Tara Struik

4065700

Research Paper

Nicoline van der Sijs

21-07-2015

Nederlab: An evaluation

1. **Introduction**

The Nederlab project[[1]](#footnote-1) was started in 2013 to fulfil the need of a unified, comprehensive corpus of the Dutch language. Up to now, materials were scattered across different institutions that each adhered to their own guidelines. This made large-scale research difficult, if not impossible. Nederlab aims to combine all available material related to the Dutch national heritage in an open access web interface. This means that materials become freely available for everyone and for all sorts of purposes. This is a very ambitious project and one that offers ample opportunity for research in the future. It is also a project that involves decision making and collaboration with other parties. The aim of this paper is to review the Nederlab project as a whole. It will involve a discussion of the decisions that were made in order to arrive at a comprehensive corpus and it will also review the collection itself. Information was gathered by means of a literature study and interviews with Erwin Komen, one of the members of the Nederlab team. Together with an evaluation of the Nederlab website itself, this will result in a comprehensive overview of Nederlab and its uses, which highlights both its strengths, weaknesses and possibilities for the future.

The paper is structured as follows. In the following section, Nederlab and its specifications will be introduced and discussed, while Nederlab itself is reviewed in section 3. Section 4 will discuss recommendation for future implementations. Section 5 will conclude the paper.

1. **Nederlab: a general introduction**

The Nederlab project was started in 2013 to bring together all digitised texts related to the language, culture and history of the Netherlands. In other words, all texts that are relevant to our national heritage. It currently hosts approximately 5 million texts from three collections. The ultimate aim is to combine all available material into one database which can be searched by scholars in a coherent and consistent manner.

* 1. **Why Nederlab?**

Nederlab is a response to ongoing changes within research in the humanities. The importance of quantitative research and statistics has been stressed repeatedly and this had the inevitable result that technology started to play a much bigger role in research in the humanities. This reliance on technology has the benefit that computers can now do a lot of the work that previously had to be done by hand. Computers work fast, compared to men, which means that much more data can be handled in the same amount of time. This is of course very convenient, but it also means that these data have to be available and that there is software that can be used to answer the researcher’s questions.

This can be a tricky enterprise, especially for historical sources. Large-scale research on modern languages is relatively easy nowadays, as most texts are born digital and most are not even available in print format. They are all created on similar machines, with similar programmes. They are coded and stored in the same format and are created with the same conventions. If they are not, it is fairly simple to convert them into the desired format. It is only a matter of collecting the relevant files and writing a program or tool that can extract the desired data to arrive at an answer to a research question, to put it simply. For historical texts this is much harder. Text will first need to be digitised and need to be freely available. Many of the available Dutch texts are scattered across different institutions, all of which have digitised these texts in their own specific ways. Texts are available in different formats, with different meta data and in different quality. This incoherency is at odds with the aims of present-day research: finding answers to complex research questions. The data is not available in the numbers necessary or they cannot be compared, because the right metadata or coding are lacking.

The study of the history of the Dutch language also suffers from incoherency of the data (van der Sijs 2014a). Different institutions make different decisions with regard to the materials that they want to digitise. This does not only apply to the decision which materials to digitise, but also to what kind of metadata is included. Furthermore, there is the problem of how to digitise. Most texts are digitised by means of OCR (optical character recognition), but this method is imperfect and often leads to many errors. Additionally, there is the problem of what tools to use. Tools that have been developed so far only work on the texts for which they are designed or they work for modern texts, but not for historical ones. This makes it very hard to conduct research on a large dataset, as it is very hard to work with data from different sources in a consistent way.

This is not only the reason why there was no comprehensive research corpus up to now; it is also the reason why one is needed. Especially for diachronic research large quantities of data are necessary. The data needs to be annotated in a consistent manner and also metadata needs to be complete and consistent across texts, where possible. Researchers from different disciplines working in the field of Dutch national heritage can benefit from a comprehensive database of all texts related to the Dutch language. This is why Nederlab was brought to life, to unite all texts relevant to the Dutch national heritage in one unified work space.

The remainder of this section will discuss the behind the scenes of Nederlab. It will discuss the materials included, as well as how these were gathered and prepared. It will also discuss how these texts are enriched and finally how the user is able to get the information that he needs.

* 1. **Materials in Nederlab**

The Nederlab project is still ongoing and the first version was launched on March 13, 2015. This version, however, does not contain all the files that ultimately will be included, nor does it have all the functionalities implemented. The project runs until 2017 and will continuously be adding new text files and tools to the database. The ultimate aim is to add all available material from 800 a.d. to now. Currently, the bulk of the data comes from the National Library of the Netherlands (KB), the Digital Library of Dutch Literature (DBNL) and Early Dutch Books Online (EDBO). At the time of writing the collection included 4.574.420 titles written by 94.777 authors. A new version of Nederlab was launched on 7 July 2015 and the collection was expanded. It now includes 13.485.158 titles written by 99.030 authors[[2]](#footnote-2).

The collection includes both autonomous (*zelfstandig*) and non-autonomous (*niet-zelfstandig*) titles. Autonomous titles are books, both fiction and non-fiction. It is possible that the same text occurs twice or more in the database, because it is incorporated in more than one collection or because there are two editions of the same text. Newspapers, magazines and annuals are not considered autonomous, but are divided on an article-based level. For this reason, they are considered non-autonomous. The articles in one newspaper are united under one header.

Nederlab provides information of the number of texts only in graphical format. These graphs have been included in Appendix A. They show that the bulk of the texts come from the DBNL and newspaper articles (from the KB collection). In terms of period, the texts from the DBNL are most diverse, ranging from 1600 to now. The bulk of the texts are from the 20th century, followed by a large collection of 19th century texts. The EBDO collection is fairly small and all texts are from the 18th century, with a peak around 1775-1790. The newspaper articles are mostly from the early 19th century. There are no newspaper articles available from after approximately 1850. However, the pictures shown at the homepage are not entirely accurate. The graph for the distribution of documents from EDBO indicates that there is a fair amount of material from around 1700, but the EDBO only includes materials from 1781-1800, so this is impossible.

* + 1. **OCR errors and solutions**

Most of the texts are digitised by means of OCR: optical character recognition. Specialised OCR software converts the hardcopy versions of texts into machine-readable text. The quality of these digitisations is not always consistent and often leaves much to be desired, as it often depends on the purpose for which it was digitised. In some cases the aim of digitisation was to digitise as much data in a short time frame, which often goes at the cost of quality, as it serves to give a general impression of the material that needs to be covered. Some texts are digitised for preservation purposes, while some are needed to answer specific research questions. This requires different quality standards. Nowadays, some OCR engines boast 99% accuracy, which sounds impressive, but this still means that for every 100 words, one will be misrecognised. Reynaert (2008) illustrates the severeness of this problem by identifying all forms of the word *belasting* in contemporary parliamentary texts. He found 1,577 different forms of the word *belasting*. Evaluation of one newspaper from 1918 yielded 1,468 variants of the word *regering*. It should be noted that these variants included all morphological and compound forms, but it still shows that the quality of OCRed text is poor. Errors can be insertion, deletion, transposition, substitution or a combination of these, leading the word *regeering* to be recognised by the OCR engine as *regeermg.*

Reyneart (2014b) points out the possible effects OCR errors can have on the data in Nederlab. First of all, it is unclear how accurate the actual results from a query are. Van der Sijs (2014a) illustrates this with an example on the origin of the word *televisie,* of which we find the first attestation in 1853 and the example of *tweet* which appears first in 1646. We know from experience that this is impossible and closer inspection of the texts reveal indeed that these examples are misrecognitions. Second, if the corpus is to be enriched linguistically, there will be a direct influence on the tools that are used for this purpose. Both cases are good incentives to find a solution to this problem that is both time- and cost-efficient. Nederlab has (partly) remedied this problem by using Text-Induced Corpus Clean-op (TICCL).

