* with one syllable in between in Expert Search
- To see which unique forms occur as a result of your search, use the Group hits by feature.
  - example Group by Context (advanced): all words following *lieve*
  - example Group by Word before: different words preceding the word *huis*
- To explore the distribution of document properties in the corpus, use the Explore feature
  - example: characteristics about the signature
  - example: speaker age distribution

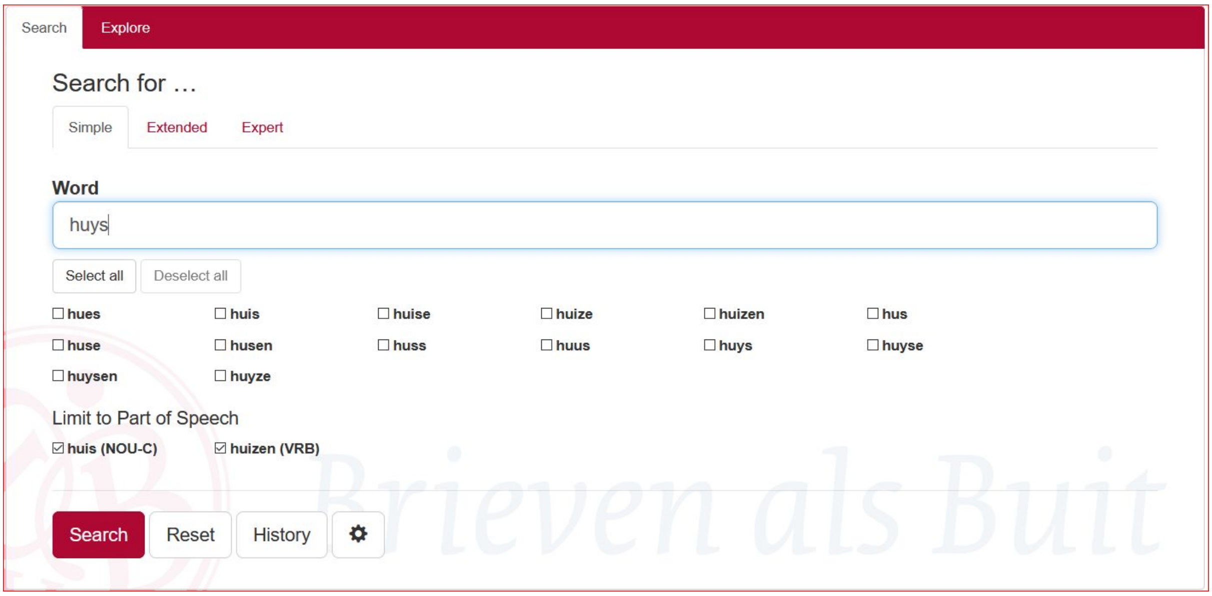
Searching the corpus

Simple search

Search

The Simple Search allows you to quickly search for specific word forms (e.g. *huys*). After entering a search term, a spinner briefly appears on the right side of the search bar. Based on the keyed in word, suggestions are given of possible variants of spelling and/or form from the GiGaNT-lexicon.

Based on the information in this lexicon all spelling variants of the search term found are suggested (see screenshot below). You can then choose from the presented suggestions or select all at the same time (Select all). To make your search even more targeted, it is also possible to limit the search to the parts of speech that were found in GiGaNT-HILEX in connection to the search term.



Word

huys

□ hues □ huis □ huise □ huize □ huizen □ hus
□ huse □ husen □ huss □ huus □ huys □ huyse
□ huysen □ huyze

Limit to Part of Speech

☑ huis (NOU-C) ☑ huizen (VRB)

If you know exactly which word you're looking for, you can also - while the wheel is spinning - press Enter directly. The search will then start immediately.

It is also possible to enter a phrase: *voor eeniege dagen* or *watt myn aangaat*. You will then find all occurrences of that exact phrase.

Note that in Simple Search the patterns will be matched case-insensitively**:** *capitein* for instance will deliver the same results as *Capitein* or *CAPITEIN*. See the paragraph Grouping results in Per Hit view to see how it is nevertheless possible to distinguish between uppercase and lowercase letters.

Wildcards

In Simple Search, the use of wildcards can prove good service to search for specific word forms or lemmata. A wildcard is a symbol used to replace or represent one or more characters. The following two wildcards are supported:

* The asterisk matches any character zero or more times. Therefore, *a*n* matches all values that start with an *a* and end with a *n*, e.g. *aan*, *adverteeren* and *alschoon*.

? The question mark matches a single character once. Therefore, searching for *a?n* matches *only* three-letter values starting with an *a* and ending with a *n*, e.g. *aan, aen, ahn, ann, adn* en *aen*.

This wildcard can be used more than once. Thus *a???n* matches words like *armen, allen, abben* en *agten*. Note that searching with wildcards is limited to Simple Search and Extended Search. [In Expert Search you can use so-called regular expressions instead of wildcards.]

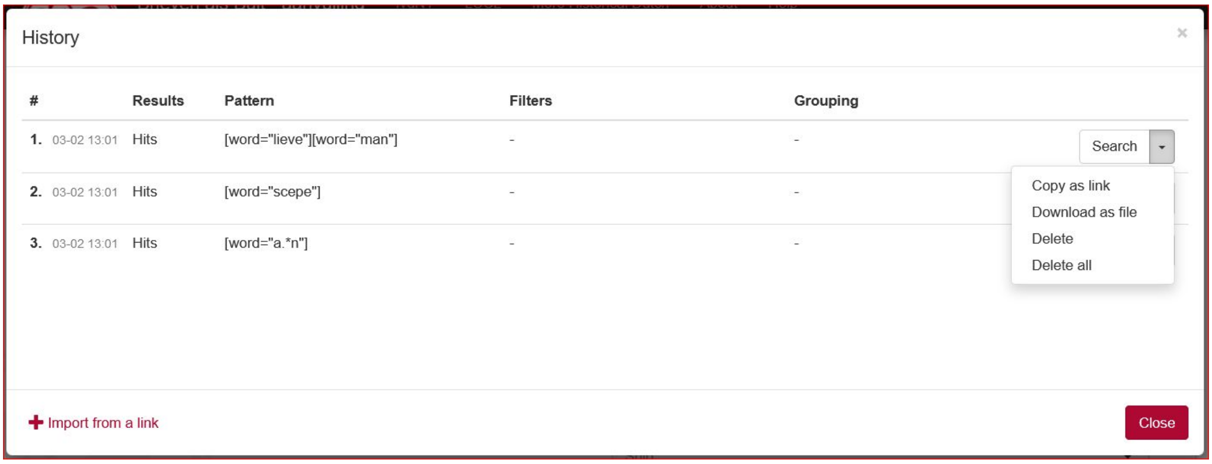
Reset

You can start a new search by pressing the Reset button. By doing so, both the search query and the hits found will be cleared. Your search history, however, will remain unchanged.

Note that it is also possible to start a new search by entering a new word or phrase in the search field Word.

History

The History button will display your query history. Per search query there are several possibilities (as shown in the screenshot below): you can perform the search query again (Search), you can copy the search query as a link (Copy as link), you can download the search query as a file (Download as file), you can delete a single search query (Delete) or delete all search queries (Delete all).



History

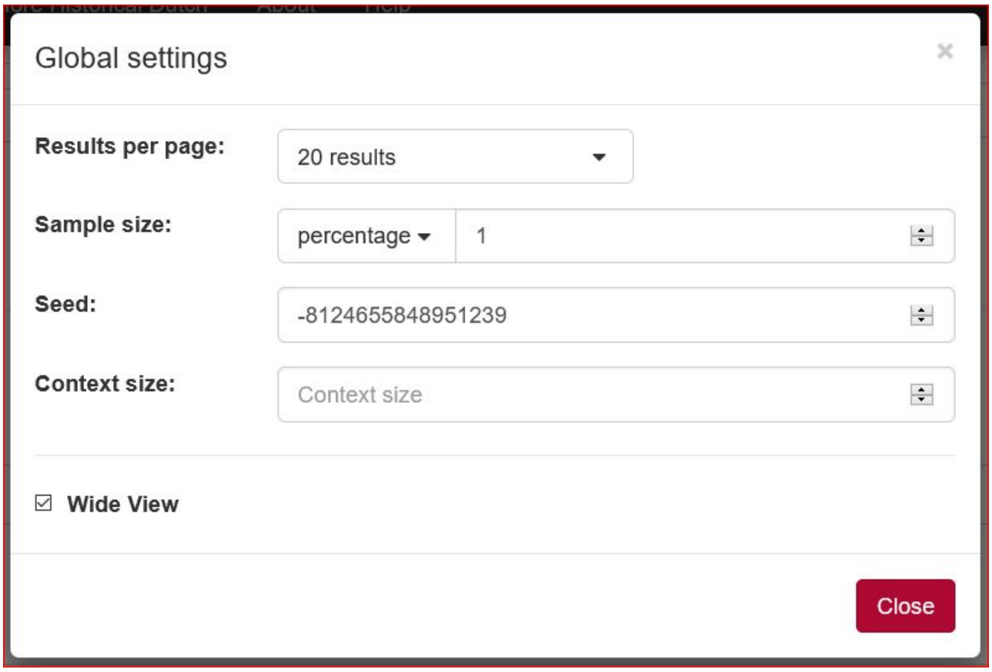
#		Results	Pattern	Filters	Grouping
1.	03-02 13:01	Hits	[word="lieve"][word="man"]	-	-
2.	03-02 13:01	Hits	[word="scepe"]	-	-
3.	03-02 13:01	Hits	[word="a.*n"]	-	-

Every search query has its own url. If you copy this url via History (Copy as link) or directly from the address bar of your browser, you can send it to someone else who can import this link via Import from a link. It offers that person the possibility to run the search on his or her own computer.

