

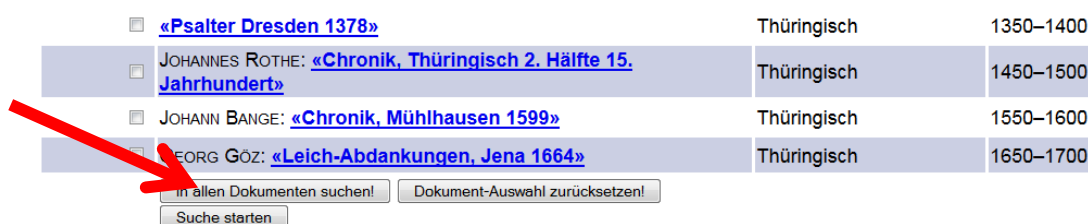
Konkordanzen aus dem Bonner Frühneuhochdeutschkorpus exportieren

Dieses Tutorial ist veraltet, da das Bonner Frühneuhochdeutschkorpus inzwischen in ANNIS verfügbar ist. Allerdings funktionieren dort derzeit (Stand November 2017) noch nicht alle Exporter bei allen Anfragen einwandfrei, weshalb ich das Tutorial zunächst noch online lasse. Das alte FnhdC-Interface findet sich unter https://korpora.zim.uni-due.de/Fnhd_alt/.

Das Bonner Frühneuhochdeutschkorpus (FnhdC) ist derzeit eine der besten Ressourcen, die wir zu dieser Periode der deutschen Sprachgeschichte haben. Einer der größten Vorteile besteht darin, dass es von Hand annotiert wurde und daher das Tagging nicht so fehlerbehaftet ist wie bei automatisch getaggten Korpora. Ein Nachteil des alten Interface besteht darin, dass man nur nach einzelnen Wörtern suchen und keine komplexen syntaktischen Suchanfragen stellen kann. Auch gibt es derzeit leider keine native Möglichkeit, die Konkordanzen in Tabellenform zu exportieren. Daher empfiehlt es sich für viele Zwecke, die im xml-Format verfügbaren Korpusdateien herunterzuladen und damit zu arbeiten. Dies erfordert jedoch ein wenig Einarbeitung. Meines Wissens ist derzeit eine ANNIS-Version des Bonner Frühneuhochdeutschkorpus in Arbeit; darüber hinaus entsteht im Moment das Referenzkorpus Frühneuhochdeutsch, das sicherlich bald zur neuen Standardressource für die Untersuchung dieser Sprachperiode werden wird. Für alle, die nicht so lange warten können und die nicht mit den heruntergeladenen FnhdC-Daten arbeiten können oder wollen, habe ich ein Skript geschrieben, das die Daten aus der Suchschnittstelle des FnhdC direkt in Tabellenform überträgt. Es findet sich im Ordner „FnhdC“ und trägt den Namen *getFnhdC*.¹

1. Suche im FnhdC-Interface

Zunächst rufen wir unter <http://www.korpora.org/fnhd/Suche> das Such-Interface auf. Hier sind alle Texte des Korpus aufgelistet. Wir wählen „In allen Dokumenten suchen“.



<input type="checkbox"/>	«Psalter Dresden 1378»	Thüringisch	1350–1400
<input type="checkbox"/>	JOHANNES ROTHE: «Chronik, Thüringisch 2. Hälfte 15. Jahrhundert»	Thüringisch	1450–1500
<input type="checkbox"/>	JOHANN BANGE: «Chronik, Mühlhausen 1599»	Thüringisch	1550–1600
<input type="checkbox"/>	GEORG GÖZ: «Leich-Abdankungen, Jena 1664»	Thüringisch	1650–1700

Tipp: In einigen Fällen kann es sinnvoll sein, nur in einem Teil der Dokumente zu suchen, gerade wenn man alle Belege für relativ frequente Tokens finden möchte. Das Such-Interface gibt nämlich nur max. 800 Belege aus.

Als Beispielanfrage suchen wir das Lemma „Wort“.

¹ Weil das Bonner Frühneuhochdeutschkorpus in seiner jetzigen Form ohnehin bald überholt sein wird, habe ich keine Funktion dafür ins R-Paket „concordances“ integriert, sondern ein separates Skript erstellt.