TICCL is not only meant to correct OCR errors; it looks for any kind of lexical variation (Reynaert 2008, 2010) varying from historical spelling changes, typographical and/or typesetting changes, transmission noise and morphological variation. This leads to a significantly improved output that is much more useful for the research purposes for Nederlab. This can be explained by considering the ground truth of an OCR text and the OCR post-correct gold standard (Reynaert 2014a).

The ground truth of an OCRed text is an exact digital replication of the source text. However, this means that the digitised text is still subject to the physical limitations of the source text’s print form, meaning that any split words or run-on words are still there in the digitised text. Additionally, many manually typeset texts suffer from (spelling) mistakes that need to be resolved. Finally, there is the issue of spelling variation in historical texts. This can be resolved in various ways. The first is to adapt the spelling to present-day spelling conventions, so that the tools to analyse the texts needs not be adapted. However, this has the potential problem that the lexicon has changed significantly over time, meaning that some words might not even have a modern equivalent. It is also possible to transcribe the nearest historical variant, but this raises the question which historical variant needs to be transcribed.

Removing changes to the text as a result of physical limitations to the printed text and resolving spelling variation as well as OCR errors leads to a perfect, digitised version of the text. This is called the gold standard. Reynaert (2014a) has manually created a gold standard for one Dutch book. He created both a historical gold standard (for the historical text in its printed spelling) and a contemporary gold standard (in terms of spelling of individual words). The reported accuracy of the OCRed text compared to the ground truth is 87.54%, meaning that the result of OCR is not an exact duplication of the source text. The accuracy compared to the historical gold standard is 88.75% and to the contemporary gold standard is 79.85%, showing that there is room for improvement. However, instead of arguing that the OCR process has to be improved, proposed by for instance de Does & Depuydt (2013), Reynaert argues that there should be OCR-post-correction, because an improvement of the OCR process will not result in the quality that is desired for purposes of linguistic research.

The question is what OCR post-correction with TICCL can achieve. To find an answer to this, TICCL needs to be explained in more detail. TICCL differs from most spelling correction systems in that it is a global, character confusion-based operation, rather than a local word-based variant retrieval procedure. It identifies “all the pairs of text strings in the text collection that display a particular confusion,” instead of only considering one confusion at the time Reynaert (2010: 173). The system derives statistics on the confusions from the corpus itself, meaning that no training material is required, something that other system do need. The system is based on anagram hashing. The details of this process will be left for the reader to consider at his own leisure, but the idea is that the algorithm converts each character in a string into a numerical value. The value that this gives is an identifier of the word. This value is then compared to all other values in the corpus to find deviation. If the deviations are within 2 Levenshtein distances then the deviations are corrected.

The advantage of TICCL is that it does not require training materials. A lexicon is created of all the words in the corpus. However, the post-correcting significantly benefits from adding a separate word list and/or a list of relevant names, place names, etc. The developers have chosen a sequential character confusion-based approach, because this has proven to be significantly faster than a focus word approach, in which all possible deviations are calculated for every single word in the text by descending order of the word forms’ frequencies. The sequential character confusion-based approach first calculates all the possible character confusion of a particular alphabet and then compares this to the data. It returns a list of word pairs that display particular numerical differences. The output is sorted “according to the observed order of having been filtered according to the sizes of their CC [character confusion] membership set, then according to the LD between canonical form and lexical variant, and finally according to the frequency of the canonical form” (Reynaert 2010: 179).

To post-correct the vast amount of data in Nederlab TICCL has been equipped with the diachronic lexicon and historical name list developed at the INL (Reynaert 2014b). The sorting process described in (Reynaert 2010) is not as solid as desired, because either of the three ways of sorting may have to be given prominence, which is why in Nederlab each candidate list for each of the sources of ranking information is sorted separately. Each candidate list is given an ascending separate numeric ranking. The rankings for each CC are summed and divided by the total number of rankings. Reynaert (2014b) shows that TICCL (at time of the publication of the paper) reached an accuracy of 95.97% when two lexicons and a word list were included on the DPO35 book as part of the 10,000 books collection of EDBO. This is a highly encouraging result, but Reynaert notes that there are still major challenges to be overcome to even further increase the accuracy if linguistic enrichment is to be implemented, something that can never be done by post-correction tools alone.

**2.2.2. Linguistic annotation**

One of the plans for the future is to add linguistic information to the data. This will be done by using Frog, originally known as Tadpole. Frog (van den Bosch et al. 2007) is a morpho-syntactic analyser and dependency parser. The program combines memory-based natural language processing modules. It tokenises tags, lemmatises and morphologically segments word tokens. It also creates a dependency graph and identifies base phrase chunks in sentences. Finally, it will attempt to label all entities. Its output is in FoLiA format (van Gompel & Reynaert 2013). The program was initially built to work on modern Dutch texts, so it has to be recoded in order to work with older texts. Frog has been tested on older texts (Sang 2014), but its performance was not optimal, which came as no surprise. Especially recognising lemmas, word classes and names were problematic. The following solutions were proposed:

* Improvement of character recognition. It is not possible to redo OCR, but a programme such as TICCL can be used to reduce OCR mistakes, which gives Frog a more solid base to work with, which can lead to better performance.
* Only content needs to be processed. This means that any page numbers, notes or headers will need to be processed separately. These will need to be marked or Frog will need to be able to recognise them.
* Adding a lexicon of older Dutch. This would significantly improve the performance on lemmas and word classes.
* The use of capital letters is very different in older texts than it is in modern texts. This leads to difficulty for Frog to recognise names. Standardising the use of capital in the texts would improve its performance.

Another possible strategy is to modernise the older texts. This strategy is employed in more projects that involve collection of historical data (cf. Erjavec 2011) and which has produced reliable results. The reason why this works so well is that spelling is most often the most striking difference between modern languages and their historical counterpart (Pattersson et al. 2014). Patternson et al. (2014) compare three methods for normalising historical text: a simplistic filtering model, a Levenshtein-based approach and a character-based machine translation approach. They tested the methods on 5 different languages (English, Germanic, Hungarian, Icelandic, Swedish) and found that all three methods significantly improve performance of NLP tools on historical data. In most cases, the character-based machine translation approach worked best. The Levenshtein-approach has the advantage that no training data is required, but using the SMT approach is likely to improve results. This only requires a small training set, since it is character-based and so should be doable.

Nederlab also wants to add a layer of modernised Dutch to the older texts. However, they will first need to experiment to see what works best before it will be implemented and introduced to the public.

**2.2.3. Infrastructure**

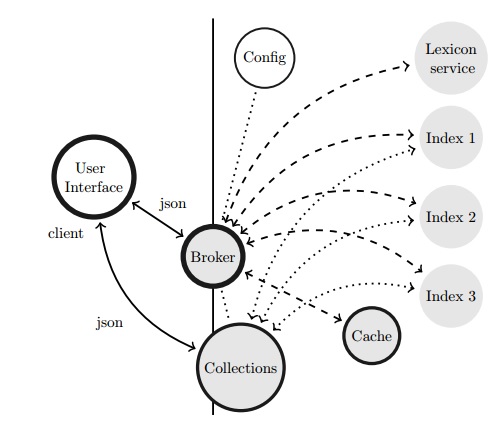
Before the collection can be searched effectively, some other issues have to be resolved. One of them is the different kind of metadata formats that are used by the source collections. Each collection has its own metadata schema with its own richness and gaps. Combining the metadata under one denominator results in information loss, which is undesirable. CLARIN has developed the Component Metadata Infrastructure (CMDI) to solve this problem (Broeder et al. 2011). The CMDI is a framework that can accommodate different XML-based metadata. The CMDI is built on different metadata components, each of which describes their own piece of metadata (such as title of the document, author, date of composition, etc.). Each component can include subcomponents. These metadata components are then compiled into metadata profiles, which can then be used as a metadata schema for any type of document. Nederlab uses its own core metadata schema, but it is fully compliant with the CMDI metadata framework (Brouwer et al. 2013a). By using its own metadata schema Nederlab facilitates the indexing of metadata, as well as support their editorial processes and focus specifically on the diachronic aspects of the data[[3]](#footnote-3).