Global settings

The Global settings dialogue, activated by pressing the wheel button, allows you to configure five settings: Results per page, Sample size, Seed, Context size and Wide View.

- *Results per page*: you can choose whether you want 20, 50, 100 or 200 results to be shown;
- *Sample size:* selecting a value here will instruct the search engine to return a random sample drawn from the complete result set. (Pressing the Reset button does not change the sample size. It must be changed manually.) The sample size can be limited by
  - a percentage of the total number of search results (percentage)
  - the number of results displayed (count);
- *Seed:* a 'random seed' is a number used to initialize a so-called pseudo-random number generator. Keeping the same seed will ensure that two samples drawn from the same result set are identical. A new seed will most likely result in a different sample;
- *Context size*: by entering a number you can determine the number of words Before hit and After hit;
- *Wide View:* the default setting is 'small view'; you can change to Wide View by ticking the checkbox.



Global settings

Results per page: 20 results

Sample size: percentage 1

Seed: -8124655848951239

Context size: Context size

☑ Wide View

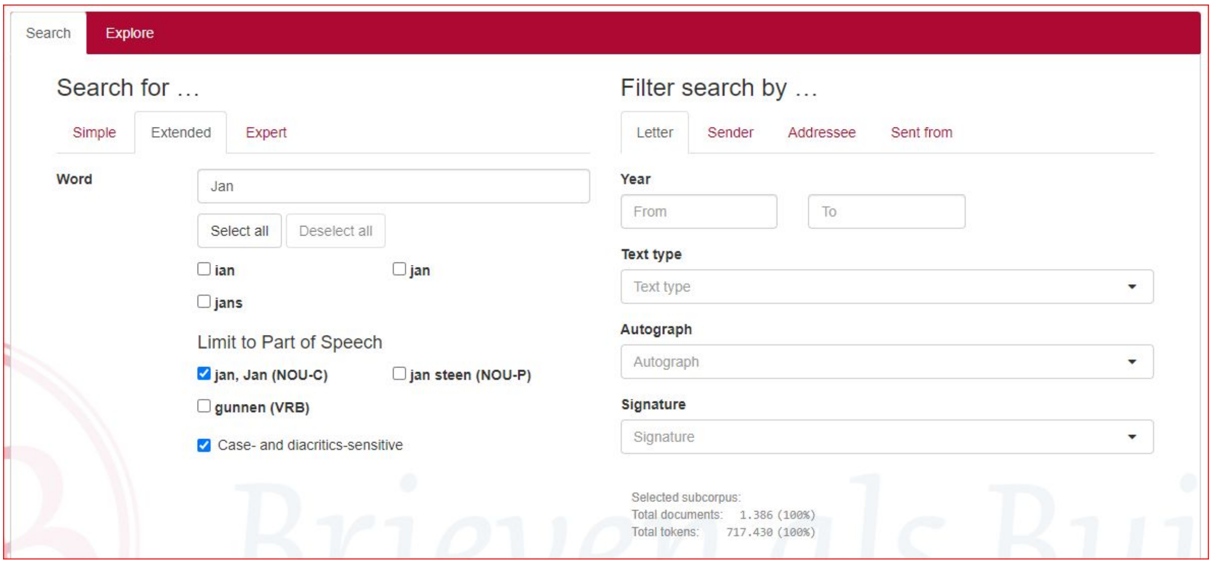
Extended search

Like in Simple search Extended Search allows you to quickly search for specific word forms (e.g. *huys*). In addition, Extended search offers some extra possibilities to refine your search.

After entering a search term, a spinner briefly appears on the right side of the search bar. Based on the keyed in word, suggestions are given of possible variants of spelling and/or form from the GiGaNT-lexicon.

Based on the information in this lexicon all spelling variants of the search term found are suggested (see screenshot below). You can then choose from the presented suggestions or select all at the same time (Select all). To make your search even more targeted, it is also possible to limit the search to certain parts of speech that were found in GiGaNT-HILEX in connection to the search term. It is also possible to enter a phrase: *die daar toe gebruykt zoude worde* or *zal gehad hebbe*.

In Extended Search it is also possible to search case- and diacritics-sensitive. Note that the default setting for search is case- and diacritics-insensitive. For example, searching for the Word *jan* (& jan, Jan (NOU-C)) will result in 1026 occurrences of this name. By ticking the box Case- and diacritics-sensitive you will only find the Word *jan* (149x), but not the variant *Jan.* In order to directly find only occurrences of the Word (form) *Jan* (877x), tick the box Case- and diacritics-sensitive under the search field Word (as shown below).



Word: Jan

☐ ian ☐ jan
☐ jans

Limit to Part of Speech

☑ jan, Jan (NOU-C) ☐ jan steen (NOU-P)
☐ gunnen (VRB)

☑ Case- and diacritics-sensitive

Year: From To

Text type: Text type

Autograph: Autograph

Signature: Signature

Selected subcorpus:
Total documents: 1.386 (100%)
Total tokens: 717.430 (100%)

If you know exactly which word you're looking for, you can also - while the wheel is spinning - press Enter directly. The search will then start immediately.

Like in Simple Search, wildcards are supported in Extended Search. (See for a short explanation of wildcards Simple Search.)

In the search field Word it is possible to search for different values simultaneously by separating them without spaces by a vertical line, e.g. *god|man|lief* or - with the use of wildcards - *god|aan**|*hond*.

For the search field Word it is possible to search for a series of tokens by entering multiple values - including wildcards - separated by a space, e.g. *liefste suster, liefste* * or * *suster.*

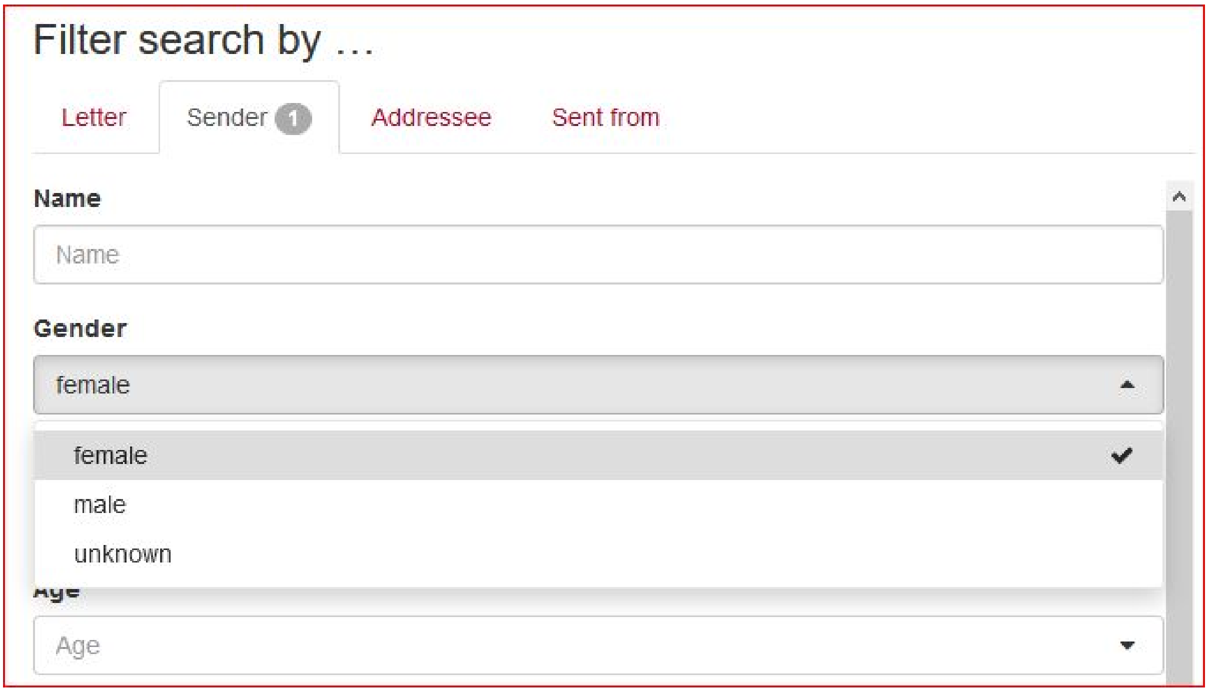
Starting a new search

You can start a new search by pressing the Reset button. By doing so, both the search query and the hits found will disappear. Your search history, however, will remain unchanged.

If you use the field Word, there are two possibilities to start a search: fill in the desired value and press enter or fill in the desired value and then click the Search button. Filter search by

At the right side you will find the option to limit your query to a subset of documents with specific metadata values. You can apply different filters for Letter (*Year, Text type, Autograph, Signature*), Sender (*Name, Gender, Class, Age, Region of residence, Relationship to addressee*), Addressee (*Name, Place, Country, Region, Ship*) and Sent from (*Place, Country, Region, Ship*). (To view the results for all documents simply leave the attributes in the filtering form empty.)

There are two different ways to specify a filter, depending on the field type. You can either fill in a value yourself - for instance Sender Name - or choose one or more values from a drop-down list - for instance Class. The drop-down list has been applied especially when the number of values to choose from is relatively small. Sender Gender for instance has only three possibilities (female, male and unknown). You can pick one of these values by clicking on it; your choice will be marked with a tick. It is possible to choose several values. If you want to delete a selection, you can click on the corresponding line again. (To close the drop-down list, you can either press the upward pointing arrow in the upper right corner or simply press escape.)