Wenn Sie eine Liste der 70449 Wortform-Types sehen wollen, kann es lange dauern, bis die Seite geladen und dargestellt ist.

Wortform: Wortform als **Muster** angeben
Wortform-Feld anpassen (Muster/Liste) Suchen

Wenn Sie eine Liste der 8238 Lemma-Types sehen wollen, kann es lange dauern, bis die Seite geladen und dargestellt ist.

Lemma: Lemma als **Muster** angeben
Lemma -Feld anpassen (Muster/Liste) Suchen

Numerus:
(egal)
singular
plural
unbekannt

Wie Sie sehen, wenn Sie nach unten scrollen, sind durchaus auch komplexere Suchanfragen möglich, z.B. nach Numerus (bei Substantiven) oder Tempus und Modus (bei Verben). Für unsere Beispielanfrage suchen wir aber einfach nach allen Belegen für das Lemma „Wort“ und klicken auf „Suchen“ rechts neben dem Lemma-Feld. Es erscheint die Ergebnisseite (eventuelle Warnmeldungen zu Javascript oder Cookies ganz oben können Sie ignorieren, solange Sie die Ergebnisse sehen):

FnhdC/S

Such-Resultate (maximal 800):

Die syntaktischen Informationen einzelner Wortformen erscheinen beim Überfahren mit der Maus. Zur Bedeutung von Hervorhebungen s. [[FnhdC.HTML.Auszeichnungen.html](#)].

- Gefunden in [Wilhelm Durandus: „Rationale, Wien 1384“ \(Gegend: Mittelbairisch \(Wien\), Zeitstufe: 1350–1400\):](#)
[Seite 1] ¹⁴ vercherent der gerechten **wort**. Darumb auch bei alter romischer macht
- Gefunden in [Wilhelm Durandus: „Rationale, Wien 1384“ \(Gegend: Mittelbairisch \(Wien\), Zeitstufe: 1350–1400\):](#)
[Seite 2] ⁰² Von dem ob geschriben **worte**, edler furste, habt ir ewch auzgenomen,
- Gefunden in [Wilhelm Durandus: „Rationale, Wien 1384“ \(Gegend: Mittelbairisch \(Wien\), Zeitstufe: 1350–1400\):](#)
[Seite 2] ⁰⁵ lere, wann ich, wenn ir habt gehoret und auch aufgenomen daz **wort**
- Gefunden in [Wilhelm Durandus: „Rationale, Wien 1384“ \(Gegend: Mittelbairisch \(Wien\), Zeitstufe: 1350–1400\):](#)
[Seite 4] ¹¹ ist: Himel und erde werdent zergan, di **gotes worte** pleibent
- Gefunden in [Wilhelm Durandus: „Rationale, Wien 1384“ \(Gegend: Mittelbairisch \(Wien\), Zeitstufe: 1350–1400\):](#)
[www.google.de](#) ¹² **wort** ist: Das wîr di schêft der **gotes worte** als **worte** und auch

2. Seiten-Quelltext anzeigen und exportieren

Wir müssen uns nun den Seitenquelltext anzeigen lassen. In den meisten Browsern (z.B. Firefox, Chrome) geht dies ganz einfach über Rechtsklick > Seitenquelltext anzeigen.

Hinweis: Wenn Sie über den Link auf <http://www.korpora.org/fnhd> auf die Seite gelangt sind und **nicht** den oben angegebenen direkten Pfad <http://www.korpora.org/fnhd/Suche> benutzt haben, müssen Sie sich statt des Seitenquelltexts den **Frame-Quelltext** anzeigen lassen (in Firefox unter: Rechtsklick > Aktueller Frame > Frame-Quelltext anzeigen).

Den kompletten Quelltext **kopieren** wir nun (Strg+A, Strg+C) und fügen ihn in ein neues Textdokument ein (in Notepad++ bzw. TextWrangler oder einem anderen guten Texteditor; vgl. dazu Tutorial „01-Grundlegendes“).

Wichtig: Bitte benutzen Sie nicht Datei > Speichern, um den Seitenquelltext zu exportieren; dann funktioniert das Skript, das unten vorgestellt wird, nämlich nicht. Bitte copy&pasten Sie den Quelltext in das Textdokument.