The second issue is that all texts need to be converted to a unified XML format. The format that is chosen is the Format for Linguistic Annotation (FoLiA) XML format (van Gompel & Reynaert 2013). One of the advantages of the FoLiA format is that it is compatible with many tools, because it was developed from a bottom-up approach, rather than ad-hoc with ad-hoc extensions that are poorly schematised and hard to validate. This might work well for a temporary data format, but for a data exchange or corpus storage format, such as Nederlab, a more extensible and formalised format is required. Because of its inline annotation FoLiA has the advantage over other formats as it makes it accessible and human-readable. Another big advantage of using FoLiA is the set of tools that are already available to work with the format, such as TiCCL for corpus clean-up, Frog for linguistic enrichment and the broker, the corpus retrieval engine we will discuss next.

The data is first indexed and by means of Apache SOLR[[4]](#footnote-4). The broker environment is responsible for the actual search. The broker is introduced to handle the heterogeneity of the different datasets that are currently in Nederlab (Brouwer et al. 2013b) and can be seen as an extra layer between the back-end of the infrastructure and the front-end. It is said to “increase coverage, improve the effectiveness of retrieval and increase the ease of use” (Brouwer et al. 2013b: 1).

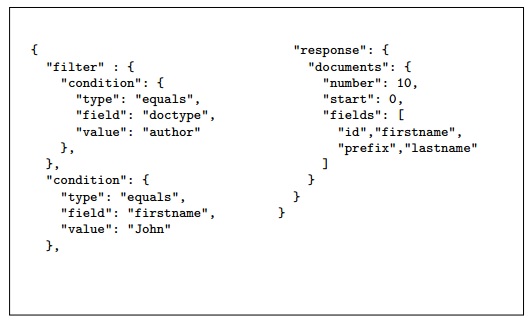
A broker has often been opted as a solution for searching large sets of data with a lot variation. It is also effective in aggregated search to arrive at one output format compiled from different nuggets of information. Within distributed information retrieval the use of a Broker has been studied extensively. Thomas (2012) discussed the alleged advantages of distributed information retrieval over information retrieval that comes from one single source. His arguments are mainly aimed at internet searches and he suggests that the advantages might not be as large as they seem. However, he admits that in some cases of distributed information retrieval a broker can be useful, because in these cases there are “organizational and business decisions that prevent full cooperation.” Additionally, Sushmita et al. (2009) show by means of a task-based evaluation of an aggregated search interface that the use of such a solution can lead to an increase in relevant pieces of information retrieved, but also in user evaluations in terms of user friendliness.

The broker that is used in Nederlab is a PHP based JSON web service, which is accessed through the user interface. When a user wants to search for a particular word, the broker transforms this into a sub request which is send to the relevant index. It then returns this response to the user. This is illustrated in Figure 2.1 (adapted from Brouwer et al. 2013b). The indexes and their functionalities have been made available to the Broker using configuration files. The field names that are going to be used are defined and the mapping between these names and the corresponding native names as used within each available index. These mappings are important, because this is what the broker uses to construct sub requests and responses to these requests in interaction with the indexes.



*Figure 2.1. Broker Architecture providing searchability with multiple indexes, collections, caching and additional service in a cloud environment.*

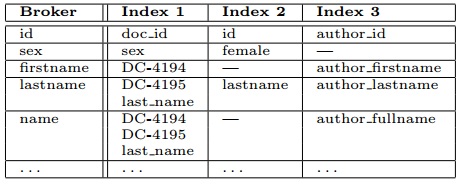
The Nederlab broker uses its own syntax for requests. The broker translates a request into a request consisting of a filter, a condition and a response. A request searching for the first name John will look as follows (adapted from Brouwer et al. 2013b):



*Figure 2.2 Example JSON request searching for first name* John *in* authors *and returning id, first name, prefix and last name for the first 10 results.*

After the request the broker goes to work and first establishes the index that needs to be addressed, which is why an inventory of all the field names used in the filter, condition and response is made. The broker determines whether it is possible to go for a *single-index-solution* and which index should be used. When this index is found, the broker constructs a suitable query using the configured field name mapping. This query is sent as a sub request to the index. A caching mechanism is implemented to increase performance. The response is also analysed and translated by the broker and sent back to the client.

Each index has its own mapping, which allows each index to be mapped to one or more native field names within the index. This process of mapping field names is illustrated in Figure 2.3 (adapted from Brouwer et al. 2013b), in which the broker field name id is mapped to a native field name. Mapping does not have to occur to every index and one-to-many and many-to-many relation are also possible.



*Figure 2.3. Example of mapping field names between* Broker *and indexes.*

The developers of the Nederlab broker have implemented a function which allows the user to store the result of the query as a collection, which can be defined as a list of identifiers for all items in the result. This allows users to temporarily store and handle collections and use them as a basis for a narrower search. This can be seen as a *join* operation. The syntax within the broker was therefore extended with a *join* object to a *filter*. These *filters* with a *join* were also given a sub filter. This allows users to immediately narrow down their search results, without having to store their collections in between queries. However, this can only be implemented in collections of a size smaller than 1,000,000 as otherwise performance will become unacceptable. The implementation of a join operation allows users to narrow down their query in different ways, without having to save the output in between. This means that if we want to search for the word *kip* ‘chicken’ used in fictional work written by women, we can define this in one go and do not need to search for *kip* first, then do a search on this collection for fictional work and then search that collection for work written by women.

The broker is also equipped with a *lexicon service*. This “provides the client with an option to automatically expand an elementary condition, i.e. a certain field having to equal a given string, to the new composite condition on this field to equal the given string or a variant as found with the *lexicon service*” (Brouwer et al. 2013b). To put this in simple words; the collection is enriched with a historical lexicon (developed by the Instituut voor Nederlandse Lexicologie ‘Institute for Dutch Lexicology’ or INL). This means that the user does not need to know all the historical variants of, for instance, *kip*. The broker will automatically return the results for *kippe, kippen, kips, kipse en kyp* as well.

The advantage of the way the broker is currently designed is that it is relatively easy to add infrastructural concepts without having to disturb the user interface. The way it is built up allows an easy integration of additional functionality, such as visuals like n-gram viewers. Nederlab, for instance, has added an R visualisation web server (Komen 2015), which allows a visual outcome of the search results.

1. **Nederlab in practice**

On 13 March 2015 the first version of Nederlab was launched by Antal van der Bosch during the CLARIAH kickoff meeting. From this moment onwards Nederlab became available for the public. The following section will discuss Nederlab and its possibilities in a critical way. It will highlight its strengths and weaknesses by means of a (fairly) simple case study. The study is not meant to find any conclusive answers, but only serves as a backdrop for the evaluation of Nederlab. Before going into Nederlab itself, the possible research questions that can be answered will be discussed, as well as the research question that will serve as a guide for this evaluation.