Filter search by …

Letter | Sender 1 | Addressee | Sent from

Name: Name

Gender: female

When on the other hand the set of possible values is rather large (e.g. Addressee Name), you have to type a specific value in the search field. After entering a single character, a list of possible values is suggested. Clicking on an auto-completed value will paste that value in the field. Note that this only works with a single word, like *abraham*. In order to search for an exact phrase, i.e. a multiple word value, it must be surrounded by double quotes. For instance, in the field Sender: Name *"abraham bor"* will result in a single letter sent by him.

By means of a number at the top of 'Filter search by', the number of values used to filter on, is displayed as can be seen in the above screenshot.

For a detailed description of the metadata, see the section Metadata categories.

Expert search

The Corpus Query Language (CQL) editor allows you to type your own CQL query, to import a previously downloaded query and to upload a tab separated list of values to substitute for gap values (see below for further explanation).

CQL queries are expressions built up with the help of a few sequence operators and brackets from basic blocks enclosed by square brackets, in each of which one or more token attributes are specified.

In CQL, spaces only affect a search if they are included in quotes. Whether the search command is [word="schip"] or [word = "schip"] does not make any difference to the result. However, there is a difference between the queries [word="schip"] and [word=" schip"]. The first search results in exactly 13 hits, but the second one in zero!

Some examples:

- Simple: `[word="schip"]`, e.g. the attribute word matches the regular expression *schip*; `[word!="schip"]`, e.g. the attribute word does **not** match the regular expression *schip*; `[word=".*man"]` matches all words ending with *man*, including *man* itself. (Note that [word="*man"] will not give any results, because in Expert Search an asterisk is not a wildcard but a repetition operator.)

- Combination of attributes (combining operators are &, |, !), e.g. [word="hoop"|"geloof"|"liefde"] matches either the word *geloof*, the word *hoop* or the word *liefde*.
- The empty [] matches any token, e.g. [word="man"][]{3}[word="god"] matches a sequence of *man* followed by *god* with three arbitrary tokens in between.
- Operators |, & and parentheses () and the repetition operators (+, *, ? and {}) can be used to build complex sequence queries. Example: "lieue" "man" | "almagtige" "god", or even ("lieue" "man" | "almagtige" "god")+, matching any sequence of *lieue man* or *almagtige god*.

This short list does not cover all CQL features. For more detailed information on how to write CQL, please consult the short CQL manual in the appendix, which contains further pointers.

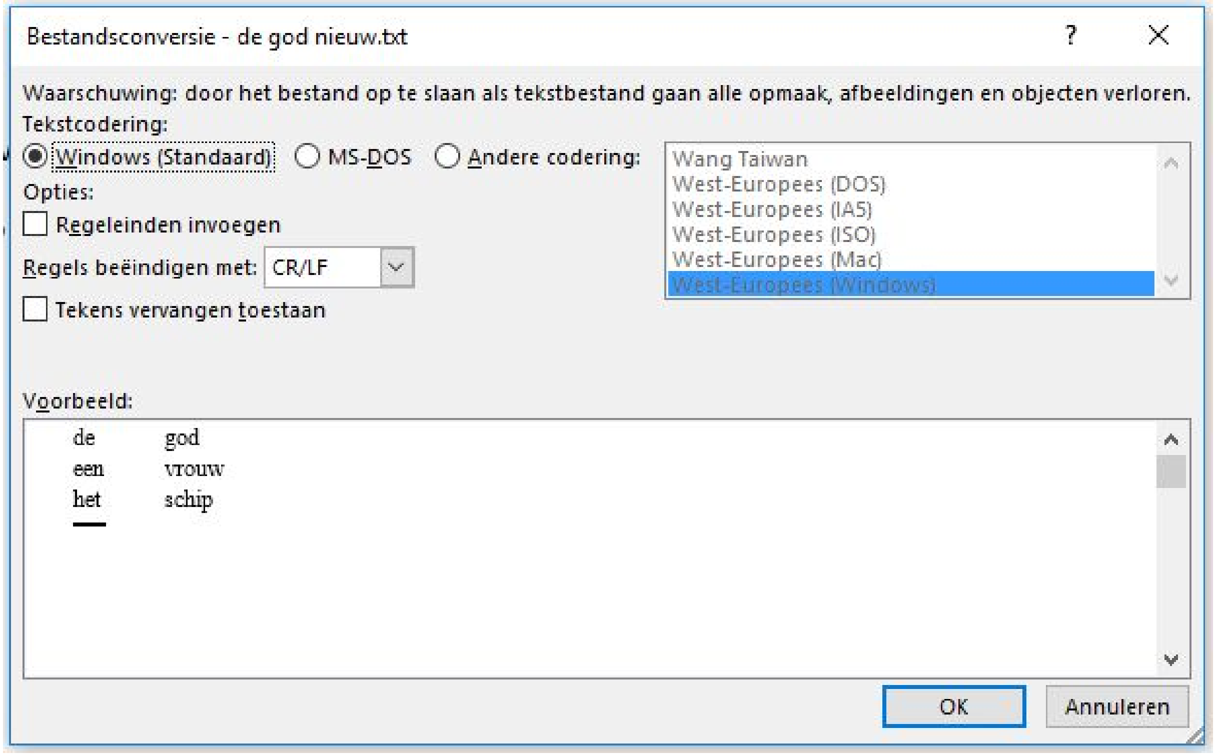
Import query

If you have entered a search query, you can find it back by clicking the History button. On the right hand side you can select Download as file in the drop-down menu (default value is Search) and afterwards save the file. (For a more elaborate description of the History button see Simple Search)

Previously saved queries can be used again by uploading them through the Import query button.

Gap filling

Use this button to upload a Tab Separated Values (TSV) file, which is a simple text format for storing data in a tabular structure. Each record in the table is one line of the text file. Each field value of a record is separated from the next by a tab character. It is also possible to upload a plain text file (.txt) that has the same properties, as is shown in the following screenshot:



Waarschuwing: door het bestand op te slaan als tekstbestand gaan alle opmaak, afbeeldingen en objecten verloren.

Tekstcodering:

● Windows (Standaard) ○ MS-DOS ○ Andere codering:

Opties:

□ Regeleinden invoegen

Regels beëindigen met: CR/LF

□ Tekens vervangen toestaan

Voorbeeld:

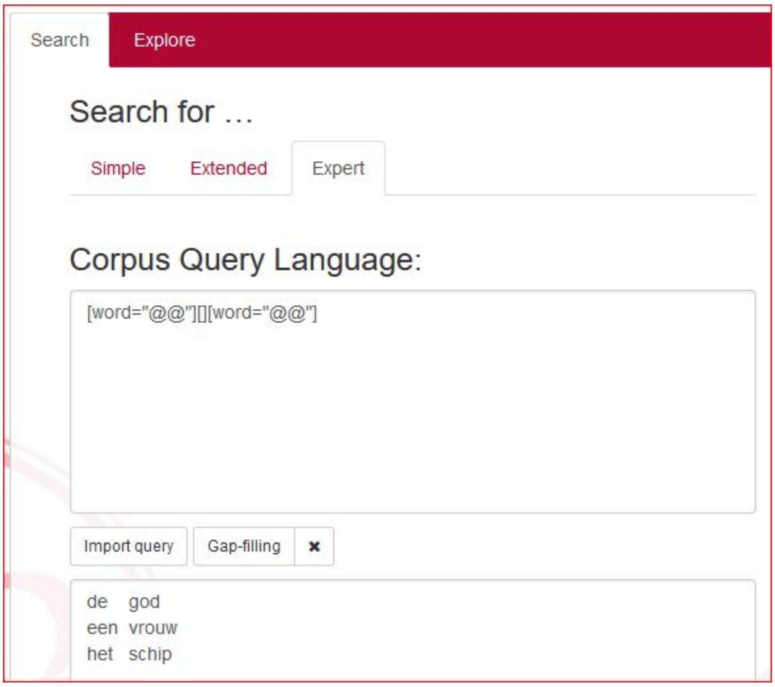
```
de	god
een	vrouw
het	schip
```

A .tsv file or a comparable .txt file enables you to complete a query with marked gaps.