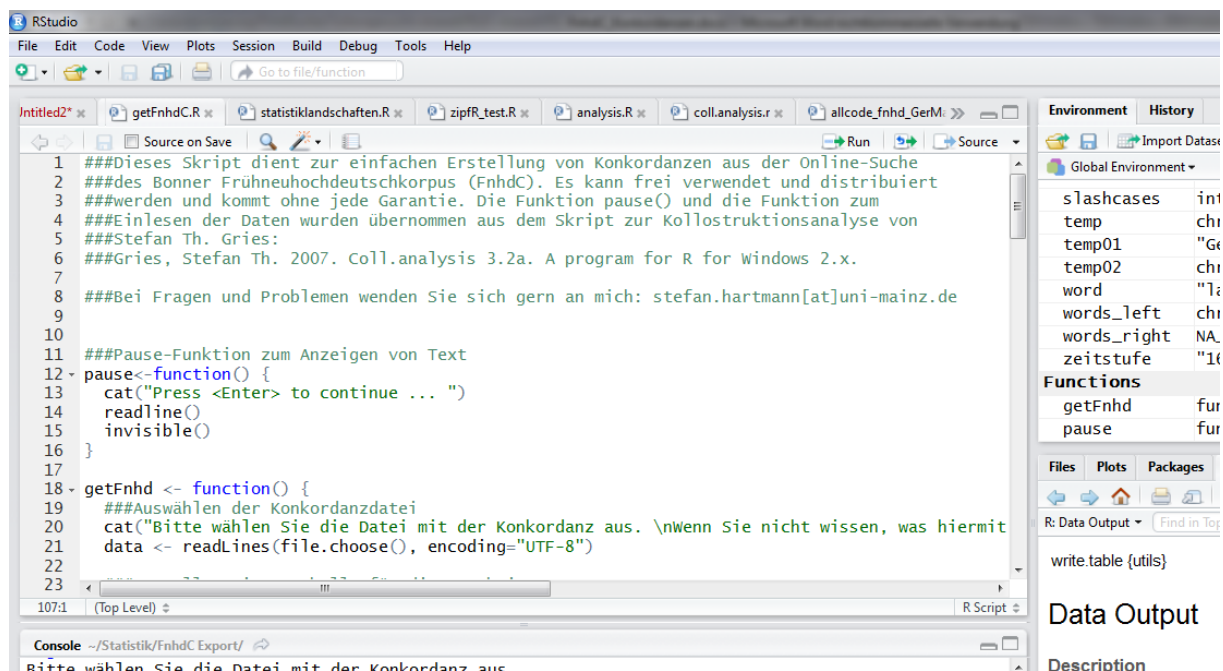
Wir speichern das Textdokument unter einem beliebigen Namen, z.B. wort.txt.

3. R herunterladen und installieren

Zum Ausführen des oben erwähnten Skripts benötigen Sie R; zusätzlich empfehle ich noch RStudio (siehe dazu Tutorial „01-Grundlegendes“). Dort können Sie das Skript einfach öffnen und ausführen können. (Alternativ können Sie auch unter <http://goo.gl/B3pbZM> angegebenen Code einfach in R copy&pasten.

4. Skript ausführen

Wenn Sie R Studio heruntergeladen haben, können Sie darin ganz einfach die Datei „getFnhdC.R“ öffnen. Sie wird Ihnen nun im Fenster links oben angezeigt:



```
1  ###Dieses Skript dient zur einfachen Erstellung von Konkordanzen aus der Online-Suche
2  ###des Bonner Frühneuhochdeutschkorpus (FnhdC). Es kann frei verwendet und distribuiert
3  ###werden und kommt ohne jede Garantie. Die Funktion pause() und die Funktion zum
4  ###Einlesen der Daten wurden übernommen aus dem Skript zur Kollostruktionsanalyse von
5  ###Stefan Th. Gries:
6  ###Gries, Stefan Th. 2007. Coll.analysis 3.2a. A program for R for Windows 2.x.
7
8  ###Bei Fragen und Problemen wenden Sie sich gern an mich: stefan.hartmann[at]uni-mainz.de
9
10
11 ###Pause-Funktion zum Anzeigen von Text
12 - pause<-function() {
13   cat("Press <Enter> to continue ... ")
14   readline()
15   invisible()
16 }
17
18 - getFnhd <- function() {
19   ###Auswählen der Konkordanzdatei
20   cat("Bitte wählen Sie die Datei mit der Konkordanz aus. \nWenn Sie nicht wissen, was hiermit
21   data <- readLines(file.choose(), encoding="UTF-8")
22
23
```