* 1. **Research questions for Nederlab**

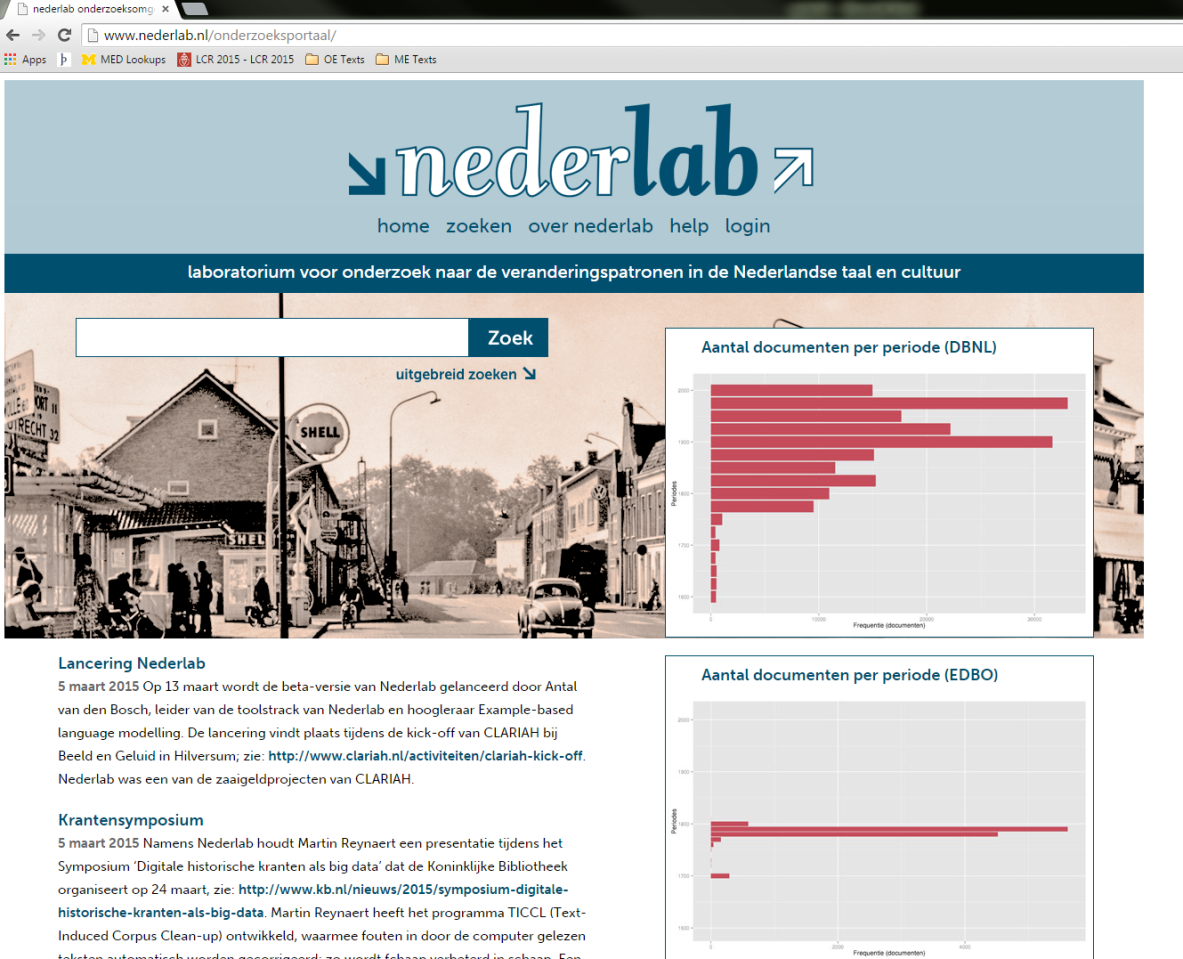
The charm of Nederlab is that because of the wealth of material almost any research question evolving around the history of the Netherlands can be posed. While the options to answer quantitative linguistic questions are at this point limited, since linguistic annotation has not been added, data on sociolinguistic questions can easily be gathered, because of the richness and accessibility of the metadata. This richness does not only allow for sociolinguistic research questions, it opens up possibilities to answer many multidisciplinary research questions in the field of humanities. Nederlab allows anthropologists to explore the relations between people of different nationalities. The Netherlands have had quite a turbulent history of contact with the Spaniards, for instance. The data found in Nederlab might shed light on what the perception of the Dutch was toward the Spaniards, but it is also possible to do this for the English, with which the Dutch were on much more amicable terms. It is possible to distinguish between texts from lay persons, to newspaper accounts and to official documents. Another possibility is to investigate what preceded the passing of a new law and what kind of impact this had on the public debate. Another study might to use Nederlab to find out what was said about slavery before it was abolished, while another may be interested in the opinion of men and women on the emancipation issue.

For the purposes of this paper, I will try to find out what questions Nederlab can answer and will discuss its functionalities along this exploration. One of the characteristics of language is that it sees words come and go. I will explore the history of a word that has disappeared from the Dutch language. Muller (2014) published a list of words that disappeared from the Dutch that, in his view, should be reintroduced and is based on Boon (2006). The article was published in *HP De Tijd* and was published from entertainment rather than research purposes. The article, however, provides a wealth of words no longer productive in Dutch. I have randomly selected a word that I found particularly charming: *bedsermoen* ‘bed sermon.’ A bed sermon used to be a word for the reprimand a woman gives to her husband while in bed. This word has left the Dutch language, but Nederlab might be able to tell us when it appeared, when it disappeared and why and how.

* 1. **Searching in Nederlab**

Nederlab is an open access platform and is accessible for the general public. When entering the *onderzoeksportaal* ‘research portal’ the visitor is welcomed on the homepage (Fig 3.1.), showing the latest news, general information about the corpus and the search engine for simple searches. Via the button *uitgebreid zoeken* ‘elaborate search’ the user can do a more sophisticated search. Nederlab can be browsed by the general public, but the possibilities for further research are limited. Via the login button on the right of the menu users can log in through their institution. Once they are logged in, all Nederlab functionalities become available.

When a registered user opens the *onderzoeksportaal* he sees his saved corpora, as in Fig. 3.2. Since most functionalities are only available to registered users, we will continue our evaluation from here. To do a new search, the user has to go to the search menu at the top of the screen. Three options are presented, which will each be briefly introduced. The first is *zoeken in tekst* ‘search text’. This allows users to find words and combinations of words within texts. The user can simply enter one word, but it is also possible to search for a part of a sentence, which is useful when you want to find the occurrence of a sequence of words. It is also possible to enter a number of words and ask the search engine to retrieve one or more of the words provided. It is also possible to exclude words, so that Nederlab can be queried for one word, but without interference from another.



*Fig. 3.1. Nederlab’s homepage*



*Fig. 3.2. The saved corpora in mijn nederlab ‘my nederlab’*

The user can select whether he wants to search the texts only, or whether he also wants to include searches in the metadata. In addition, it is possible to search for different variants of the words, but searching for the variant entered into search engine alone is the default option. The search can be made case sensitive and it allows the user to enable wildcards. Finally, it is up to the user whether automatically corrected texts need to be incorporated.

The second search option is *zoeken in titelgevens* ‘search titles’. This allows the user to search for words in titles of publications. The third search option is *zoeken in auteursgegevens* ‘search data on authors’, which allows users to find works by particular authors or differentiate between the role of the author. It is also possible to search for texts by male of female authors, to search for authors that were born or died in particular year or were born and lived in a particular period. For some of the texts these dates are estimated and it is possible to only include texts for which the exact dates are confirmed. Texts can be selected on the basis of country of birth, province of birth, place of birth and place of death. One major setback is that there is no possibility to limit your query according to the date of publication. The output of the query can only be restricted in terms of period of publication once the query has been run.

* 1. **Studying output**

For the present paper I would like to find out when *bedsermoen* was used and whether it was used more frequently by men or women. I queried Nederlab for *bedsermoen* in texts only and I included all word varieties. I also included automatically corrected texts. This returned 118 titles. Nederlab does not allow the user to do a query that distinguishes between men and women, so two queries need to be done. One including texts by male authors, another including texts by female authors. Unfortunately, however, filtering the data on gender returns 11 hits for texts written by men and only 2 written by women, which suggests that information of the gender of the author is not available for all texts. Instead, I will compare the use of *bedsermoen* to the use of *vroegpredikatie (doen)* ‘to give an early sermon,’ which is a similar way of indicating that a woman is scrutinising her husband (early in the morning) for his behaviour, so that we can still explore the possibility to compare corpora in Nederlab.