If, for instance, you are interested in the distribution of words that can be placed between two specific words you can create this query in the Corpus Query Language field:

```
[word="@@"][][word="@@"]
```

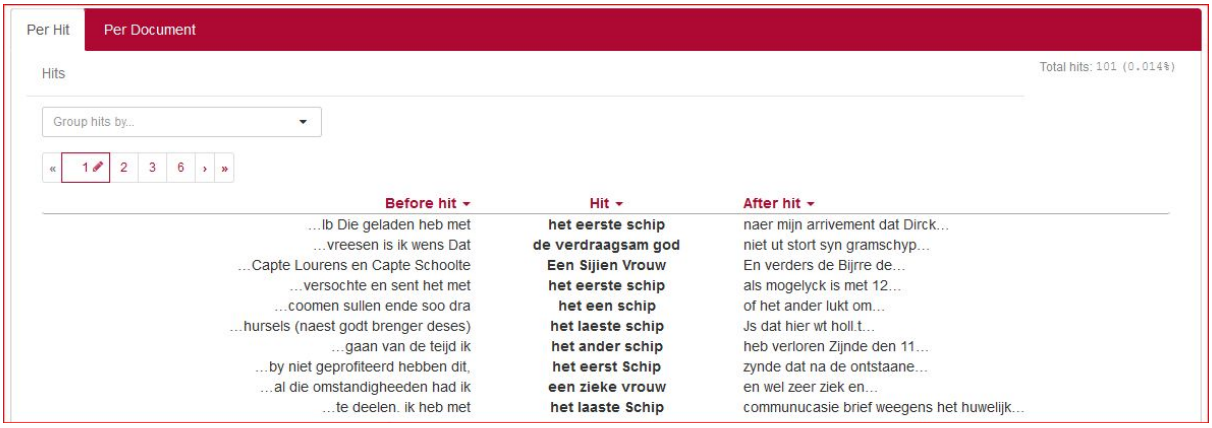
By clicking Gap-filling you can upload a file with a tab-separated list of values from your computer to substitute them for the gap values, i.e. the at signs (@@) in your query. After the upload your values will appear in a separate box:



```
[word="@@"][][word="@@"]
```

de	god
een	vrouw
het	schip

The values in the first column - *de, een, het* - will be entered at the position of the first gap (@@) and the values in the second column - *god, vrouw, schip* - at the position of the second gap. With these values, gap-filling yields the following results (titles are hidden):



Before hit	Hit	After hit
...Ib Die geladen heb met	het eerste schip	naer mijn arrivement dat Dirck...
...vreesen is ik wens Dat	de verdraagsam god	niet ut stort syn gramschyp...
...Capte Lourens en Capte Schoolte	Een Sijien Vrouw	En verders de Bijrre de...
...versochte en sent het met	het eerste schip	als mogelyck is met 12...
...coomen sullen ende soo dra	het een schip	of het ander lukt om...
...hursels (naest godt brenger deses)	het laeste schip	Js dat hier wt holl.t...
...gaan van de teijd ik	het ander schip	heb verloren Zijnde den 11...
...by niet geprofiteerd hebben dit,	het eerst Schip	zynde dat na de ontstaane...
...al die omstandigheeden had ik	een zieke vrouw	en wel zeer ziek en...
...te deelen. ik heb met	het laaste Schip	communucasie brief weegens het huwelijk...

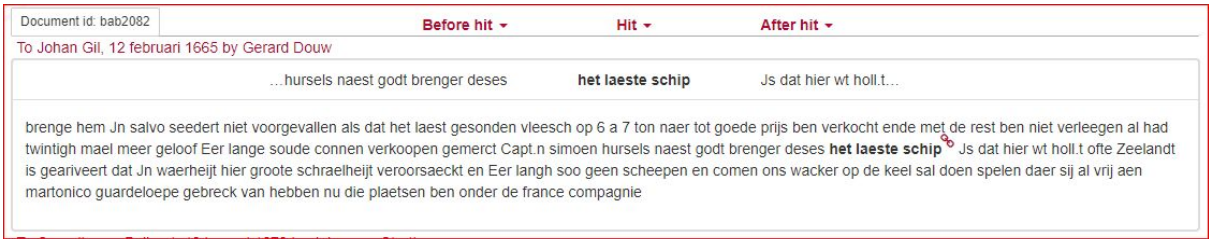
Please note that for this to work, you do need to enter @@ in the field where you want the substitution to take place. An empty field ([]) will match any term.

Viewing results

Results can be viewed in two ways: Per hit (hit is defined as one token or a group of tokens that matched the query), or Per document (each document listed contains at least one hit).

Per Hit view

Click a hit - i.e. a line with the bold words in the column Hit - to display the properties and values of the hit (in the following example **het laeste schip)**. Click the hit again to close.



Document id: bab2082

To Johan Gil, 12 februari 1665 by Gerard Douw

Before hit	Hit	After hit
...hursels naest godt brenger deses	**het laeste schip**	Js dat hier wt holl.t...

brenge hem Jn salvo seedert niet voorgevallen als dat het laest gesonden vleesch op 6 a 7 ton naer tot goede prijs ben verkocht ende met de rest ben niet verleegen al had twintigh mael meer geloof Eer lange soude connen verkoopen gemerct Capt.n simoen hursels naest godt brenger deses **het laeste schip** Js dat hier wt holl.t ofte Zeelandt is geariveert dat Jn waerheijt hier groote schraelheijt veroorsaeckt en Eer langh soo geen scheepen en comen ons wacker op de keel sal doen spelen daer sij al vrij aen martonico guardeloepe gebreck van hebben nu die plaetsen ben onder de france compagnie

Hit rows are always preceded by a row containing the document title in which those hits occurred, in this case To Johan Gil, 12 februari 1665 by Gerard Douw. The document titles can be toggled on or off by using the Hide Titles (or Show Titles when titles are hidden) button at the bottom of the page. (If you hover the mouse over the title, the identification number of the document appears, in this case: bab2082.)

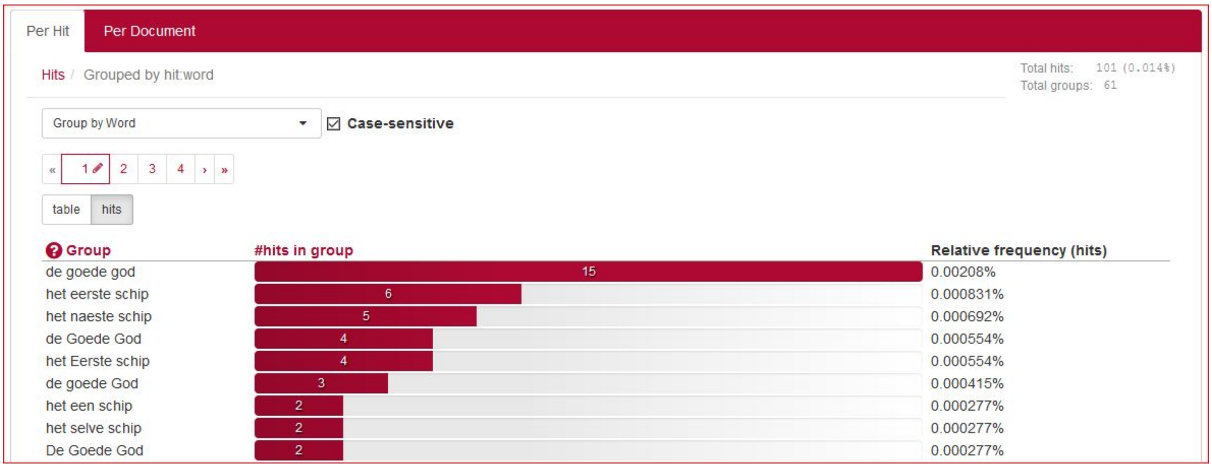
Sorting results

Click on any of the column headings (i.e. Before hit, Hit or After hit) to sort the hits on Words within that column, clicking again inverts the sorting.

You can also sort the results by means of the drop-down menu at the bottom of the page (Sort by ...), which offers you the possibility to sort by various attributes as Hit, Before hit, After hit, Letter, Sender, Addressee, Sent from.

Grouping results

Results Per Hit can be grouped by properties of Hit, Before hit, After hit and by the metadata of the documents in which those hits occur (Letter, Sender, Addressee and Sent from). Grouping is facilitated by the drop-down menu Group hits by. By selecting one of the properties a tick box appears that makes it possible to distinguish between case sensitive and case insensitive.



Hits / Grouped by hit:word

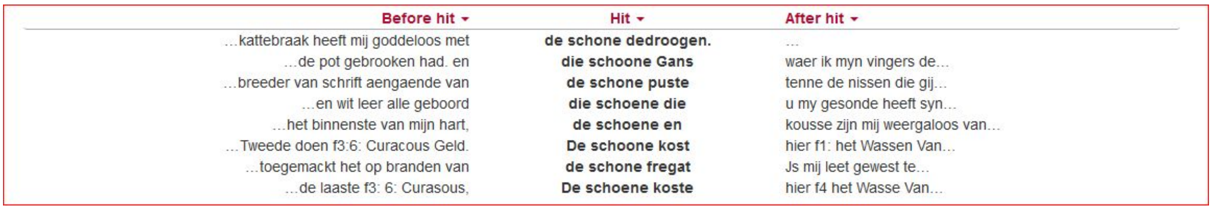
Total hits: 101 (0.014%)
Total groups: 61

Group by Word ☑ Case-sensitive

Group	#hits in group	Relative frequency (hits)
de goede god	15	0.00208%
het eerste schip	6	0.000831%
het naeste schip	5	0.000692%
de Goede God	4	0.000554%
het Eerste schip	4	0.000554%
de goede God	3	0.000415%
het een schip	2	0.000277%
het selve schip	2	0.000277%
De Goede God	2	0.000277%

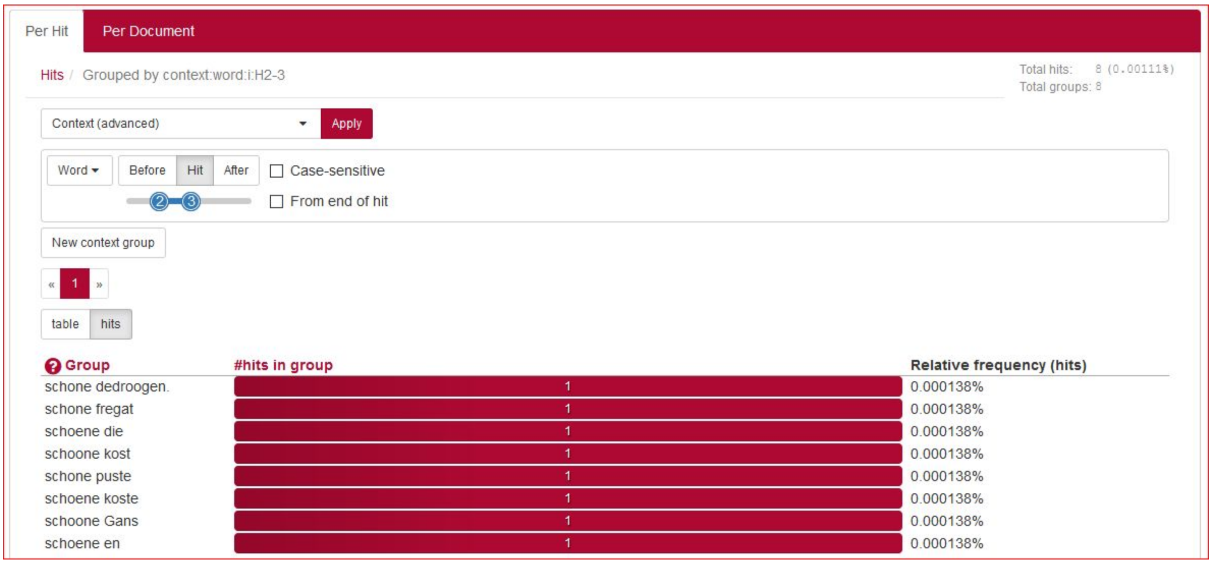
Advanced grouping options are available by selecting the option Context (advanced). It allows you to group the results by up to 5 tokens before or after the hits. It also allows you to group the results based on (parts of) the hits. By pressing the New context group you can group the results by another property or another range.