Das Fenster links oben zeigt **Skripts** an. Befehle, die in einem Skript gespeichert sind, kann man sich wie Kugeln in einer Kanone vorstellen: Sie tun nichts, solange sie nicht abgefeuert werden. Um die Befehle „abzufeuern“, müssen sie in die **Konsole** übertragen werden, die unten links zu sehen ist. Da das Skript recht lang ist, wäre es natürlich sehr mühsam, jede einzelne Zeile auszuführen. Zum Glück können wir einfach mit Strg+A den gesamten Text markieren und mit Klick auf „Run“ über dem Skriptfenster oder mit Strg+Enter komplett ausführen.

Hinter den Kulissen (Diese Passage können Sie überspringen, wenn Sie das Skript einfach nur benutzen wollen, ohne genau zu wissen, wie es funktioniert)

Das Skript besteht aus drei Teilen:

Teil 1 definiert eine Funktion *pause()*, die innerhalb der nächsten Funktion verwendet wird. Die Funktion ist aus Gries (2007) übernommen.

Teil 2 definiert die für uns relevante Funktion *getFnhdC()*.

Teil 3 besteht nur aus dem Befehl *getFnhdC()* und sorgt dafür, dass die Funktion ausgeführt wird.

5. Daten bearbeiten

Am Ende wird die im Skript definierte Funktion automatisch ausgeführt. Nun spielt sich alles in der **Konsole** (unten links) ab. Dort werden Sie aufgefordert, eine Datei auszuwählen.

```
87 metacases <- grep('<', output$Keyword)
88
89 - for(j in 1:length(metacases)) {
90   output$Keyword[metacases[i]] <- unlist(strsplit(output$Keyword[metacases[i]], "<"))[1]
91   "
113:1 (Top Level) ↕ R Script

Console ~/Statistik/FnhdC Export/ ↻
+ Bitte beachten: Wenn Sie eine BEREITS EXISTIERENDE Datei wählen, wird diese /n
+ ÜBERSCHRIEBEN. Es erfolgt keine Rückfrage, ob Sie die Datei überschreiben wollen."); pause()
+ write.table(output, file.choose(), sep="\t", row.names=F, quote=F, fileEncoding = "UTF-8")
+ }
+ }
>
> ###Funktion ausführen
> getFnhdC()
Bitte wählen Sie die Datei mit der Konkordanz aus.
Wenn Sie nicht wissen, was hiermit gemeint ist,
konsultieren Sie bitte die Anleitung, verfügbar unter
Um weiterzumachen, drücken Sie bitte <Enter>...
```

Wichtig: Bevor Sie die Eingabetaste drücken, wechseln Sie zunächst mit einem Mausklick in das Konsolenfenster, denn sonst wird der gesamte markierte Text im Skriptfenster gelöscht (wenn das passiert ist, einfach mit Strg+Z wieder rückgängig machen). Wenn Sie im Konsolenfenster die Eingabetaste gedrückt haben, öffnet sich ein Fenster, in dem Sie eine Datei aussuchen können. Navigieren Sie zu dem Ort, wo Sie *wort.txt* gespeichert haben, und wählen Sie es aus.