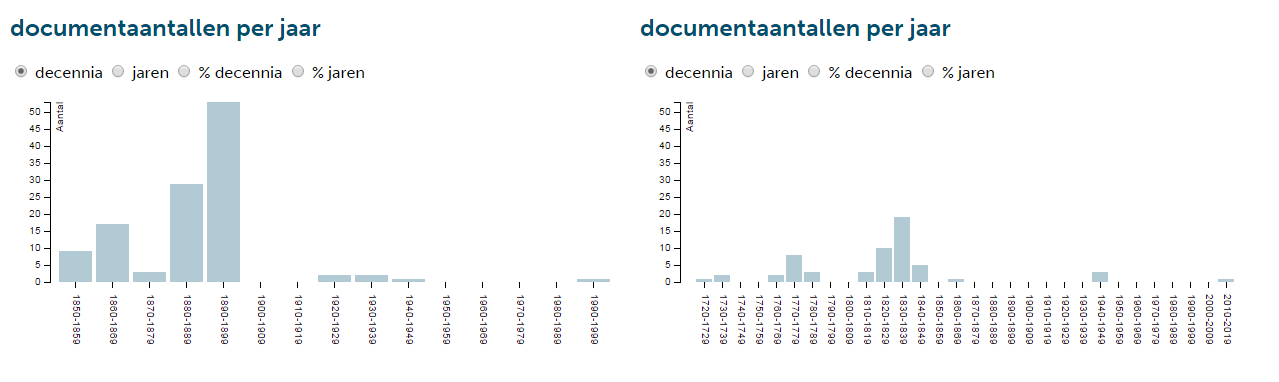
The search on *bedsermoen* yielded 118 titles and the search on *vroegpredikatie* yielded 63 titles. It is important to note that the number of hits that are returned is not necessarily the frequency of the word. It could very well be that a word is used more frequently than is being represented here, which makes Nederlab much less useful for quantitative research. This is also reflected in the way the hits are represented on the screen. The emphasis is on the titles in which the word occurs, rather than the context of the titles, as is illustrated in Fig 3.3[[5]](#footnote-5). It is possible, however, to get a visual representation of the distribution of the data. Unfortunately, this function returned an error at the time of writing.



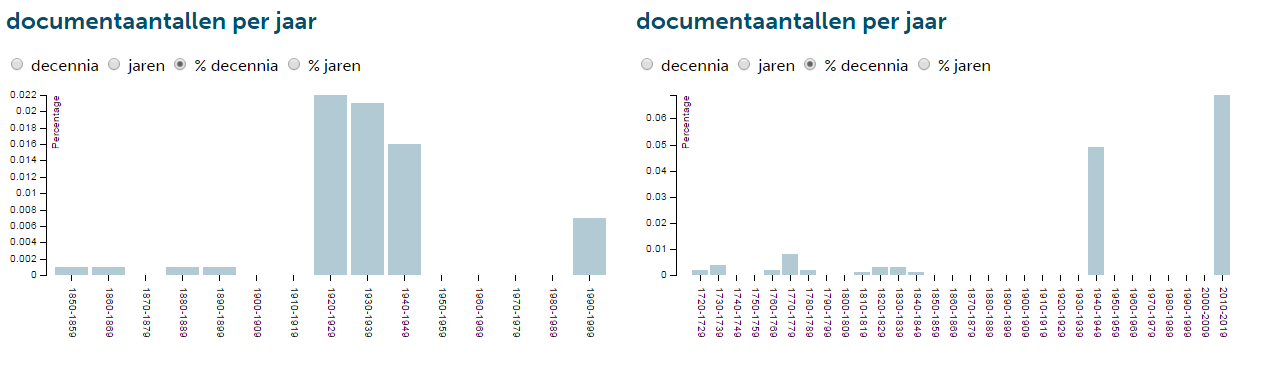
*Fig 3.3. Example of hits of bedsermoen. Instead of focussing on the word in context, the output focuses on the title in which it occurs.*

Furthermore, if we want to study an example in more detail we need to click on the title. We are directed to a page with information on the title, such as the data, the source, the number of pages, when it was entered into Nederlab and when it was last edited. This is followed by the hit(s) with some context and later on the entire text. It is not possible to be immediately directed to the hit in the original text. If the user wants a more elaborate context, he will need to browse the entire text for the particular word. The page is accompanied by links to the original source, Wikipedia and Google Scholar, which is a nice gesture to make additional research on a text easier. In case of OCRed text, the original text as well as the OCRed version is available.

A very useful functionality in Nederlab is the possibility to compare different corpora. It is possible to compare corpora on various levels. First of all, the different number of documents can be compared per year or per decennium. The data can be displayed in non-normalised format or in percentages of the entire Nederlab collection, as is illustrated in Fig. 3.4 and 3.5, but the non-normalised data do not tell us much.



*Fig. 3.4. Number of documents per decennia for* bedsermoen *(left) and* vroegpredikatie *(right)*



*Fig. 3.5 Percentage of documents per decennia for bedsermoen (left) and vroegpredikatie (right)*

Fig. 3.4 shows that *bedsermoen* is used most frequently between 1890-1899, but if we recall the number of documents within Nederlab and their distribution across the years, we see that most of the texts are from the 19th century. This means that the picture that emerges in the comparisons can be severely skewed. *Vroegpredikatie* seems to peak around 1830 – 1839, compared to other dates, but we have to admit that this is exactly the period from which there are most texts available. Converting the graphs to percentages might tell us more. The picture changes dramatically when this option is selected, as is illustrated in Fig. 3.5. The word *bedsermoen* is now used most frequently in the period between 1920-1929 and *vroegpredikatie* is used most frequently between 2010-2019 in terms of percentages of Nederlab as a whole. However, it was used only once in this period. The percentages are, not unexpectedly, very small and hard to read from the graph. It is unfortunately not possible to get an overview of the exact numbers in tabular form. The numbers for these queries are admittedly small, which significantly complicates creating a clear overview within such a large framework.

The second graph on the comparison page compares the distribution of male and female authors, but as noted before, the gender of the authors is mostly unknown. This is very likely because of the fact that many texts include newspaper articles, which do not always specify the author. The comparison function also allows a comparison of the birthdates and dates of death of the authors, as well as where they were born and where they died. The final graph compares the average age of the authors. These graphs have not been included here in the paper, for the sake of space, but the reader is referred to appendix B where they have been included. It becomes clear from these graphs that many data is missing, so it is hard to draw conclusions. For instance, there are only 12 known locations for the data on *bedsermoen*, while there are 10 for *vroegpredikatie*. What they do seem to suggest is that most texts were written in Holland, which is of course not unexpected as most people live in this part of the Netherlands.

A final, more general, note on the comparison possibility is that the timelines are not the same for the two corpora that are being compared. The timelines for each graph have their own interval, making it harder to compare dates. A comparison at first glance is not possible. It has to be noted, however, that the numbers is my two corpora are relatively low and that a study of larger sizes might suffer less from this, as the distribution will become more even.

The comparison function, then, provides a side to side picture of the metadata that is available on the texts in the sub corpora compared. The data that is provided is very useful in terms of the social factors responsible for the use of a particular word. However, the data is to a large extent incomplete, making it hard to draw conclusions from it, especially when the sub corpora are particularly small.

* 1. **General user-friendliness**

Nederlab itself is easy to navigate. The search interface is very easy to use and feels intuitive, and it is immediately clear what possibilities Nederlab has to offer. The only thing that can be a bit daunting to navigate is the ‘homepage’ of a saved corpus. The homepage, illustrated in Fig. 3.6, gives an overview of the title of the corpus, a description, the query, and the dates on which it was created and last modified. The title, description and query are practically the same. It is possible to change the title and description afterwards, but perhaps allowing the user to create a title and description at the moment of saving makes the page where the corpora are collected much more clear, especially when a user has more than one or two corpora in his collection.