We will work that out using an example. A noun phrase consisting of a pronoun/determiner *die*, the adjective *schone* or *schoone* and an arbitrary noun *may* be found in Expert Search with the following query: [word="die"] [word="scho.?ne"] []. This produces the following hits (Titles are hidden):



Before hit ▾	Hit ▾	After hit ▾
...kattebraak heeft mij goddeloos met	**de schone dedroogen.**	...
...de pot gebrooken had. en	**die schoone Gans**	waer ik myn vingers de...
...breeder van schrift aengaende van	**de schone puste**	tenne de nissen die gij...
...en wit leer alle geboord	**die schoene die**	u my gesonde heeft syn...
...het binnenste van mijn hart,	**de schoene en**	kousse zijn mij weergaloos van...
...Tweede doen f3:6: Curacous Geld.	**De schoone kost**	hier f1: het Wassen Van...
...toegemackt het op branden van	**de schone fregat**	Js mij leet gewest te...
...de laaste f3: 6: Curasous,	**De schoene koste**	hier f4 het Wasse Van...

It is now possible to group the hits by the second and third tokens of those hits. See below.



Hits / Grouped by context:word:i:H2-3

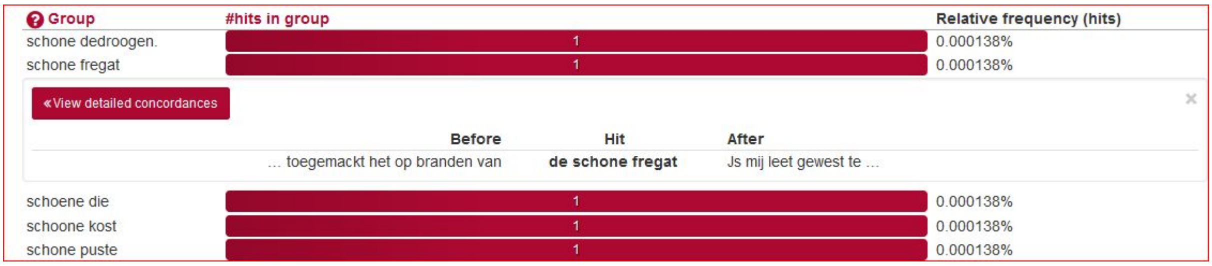
Total hits: 8 (0.00111%)
Total groups: 8

Context (advanced)

□ Case-sensitive
□ From end of hit

Group	#hits in group	Relative frequency (hits)
schone dedroogen.	1	0.000138%
schone fregat	1	0.000138%
schoene die	1	0.000138%
schoone kost	1	0.000138%
schone puste	1	0.000138%
schoene koste	1	0.000138%
schoone Gans	1	0.000138%
schoene en	1	0.000138%

Click a group to show or hide hits within that group, as shown below. Click once more on the group to close it again.



Group	#hits in group	Relative frequency (hits)
schone dedroogen.	1	0.000138%
schone fregat	1	0.000138%

Before	Hit	After
… toegemackt het op branden van	de schone fregat	Js mij leet gewest te …

Group	#hits in group	Relative frequency (hits)
schoene die	1	0.000138%
schoone kost	1	0.000138%
schone puste	1	0.000138%

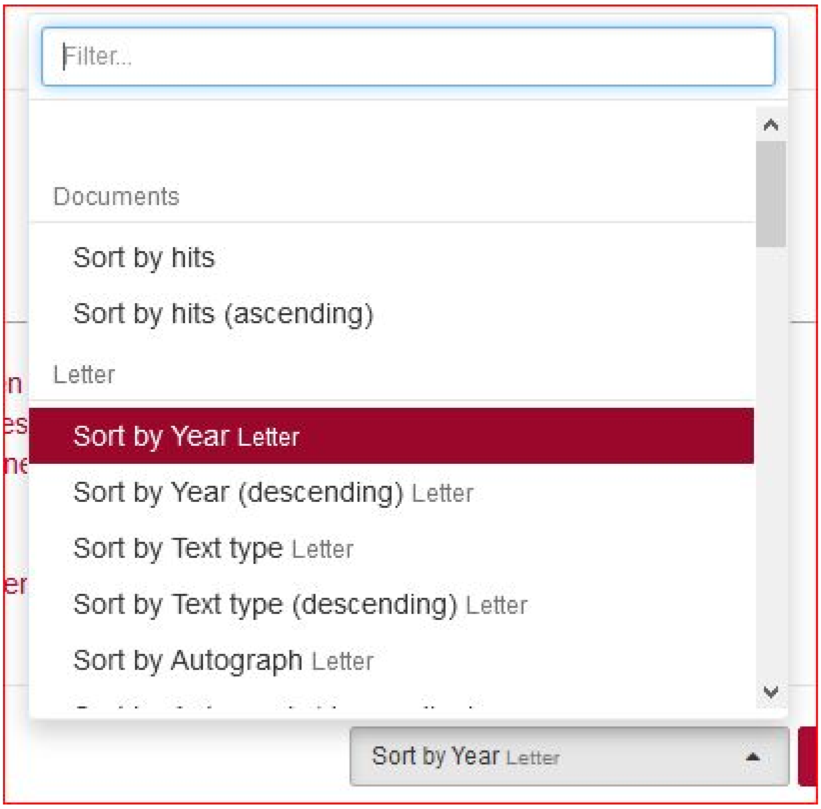
If more than twenty hits are found in a document, you can make them appear by clicking on Load more concordances; this button will appear right to the button View detailed concordances.

Click on View detailed concordances to go back to the normal hits view to see more detailed information for the hits in this group. The button Go back to grouped view brings you back to the list of groups.

Per Document view

Sorting results

Results can be sorted by means of the drop-down menu at the bottom of the page, which enables you to sort by Documents (e.g. the number of hits for your search query) and by the metadata Letter, Sender, Addressee and Sent from.



Click on a Document title to show the Content of the document in a new window. Hits from the current query will be highlighted in bold in the opened document. In the case of several hits the first hit will also appear in shadow. You can go to the next (or previous) hit within the same document by pressing the Hits (and - in the case of many hits - Pages) button.

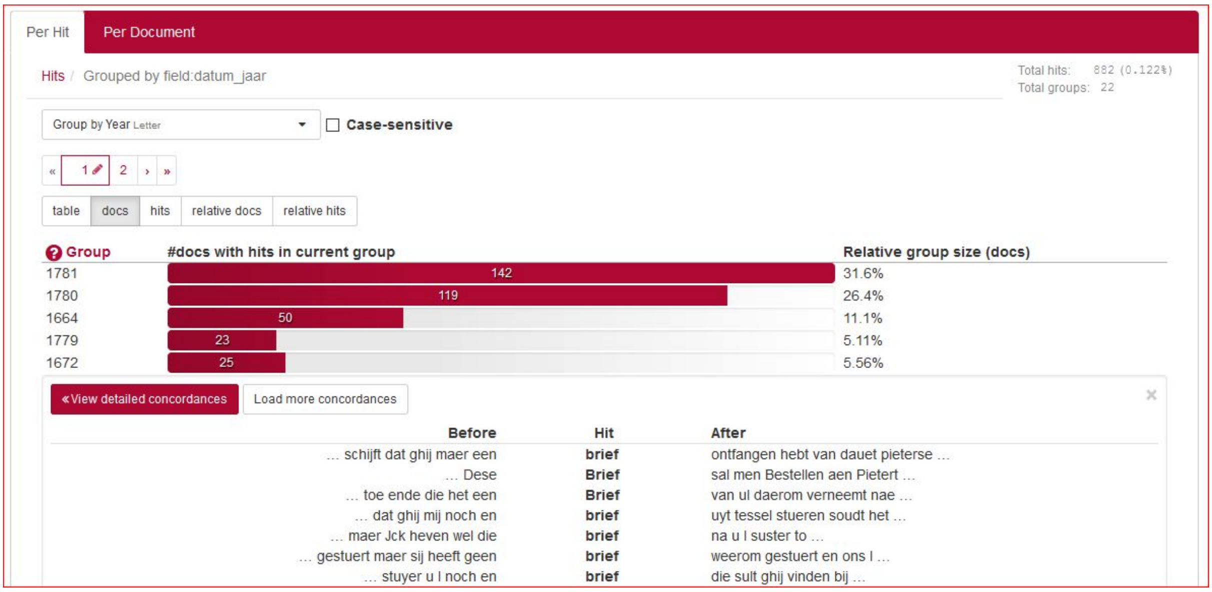
Grouping results

Results Per Document can be grouped by the metadata of the documents in which those hits occur (Letter, Sender, Addressee, Sent from). Here, grouping is facilitated by the drop-down menu Group docs by.