Nun rechnet R kürzere oder längere Zeit, dann erscheint wieder ein Text in der Konsole, der Sie auffordert, eine Datei auszuwählen, in der Sie den Output speichern möchten.

```

82 output$Lemma[(output_length)+1] <- lemma
83 output$POS[(output_length)+1] <- pos
84
85
86 ###Fälle finden, in denen noch Metadaten beim Keyword stehen, und diese entfernen
87 metacases <- grep('<', output$Keyword)
88
89 for(j in 1:length(metacases)) {
90   output$Keyword[metacases[j]] <- unlist(strsplit(output$Keyword[metacases[j]], "<"))[1]
91 }

```

113:1 (Top Level) R Script

Console ~/Statistik/FnhdC Export/

```

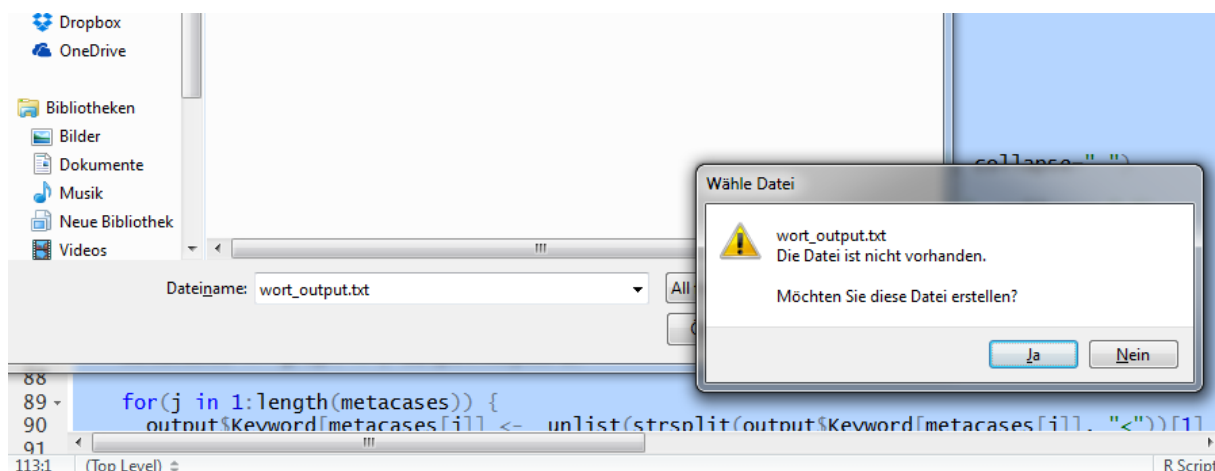
>
> ###Funktion ausführen
> getFnhd()
Bitte wählen Sie die Datei mit der Konkordanz aus.
Wenn Sie nicht wissen, was hiermit gemeint ist,
konsultieren Sie bitte die Anleitung, verfügbar unter
Um weiterzumachen, drücken Sie bitte <Enter>...

Bitte wählen Sie eine Datei aus, in der Sie die Ergebnisse
speichern möchten.
Bitte beachten: Wenn Sie eine BEREITS EXISTIERENDE Datei wählen, wird diese /n
ÜBERSCHRIEBEN. Es erfolgt keine Rückfrage, ob Sie die Datei überschreiben wollen. Um weiterzumachen, drücken Sie bitte <Enter>...
|

```

Nachdem Sie erneut die Eingabetaste gedrückt haben, erscheint ein Fenster, in dem Sie eine Datei auswählen können, in der die erstellte Tabelle gespeichert wird. Dazu zwei wichtige Hinweise:

1. Sie müssen die **Dateiendung** mit angeben, z.B. „wort_output.txt“ oder „wort_output.csv“.
2. Wenn Sie eine **bereits bestehende** Datei verwenden, wird diese **überschrieben!** R fragt nicht vorher nach. Hingegen fragt es nach, wenn Sie eine neue Datei erstellen:



Für unser Beispiel erstellen wir nun also die Datei wort_output.txt, die wir wiederum in Notepad++ öffnen.

(Es erscheint möglicherweise noch eine Warnung, die wir jedoch ignorieren können.)

6. Letzte Schritte

Nun sollten Sie in wort_output.txt die Belege für „Wort“ in einer tab-separierten Tabelle sehen, die Sie einfach in ein Tabellenkalkulationsprogramm wie Excel oder Calc copy&pasten können.

Hinweis: Die in FnhdC relativ häufigen Sonderzeichen gehen beim Export leider teilweise verloren, deshalb werden Sie relativ häufig Unicode-Nummern im Format [U+Nummer] in der Konkordanz finden.

Literatur

Gries, Stefan Th. (2007): Coll.Analysis 3.2a. A program for R for Windows 2.x