*Fig. 3.6. Homepage of the* bedsermoen *corpus*

The list of titles under the header *titels* is in itself of course useful, but the examples that are displayed in it feel superfluous, as the corpus is already a collection of titles, rather than a collection of tokens. For clarity’s sake I would prefer to have this list in tabular format, so that dates, collections and titles can be sorted alphabetically and chronologically. Currently, it is only possible to sort chronologically. The examples are not necessary in this case, since it serves as a list of titles. The examples can be found in the corpus itself. It would also be useful to add metadata to this overview. This would allow the user to – very easily – see whether the text is written by a man or a woman, for instance, or by a particular author, without having to browse the metadata for each and every text, since the metadata overview in *meta-data overzichten* only gives an overview in terms of graphs, which are the same graphs that are used to compare two corpora.

Finally, Nederlab does not always work as it should. During browsing Nederlab often gave an error or was very slow to load results. Especially the more demanding queries and requests lead to problems. This is perhaps an issue that is difficult to avoid, but it does make collecting data a cumbersome task at times.

* 1. **Conclusion**

Currently, the possibilities within Nederlab are limited and mostly aimed at qualitative analysis. However, the vastness of the data and the metadata that are collected suggests that the research questions that can be answered are endless, but have to be qualitative in nature. Nederlab provides the user with all the texts necessary to conduct research on a particular topic, without having to leave through hundreds of books, browse countless libraries or try to combine materials from different sources. It is all there for the user, presented on a silver platter.

This paper was written from the perspective of a syntactician mostly involved in quantitative analysis, which is perhaps unfair for an evaluation of Nederlab, because it still has to demonstrate its usefulness for this type of research. Syntactic annotation and parsing has not yet been made available, which means that it is not possible to, for instance, extract all objects or even only nouns from the collection or to differentiate between words that are ambiguous with regards to their word class, such as *acht* which can either be the numeral *acht* ‘eight’ or the third person singular form of the verb *achten* ‘to deem,’ making Nederlab not the most useful tool to do this type of research. However, this does not take anything from the fact that the materials in Nederlab are a goldmine for anyone interested in the (linguistic) history of the Netherlands.

1. **Possibilities for the future**

Nederlab is the result of a very ambitious wish to collect all textual data related to the Dutch national heritage. It has become an open access web interface which allows the user to search through over 4,5 million documents of text. This amount will only expand and the tools that can be used will only increase. The version that is online is only a beta-version, so it is not unexpected that not all tools have been implemented and that it is not entirely optimalised. In the remainder of this paper I will discuss some of Nederlab’s plans for the future as well as add my own recommendations

Currently, the main aim of the developers seems to have been to get the back-end of Nederlab straight. A project of this size allows careful planning and consideration to ensure the best user experience. This has resulted in a search environment that works well, even though there are still some minor issues that need to be resolved and some details that need to be taken care of.

The issue of quantifying the data has already come up several times within this paper. Nederlab can currently be seen as a librarian who gathers all relevant texts for you based on a word that you feed it with. It still requires you to carefully study the data yourself and extract all the relevant pieces of information. Addressing the data from a different perspective might make Nederlab more useful for a larger group of researcher. The SoNaR reference corpus (Oostdijk et al. 2013) is a similar project which has successfully reached this goal. It has gathered over 500 million words of linguistically annotated contemporary written Dutch and is offered to the public by means of the user-friendly online interface OpenSoNaR. The OpenSoNaR interface was specifically built to cater to the needs of researchers in four areas of research: (corpus and cognitive) linguists, communication and media studies, literary sciences, cultural sciences (Reynaert et al. 2014). This has resulted in an interface that is accessible to all sorts of researchers from all sorts of disciplines. A simple search for the word *kip* gives a wealth of information that can easily be sorted according to ones needs. It is possible to only see the hits in its context, with linguistic information. A simple press of a button adds additional context or the text in which it occurs.

SoNaR, however, deals with contemporary data that is mostly born digital, so the materials much easier to work with than the materials in Nederlab. Nederlab first had to struggle with the quality of digitised materials and with unifying different data sources. However, SoNaR is similar in infrastructure. The search engine that is used is different, however. It is build around the BlackLab search engine, with files in XML format and metadata according to the CMDI framework, which allows searches on linguistically enriched texts. This is not possible with the current infrastructure of Nederlab. Implementing BlackLab as search engine could significantly improve the queries in Nederlab, which would turn Nederlab from the librarian that it is now into an interdisciplinary research institute that can gather and collect all data necessary.

One thing that SoNaR does not have is visualisation tools. SoNaR only provides the user with textual data. Nederlab currently has some visualisation options and it would be a great improvement if these were developed further. Visualisation can be a powerful tool in research, especially when one wants to show change or contrast. The current visualisation possibilities in Nederlab are limited and are not always optimised in terms of performance. For instance, it is possible to get a graph of the distribution of documents across time, but it is not possible to select the range of dates you want to include in your graph. Furthermore, when comparing two corpora, the graphs that are shown are different with regard to the size of the units on the axis, which makes it hard to compare data.

Another very useful feature of the SoNaR corpus is the possibility to download the results as a database. Combined with visualisations researchers will be able to construct pictures of change and variation by a few button presses and mouse clicks. By exporting data the user can annotate the data according to his needs and will be able to do sophisticated statistical analyses. This possibility, however, would be at odds with the advantages of having an online workspace put forward by Brouwer et al. (2013). Nederlab will be a dynamic environment, which means that the data is sometimes subject to change. Keeping the workspace online means that users can be notified once changes have been made which might help to update their searches. Finally, exporting the dataset would also reduce reproducibility, since everything that is done offline cannot be traced, while online changes are always accompanied by a timestamp.

Finally, the dynamic nature of Nederlab allows addition of new materials. Brouwer et al. (2013) propose to make Nederlab an interactive environment by allowing researchers to add their own data sets to the Nederlab collection. The researcher can decide to use the data for personal use in his own workspace or share it with others, but the researcher can also decide to submit his data to the editorial board, which will then provide quality assessment, metadata modification and data harmonization. Van der Sijs (2014b) proposes to also add a corpus of Dutch contact Varieties (CoVaCo). She demonstrates that research into contact varieties not only gives useful insights into the contact varieties themselves, but also into how contact varieties can influence standard Dutch, and vice versa. The aim of Nederlab to include anything related to the Dutch national heritage is of course a very broad one and adding contact varieties to the database might certainly improve our understanding of the history of Dutch language and culture.

1. **Conclusion**

This paper has sought to describe the Nederlab in as much detail as possible, as well as evaluate it and indicate possible improvements for the future. It has shown that Nederlab currently hosts a vast amount of information related to the history of Dutch language and culture and that the open access web interface allows easy retrieval of all texts related to a particular subject. The project is still in its infancy and currently only presents the user with the documents that might be relevant for answering the user’s questions. The tools for handling the data are as of yet limited, but many new functionalities will no doubt be introduced in the future. The ultimate aim is to have created an interdisciplinary research environment by 2017 that can satisfy any type of researcher, from curious individuals to specialised syntacticians like me.

**References**

den Boon, T. (2006). *Verdwijnwoordenboek: Duizend en enige woorden die in de vergetelheid zijn geraakt*. De Vrije Uitgevers.

van den Bosch, A., Busser, G.J., Daelemans, W., and Canisius, S. (2007). An efficient memory-based morphosyntactic tagger and parser for Dutch, In F. van Eynde, P. Dirix, I. Schuurman, and V. Vandeghinste (Eds.), *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, 99-114.

Broeder, D., Kemps-Snijders, M., Van Uytvanck, D., Windhouwer, M., Withers, P., Wittenburg, P., and Zinn, C. (2010). A data category registry- and component-based

metadata framework. In N. Calzolari, B. Maegaard, J. Mariani, J. Odjik, K. Choukri, S. Piperidis, M. Rosner, & D. Tapias (Eds.), *Proceedings of the Seventh conference on International Language Resources and Evaluation*, 43-47.