Exporting results

The search results - both Per hit as Per document - can be exported by using the Export or the Export for Excel button at the bottom right of the page. The first button transfers the search results - including all metadata - to a Comma-Separated Values-file. These CSV-files consist only of text data, which makes it easy to implement (read and/or write) them into a spreadsheet or database program. The second button offers the possibility to export the results - including all metadata - to a CSV-file for use with Excel.

Grouped results can be exported in the same way. However, if you would like to have the metadata with each concordance of a group, you must first click on the red bar of a specific group and then on View detailed concordances (see screenshot below). The results you then see can be exported by the use of the Export buttons. This operation must be carried out for each individual group you wish to export.



Total hits: 882 (0.122%)
Total groups: 22

☐ Case-sensitive

Group	#docs with hits in current group	Relative group size (docs)
1781	142	31.6%
1780	119	26.4%
1664	50	11.1%
1779	23	5.11%
1672	25	5.56%

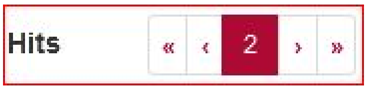
Before	Hit	After
... schijft dat ghij maer een	**brief**	ontfangen hebt van dauet pieterse ...
... Dese	**Brief**	sal men Bestellen aen Pietert ...
... toe ende die het een	**Brief**	van ul daerom verneemt nae ...
... dat ghij mij noch en	**brief**	uyt tessel stueren soudt het ...
... maer Jck heven wel die	**brief**	na u l suster to ...
... gestuert maer sij heeft geen	**brief**	weerom gestuert en ons l ...
... stuyer u l noch en	**brief**	die sult ghij vinden bij ...

Information about a document

Click on a document title to open this document in a new window: the Content window.

Content

Hits from the current query will be highlighted in bold in the opened document. In the case of several hits only the current hit will also appear in shadow. You can navigate from one hit to another by using the arrows at the Hits button:



Metadata

In the Metadata tab all metadata properties of the document are displayed. They provide information about the Letter, the Sender, the Addressee and Sent from.

Statistics

The Statistics tab shows several document statistics: the number of Tokens and the Type/token ratio. It is possible to print or to download these statistics via the menu symbol right of the title Vocabulary Growth.

Images

Under images you can find a photo of the original letter, kept in the National Archives (Kew, UK).

Exploring the corpus

The Explore tab has three subdivisions: Documents, N-grams and Statistics.

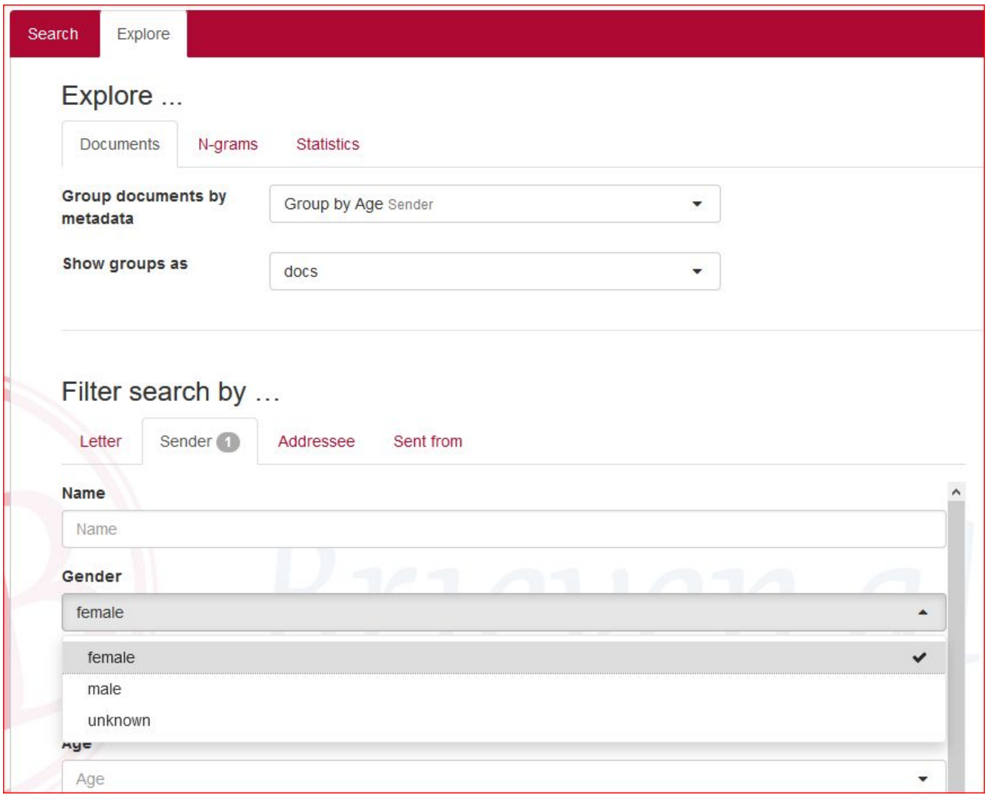
Documents

This subtab allows you to investigate the documents. It consists of two drop-down menus to specify the grouping of the metadata and to specify the way the groups are to be shown.

A simple example: suppose we want to obtain information about the age distribution of female senders within the *Brieven als Buit* corpus.

- In the Group documents by metadata drop-down menu, choose Group by Age (Sender)
- In Show groups as, select *docs*
- In the metadata search form (Filter search by), select in Sender Gender *female*

- Press Search



Explore ...

Group documents by metadata: Group by Age Sender

Show groups as: docs

Filter search by ...

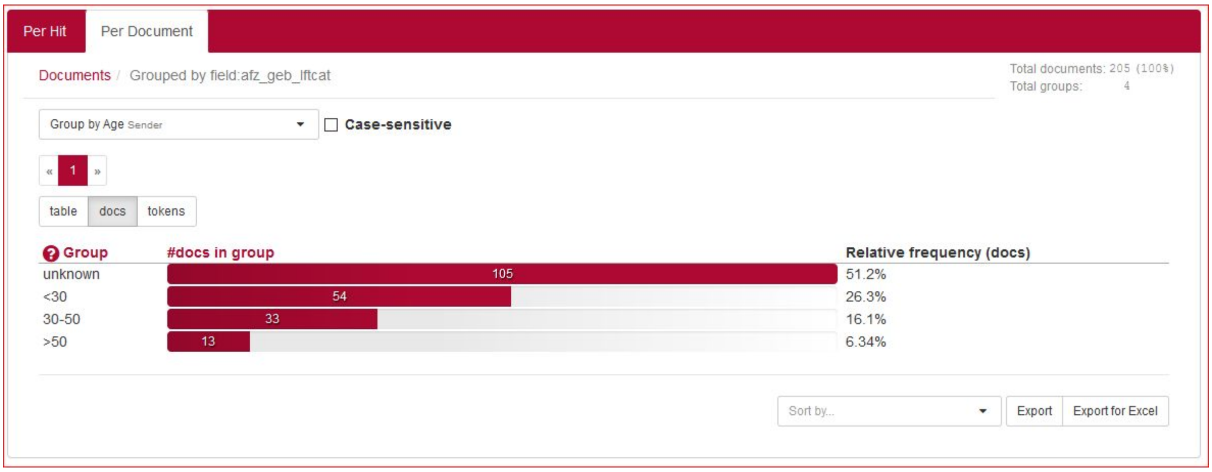
Name: Name

Gender: female

female ✓

Age: Age

You will get this result:



Documents / Grouped by field:afz_geb_lftcat

Total documents: 205 (100%)
Total groups: 4

Group by Age Sender

□ Case-sensitive

Group	#docs in group	Relative frequency (docs)
unknown	105	51.2%
<30	54	26.3%
30-50	33	16.1%
>50	13	6.34%

N-grams

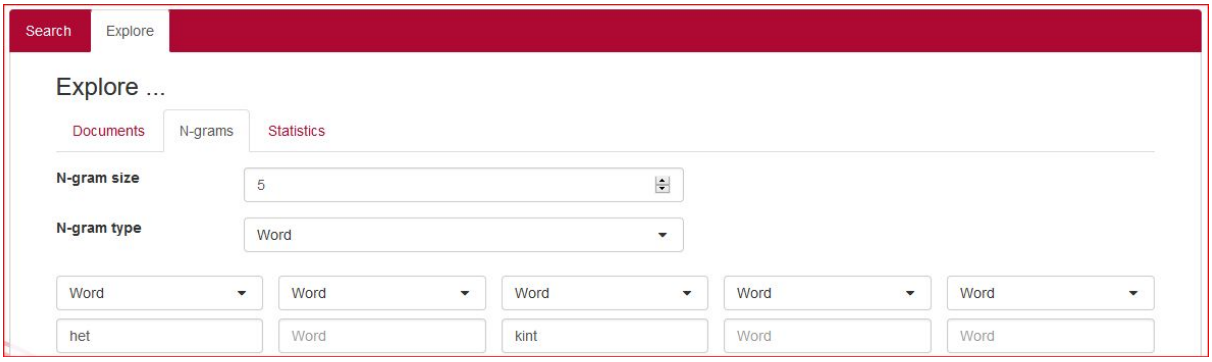
An *N-gram* is a sequence of *N* items. This option will list the frequency of different N-grams in a (sub-)corpus.