Brouwer, M., Brugman, H., Droës, M., Kemps-Snijders, M., Kunst, J.P., Tjong Kim Sang, E., Reynaert, M., Zeeman, R., and Zhang, J. (2013a). Nederlab, towards a Virtual Research Environment for textual data. Abstract submitted for The 9th edition of the Language Resources and Evaluation Conference (LREC 2014).

Brouwer, M., Brugman, H., Kemps-Snijders, M., Kunst, J.P., van der Peet, M., Zeeman, R. and Zhang, J. (2013b). Providing Searchability Using Broker Architecture on an Evolving Infrastructure‘. Abstract submitted for The 9th edition of the Language Resources and Evaluation Conference (LREC 2014).

Does, J. & K. Depuydt (2013), ‘Lexicon-supported OCR of eighteenth century Dutch books: a case study’. In: *Electronic Imaging 2013*, *Proceedings of SPIE*, 3-7 February 2013 Burlingame, California, USA, vol. 8658, 86580L-1--14.

Erjavec, T. (2011). Automatic Linguistic Annotation of Historical Language: ToTrTale and XIX Century Slove. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences and Humanities,* 33-38*.*

Gompel van, M. and Reynaert, M. (2013). FoLiA: A practical XML format for linguistic annotation – a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal* 3, 63-81.

Komen, E. (2015). An insider’s guide to the Nederlab R visualization webserver. Retrieved from: http://erwinkomen.ruhosting.nl/nedlab/2015\_NederlabRwebservice.pdf

Muller, N. (2014) Dertig verdwenen woorden die we weer zouden moeten gebruiken. Newspaper article. *HP de Tijd*. Retrieved from: http://www.hpdetijd.nl/2014-12 01/dertig-verdwenen-woorden-die-we-weer-zouden-moeten-gebruikendertig verdwenen-woorden-die-we-eigenlijk-weer-zouden-moeten-gaan-gebruiken/

Oostdijk, N., Reynaert, M., Hoste, V. and Schuurman, I. (2013). The construction of a 500 million word reference corpus of contemporary written Dutch. *Essential Speech and Language Technology for Dutch: Results by the STEVIN-programme* (eds. P Spyns, J. Odijk). Springer Verlag.

Patternson, E., Megyesi, B. and Nivre, J. (2014). A Multilingual Evaluation of Three Spelling Normalisation Methods for Historical Texts. *Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences and Humanities,* 32-41*.*

Reynaert, M. (2008). Non-Interactive OCR Post-Correction for Giga-Scale Digitization Projects. *Computational Linguistics and Intelligent Tekst Processing: Lecture Notes in Computer Science*, 4919, 617-630.

Reynaert, M. (2010). Character confusion versus focus word-based correction of spelling and OCR variants in corpora. International Journal on Document Analysis and Recognition, 14, 173–187.

Reynaert, M. (2014a). On OCR ground truths and OCR post-correction gold standards, tools and formats. *DATeCH ’14 Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, 159-166.

Reynaert, M. (2014b), ‘Synergy of Nederlab and @PhilosTEI: diachronic and multilingual Text-Induced Corpus Clean-up‘, in: Proceedings of the Ninth International Language Resources and Evaluation Conferencen Nicoletta et al. (ed.), 1224-1230.

Reynaert, M., van de Camp, M., and van Zaanen, M. (2014). OpenSoNaR: user-driven development of the SoNaR corpus interface. Proceedings of COLING 2014, the 25th Internation Conference on Computation Linguistics, 124-128.

Tjong Kim Sang, E. (2014), Verwerking van achttiende-eeuws Nederlands met Frog, internal report.

van der Sijs, N. (2014a). De voortzetting van de historische taalkunde met andere middelen, inaugural speech Radboud Universiteit Nijmegen.

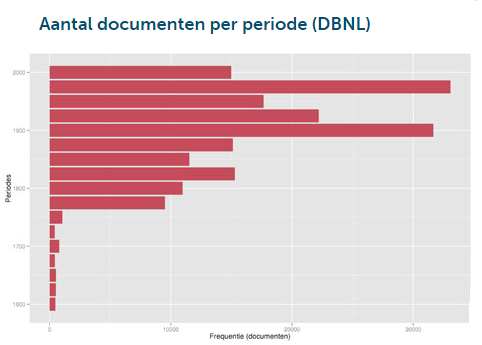
van der Sijs, N. (2014b). Systematisch onderzoek naar Nederlandse contactvariëteiten. Taal & Tongval 66, 117-131.

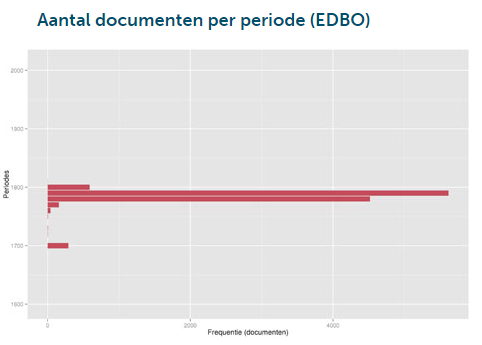
Sushmita, S., Joho, H. and Lalmas, M. (2009). A Task-Based Evaluation of an Aggregated Search Interface. *String Processing and Information Retrieval: Lecture Notes in Computer Science*, 5721, 322-333.

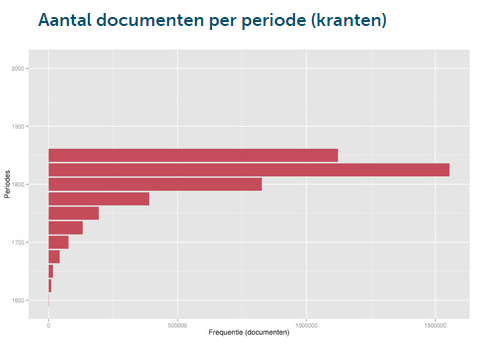
Thomas, P. (2012). To what problem is distributed information retrieval the solution? *Journal of American Society for Information Science & Technology*, 63.1, 1471-1476.