Options

- *N-gram size*: the length of the sequence (a number from 1 to 5; default setting is 5)
- *N-gram-type*: Word (i.e. word form); in the corpus *Brieven als Buit-2*, N always stands for a Word;

- It is also possible to restrict to, for instance, 5-grams with some slots already specified, as is shown in the following example.
- By using the Filter search by … you can create a subcorpus within the corpus *Brieven als Buit-2* for specific metadata.

Example

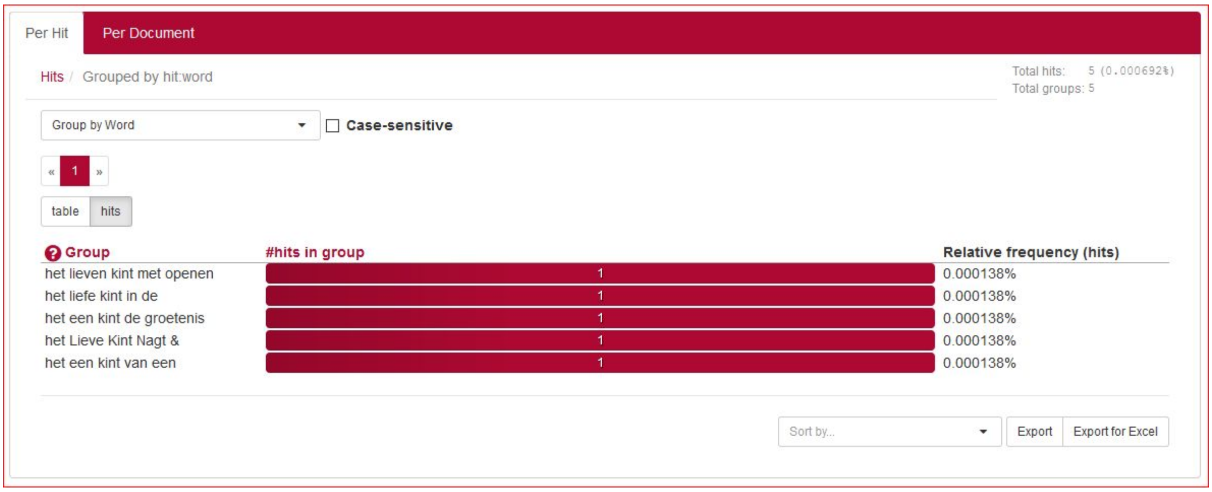


N-gram size: 5

N-gram type: Word

Word	Word	Word	Word	Word
het	Word	kint	Word	Word

Within all the documents of the *Brieven als Buit* corpus, you will find 10 occurrences of this so-called 5-gram (choose the option ‘Select all’ for both words and limit your search to Part of Speech *het* (ART) respectively *kint* (NOU-C)).



Total hits: 5 (0.000692%)
Total groups: 5

Group	#hits in group	Relative frequency (hits)
het lieven kint met openen	1	0.000138%
het liefe kint in de	1	0.000138%
het een kint de groetenis	1	0.000138%
het Lieve Kint Nagt &	1	0.000138%
het een kint van een	1	0.000138%

Statistics (frequency lists)

Here, you can produce frequency lists for the corpus. It is rather similar to the previous option, but restricted to 1-grams.

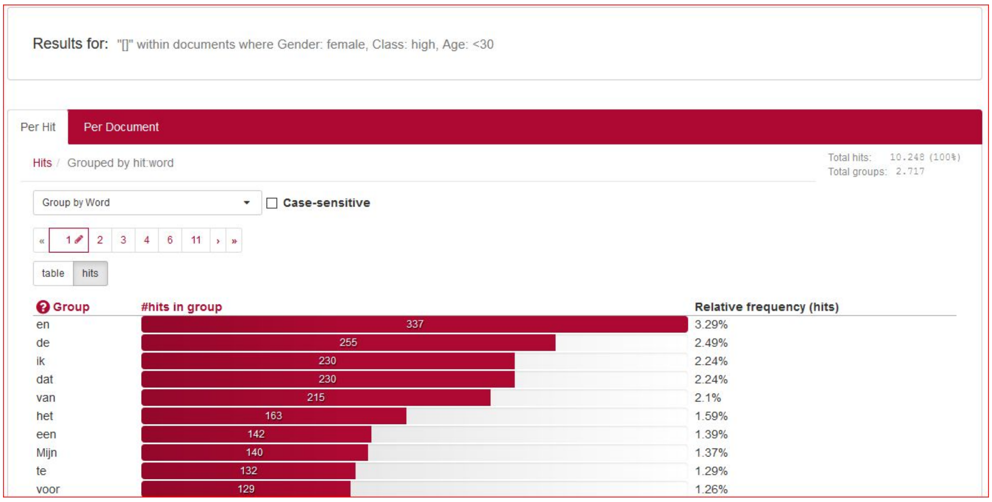
Options

- *Frequency list type:* the only possibility in the corpus *Brieven als Buit-2* is a list of Words (i.e. Word form)
- By using the Filter search by... you can create a subcorpus within the corpus *Brieven als Buit-2* for specific metadata.

Example

It is possible to determine the use of the ten most frequently used words by female writers of different classes and ages by searching for Frequency list type Word and by filtering search by Gender, Class and Age.

For female writers who are younger than 30 years old and who belong to the high class, this results in:



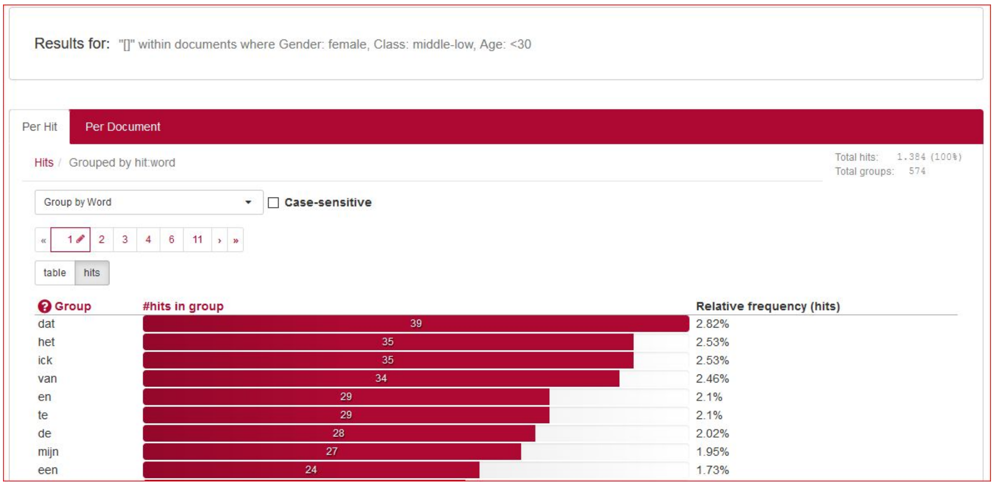
Results for: "[]" within documents where Gender: female, Class: high, Age: <30

Total hits: 10.248 (100%)
Total groups: 2.717

Group by Word
□ Case-sensitive

Group	#hits in group	Relative frequency (hits)
en	337	3.29%
de	255	2.49%
ik	230	2.24%
dat	230	2.24%
van	215	2.1%
het	163	1.59%
een	142	1.39%
Mijn	140	1.37%
te	132	1.29%
voor	129	1.26%

For female writers who are younger than 30 years old but who belong to the middle-low class, this results in:



Results for: "[]" within documents where Gender: female, Class: middle-low, Age: <30

Total hits: 1.384 (100%)
Total groups: 574

Group by Word
□ Case-sensitive

Group	#hits in group	Relative frequency (hits)
dat	39	2.82%
het	35	2.53%
ick	35	2.53%
van	34	2.46%
en	29	2.1%
te	29	2.1%
de	28	2.02%
mijn	27	1.95%
een	24	1.73%

Appendix: Corpus Query Language

BlackLab supports Corpus Query Language, a full-featured query language introduced by the IMS Corpus WorkBench (CWB) and also supported by the Lexicom Sketch Engine. It is a standard and powerful way of searching corpus.
The basics of Corpus Query Language is the same in all three projects, but in there are a few minor differences in some of the more advanced features, as well as some features that are exclusive to some projects. For most queries however, this will not be an issue.
This page will introduce the query language and show all features that BlackLab supports. If you want to learn even more about CQL, see CWB CQP Query Language Tutorial and Sketch Engine Corpus

Query Language.

CQL support

For those who already know CQL, here's a quick overview of the extent of BlackLab's support for this query language. If there is feature we don't support, yet is important to you, please let us know. If it's quick to add, we may be able to help you out.

Supported features

BlackLab currently supports (arguably) most of the important features of Corpus Query Language:

- Matching on token annotations (also called properties or attributes), using regular expressions and =, !=, !. Example: [word="bank"] (or just "bank")
- Case/accent-sensitive matching. Note that, unlike in CWB, case-INsensitive matching is currently the default. To explicitly match case-/accent-insensitively, use "(?i)...". Example: "(?-i)Mr\." "(?-i)Banks"
- Combining criteria using &, | and !. Parentheses can also be used for grouping. Example: [lemma="bank" & pos="V"]
- Match-all pattern [] matches any token. Example: "a" [] "day"
- Regular expression operators +, *, ?, {n}, {n,m} at the token level. Example: [pos="AA"]+
- Sequences of token constraints. Example: [pos="AA"] "cow"
- Operators |, & and parentheses can be used to build complex sequence queries. Example: "happy" "dog" | "sad" cat"
- Querying with tag positions using e.g. <s> (start of sentence), </s> (end of sentence), <s/> (whole sentence) or <s> ... </s> (equivalent to <s/> containing ...). Example: <s> "The" . XML attribute values may be used as well, e.g. <ne type="PERS"/> ("named entities that are persons").
- Using within and containing operators to find hits inside another set of hits. Example: "you" "are" within <s/>
- Using an anchor to capture a token position. Example: "big" A:[]. Captured matches can be used in global constraints (see next item) or processed separately later (using the Java interface; capture information is not yet returned by BlackLab Server). Note that BlackLab can actually capture entire groups of tokens as well, similarly to regular expression engines.
- Global constraints on captured tokens, such as requiring them to contain the same word. Example: "big" A:[] "or" "small" B:[] :: A.word = B.word

See below for features not in this list that may be added soon, and let us know if you want a particular feature to be added.

Differences from CWB

BlackLab's CQL syntax and behaviour differs in a few small ways from CWBs. In future, we'll aim towards greater compliance with CWB's de-facto standard (with some extra features and conveniences).

For now, here's what you should know:

- Case-insensitive search is currently the default in BlackLab, although you can change this if

you wish. CWB and Sketch Engine use case-sensitive search as the default. We may change our default in a future major version.
If you want to switch case-/diacritics-sensitivity, use "(?-i).." (case-sensitive) or "(?i).." (case-insensitive, usually the default). CWBs %cd flags for setting case/diacritics-sensitivity are not (yet) supported, but will be added.

- If you want to match a string literally, not as a regular expression, use backslash escaping: "e\.g\.". %l for literal matching is not yet supported, but will be added.
- BlackLab supports result set manipulation such as: sorting (including on specific context words), grouping/frequency distribution, subsets, sampling, setting context size, etc. However, these are supported through the REST and Java APIs, not through a command interface like in CWB. See BlackLab Server overview).
- Querying XML elements and attributes looks natural in BlackLab: <s/> means "sentences", <s> means "starts of sentences", <s type='A'> means "sentence tags with a type attribute with value A". This natural syntax differs from CWBs in some places, however, particularly when matching XML attributes. While we believe our syntax is the superior one, we may add support for the CWB syntax as an alternative.
  We only support literal matching of XML attributes at the moment, but this will be expanded to full regex matching.
- In global constraints (expressions occurring after ::), only literal matching (no regex matching) is currently supported. Regex matching will be added soon. For now, instead of A:[] "dog" :: A.word = "happy|sad", use "happy|sad" "dog".
- To expand your query to return whole sentences, use <s/> containing (...). We don't yet support CWBs expand to, expand left to, etc., but may add this in the future.
- The implication operator -> is currently only supported in global constraints (expressions after the :: operator), not in a regular token constraints. We may add this if there's demand for it.
- We don't support the @ anchor and corresponding target label; use a named anchor instead. If someone makes a good case for it, we will consider adding this feature.
- backreferences to anchors only work in global constraints, so this doesn't work: A:[] [] [word = A.word]. Instead, use something like: A:[] [] B:[] :: A.word = B.word. We hope to add support for these in the near future, but our matching approach may not allow full support for this in all cases.

(Currently) unsupported features

The following features are not (yet) supported:

- intersection, union and difference operators. These three operators will be added in the future. For now, the first two can be achieved using & and | at the sequence level, e.g. "double" [] & [] "trouble" to match the intersection of these queries, i.e. "double trouble" and "happy" "dog" | "sad "cat" to match the union of "happy dog" and "sad cat".
- _ meaning "the current token" in token constraints. We will add this soon.
- lbound, rbound functions to get the edge of a region. We will probably add these.
- distance, distabs functions and match, matchend anchor points (sometimes used in global constraints). We will see about adding these.
- using an XML element name to mean 'token is contained within', like [(pos = "N") & !np] meaning "noun NOT inside in an tag". We will see about adding these.

- a number of less well-known features. If people ask, we will consider adding them.

Using Corpus Query Language

Matching tokens

Corpus Query Language is a way to specify a "pattern" of tokens (i.e. words) you're looking for. A simple pattern is this one:

```
[word="man"]
```

This simply searches for all occurrences of the word "man". If your corpus includes the per-word properties lemma (i.e. headword) and pos (part-of-speech, i.e. noun, verb, etc.), you can query those as well. For example, to find a form of word "search" used as a noun, use this query:

```
[lemma="search" & pos="NOU-C"]
```

This query would match "search" and "searches" where used as a noun. (Of course, your data may contain slightly different part-of-speech tags.)
The first query could be written even simpler without brackets, because "word" is the default property:

```
"man"
```

You can use the "does not equal" operator (!=) to search for all words except nouns:

```
[pos != "NOU-C"]
```

The strings between quotes can also contain wildcards, of sorts. To be precise, they are regular expressions, which provide a flexible way of matching strings of text. For example, to find "man" or "woman", use:

```
"(wo)?man"
```

And to find lemmata starting with "under", use:

```
[lemma="under.\*"]
```

Explaining regular expression syntax is beyond the scope of this document, but for a complete overview, see here.

Sequences

Corpus Query Language allows you to search for sequences of words as well (i.e. phrase searches, but with many more possibilities). To search for the phrase "the tall man", use this query:

```
"the" "tall" "man"
```

It might seem a bit clunky to separately quote each word, but this allow us the flexibility to specify exactly what kinds of words we're looking for. For example, if you want to know all single adjectives used with man (not just "tall"), use this:

```
"an?|the" [pos="AA"] "man"
```

This would also match "a wise man", "an important man", "the foolish man", etc.

Regular expression operators on tokens

Corpus Query Language really starts to shine when you use the regular expression operators on whole tokens as well. If we want to see not just single adjectives applied to "man", but multiple as well:

```
"an?|the" [pos="AA"]+ "man"
```

This query matches "a little green man", for example. The plus sign after [pos="AA"] says that the preceding part should occur one or more times (similarly, * means "zero or more times", and ? means "zero or one time").

If you only want matches with two or three adjectives, you can specify that too:

```
"an?|the" [pos="AA"]{2,3} "man"
```

Or, for two or more adjectives:

```
"an?|the" [pos="AA"]{2,} "man"
```

You can group sequences of tokens with parentheses and apply operators to the whole group as well. To search for a sequence of nouns, each optionally preceded by an article:

```
("an?|the"? [pos="NOU-C"])+
```

This would, for example, match the well-known palindrome "a man, a plan, a canal: Panama!" (A note about punctuation: in BlackLab, punctuation tends to not be indexed as a separate token, but as a property of a word token - CWB and Sketch Engine on the other hand tend to index punctuation as a separate token instead. You certainly could choose to index punctuation as a separate token in BlackLab, by the way -- it's just not commonly done. Both approaches have their advantages and disadvantages, and of course the choice affects how you write your queries.)

Case- and diacritics-sensitivity

CWB and Sketch Engine both default to (case- and diacritics-)sensitive search. That is, they exactly match upper- and lowercase letters in your query, plus any accented letters in the query as well. BlackLab, on the contrary, defaults to *IN*sensitive search (although this default can be changed if you like). To match a pattern sensitively, prefix it with "(?-i)":

```
"(?-i)Panama"
```

If you've changed the default search to sensitive, but you wish to match a pattern in your query insensitively, prefix it with "(?i)":

```
[pos="(?i)NOU-C"]
```

Although BlackLab is capable of setting case- and diacritics-sensitivity separately, it is not yet possible from Corpus Query Language. We may add this capability if requested.

Matching XML elements

Corpus Query Language allows you to find text in relation to XML elements that occur in it. For example, if your data contains sentence tags, you could look for sentences starting with "the":

```
<s>"the"
```

Similarly, to find sentences ending in "that", you would use:

```
"that"</s>
```

You can also search for words occurring inside a specific element. Say you've run named entity recognition on your data and all person names are surrounded with <person>...</person> tags. To find the word "baker" as part of a person's name, use:

```
"baker" within <person/>
```

Note that forward slash at the end of the tag. This way of referring to the element means "the whole element". Compare to <person>, which means "the element's open tag", and </person>, which means "the element's close tag".

The above query will just match the word "baker" as part of a person's name. But you're likely more interested in the entire name that contains the word "baker". So, to find those full names, use:

```
<person/> containing "baker"
```

Or, if you simply want to find all persons, use:

```
<person/>
```

As you can see, the XML element reference is just another query that yields a number of matches. So as you might have guessed, you can use "within" and "containing" with any other query as well. For example:

```
([pos="AA"]+ containing "tall") "man"
```

will find adjectives applied to man, where one of those adjectives is "tall".

Labeling tokens, capturing groups

Just like in regular expressions, it is possible to "capture" part of the match for your query in a "group".

CWB and Sketch Engine offer similar functionality, but instead of capturing part of the query, they label a single token. BlackLab's functionality is very similar but can capture a number of tokens as well.

Example:

```
"an?|the" Adjectives:[pos="AA"]+ "man"
```

This will capture the adjectives found for each match in a captured group named "Adjectives". BlackLab also supports numbered groups:

```
"an?|the" 1:[pos="AA"]+ "man"
```

Global constraints

If you tag certain tokens with labels, you can also apply "global constraints" on these tokens. This is a way of relating different tokens to one another, for example requiring that they correspond to the same word:

```
A:[] "by" B:[] :: A.word = B.word
```

This would match "day by day", "step by step", etc.