**Appendix A.**

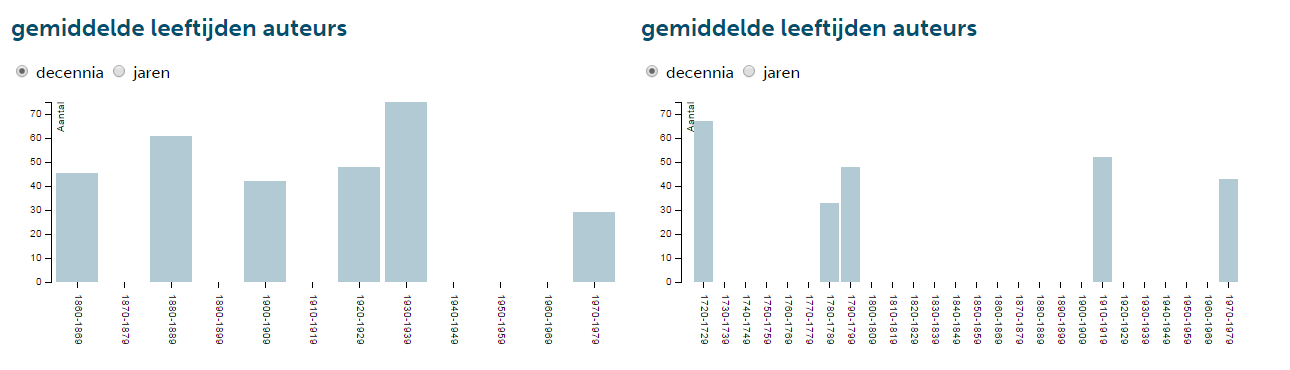
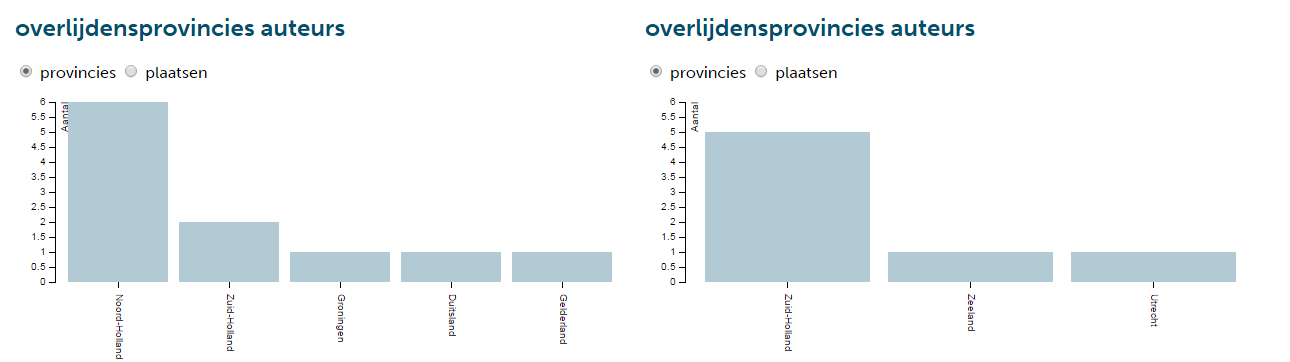
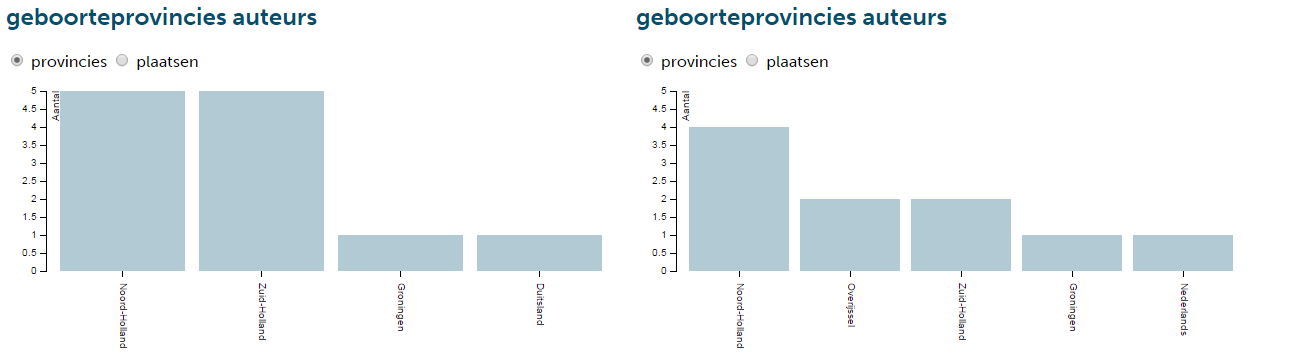
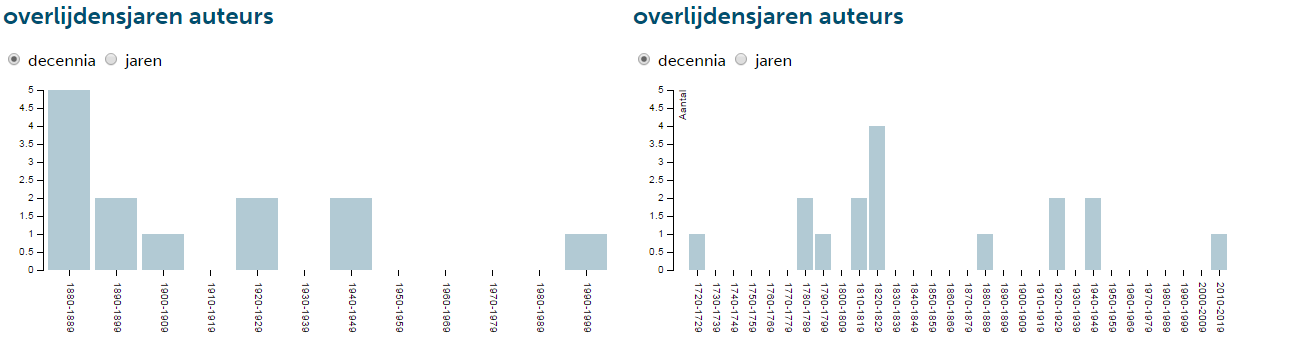
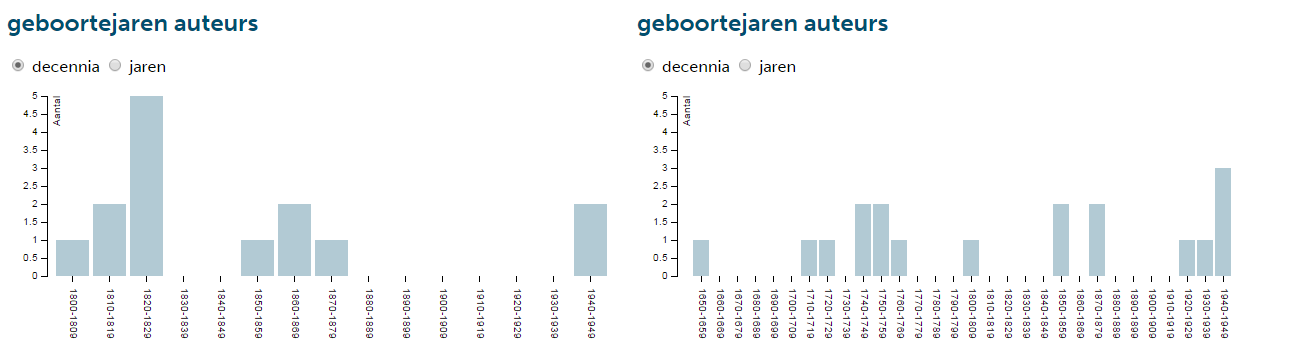
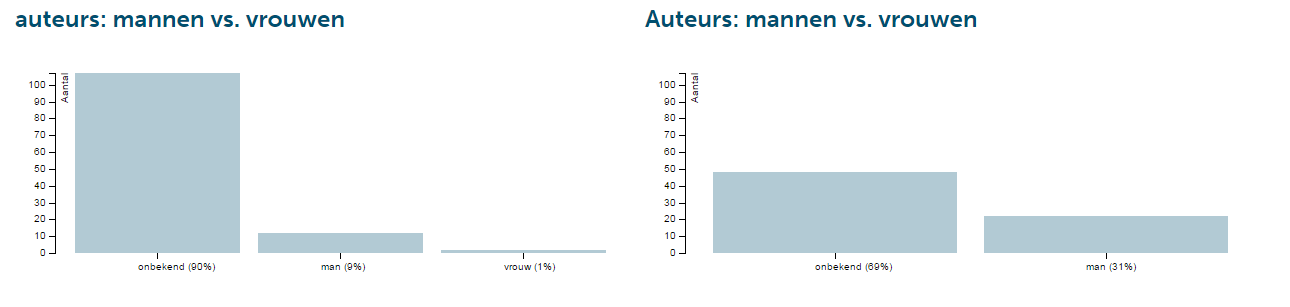
The following graphs show the distribution of texts from three different text collections.







**Appendix B.**

The graphs below compare the gender of the authors, their year of birth and year of death, their province of birth and province of death and their average age. These graphs clearly show that many metadata is missing and that more data is necessary to acquire more details about the use and disappearance of *bedsermoen* and *vroegpredikatie*.

1. http://www.nederlab.nl [↑](#footnote-ref-1)
2. The remainder of the paper is based on the first version of Nederlab that was launched. [↑](#footnote-ref-2)
3. For a full specification of the core metadata schema see https://www.nederlab.nl/wp/?page\_id=180 [↑](#footnote-ref-3)
4. http://lucene.apache.org/solr/ [↑](#footnote-ref-4)
5. The current version of Nederlab allows the user to immediately view the word in its context. [↑](#footnote-ref-5)