Inhaltsverzeichnis

stylo-ah-online Handbuch 2
Ursprung: stylo-ah-online - stylo ohne Installation 2
Links zum Thema 2
Implementierte Analyse-Pipeline(s) 2
Funktionsüberblick 2
Benutzung 3
GUI Konzept 3
Überblick zur Benutzung 4
1. Konfiguration des Browsers 4
2. Konfigurieren der Analyseschritte 4
2.1 Arbeitsschritt: Benennung 4
2.1.1 Auswahl und Anzahl der Zeichen 5
2.1.2 Mögliche Bestandteile im Dateinamen 5
Typ des Dokumentes: 5
Subjekt: 5
Status: 5
ID: 6
Datum: 6
Versionsnummer: 6
Autorname: 6
Dateiendung: 6
2.2 Arbeitsschritt: Configurations-Datei / Datenbank 6
Input eines Config files 7
Mehrer Config files 7
Serien von config files 7
Reset des Tools 8
2.3 Arbeitsschritt: Input / Replication 8
2.4 Arbeitsschritt: Normalisierung (Normalization) 9
2.4.1 Word masking / stop words 9
2.4.2 Sign equalization 9
2.4.3 Markup / Format 10
2.4.4 Word level conversions 10
2.4.5 Combinations 10
2.4.6 Transliteration 10
2.5 Arbeitsschritt: Zerlegung (features / decomposition / token) 10
2.5.1 Word level decomposition 11
2.5.2 General N-Gram decomposition 11
2.6 Arbeitsschritt: Zählung (Selection / Counting) 12
2.6.1 Most frequent token / words (per text) 12
2.6.2 Culling 13
2.7 Arbeitsschritt: Maßanwendung (Measure selection) 13
2.8 Arbeitsschritt: Gruppierung (Clustering) 13
2.9 Arbeitsschritt: Export 14
3. Erneute Berechnung und Serien 15

stylo-ah-online Handbuch

Ursprung: stylo-ah-online - stylo ohne Installation

Im Rahmen unserer Lehr- und Forschungstätigkeit haben wir das R-Paket **stylo** eingesetzt. Wir haben dieses mittels R und Python weiterentwickelt. Vornehmlich vier Dinge sind an der R-Implementierung schwierig:

1. Die Möglichkeit zur Selbstdokumentation der Experimente ist gering.
2. Die Interaktion zwischen Python und R ist problematisch.
3. Die Möglichkeit eines einfachen Einsatzes von Multiprocessing ist fraglich.
4. Die Software ist teilweise schwierig zu installieren und erscheint langsam.

Daher wollten wir die Funktionalität von **stylo** nachempfinden; ohne Installation aber mit erweiterten Dokumentations- und Vergleichsmöglichkeiten.

Links zum Thema

1. stylo das Original in R geschrieben.
2. stylo ah die R / Python Kopie mit zusätzlichen Funktionen für klassische Texte.
3. stylo-ah-online die originale Kopie in JavaScript implementiert.

Implementierte Analyse-Pipeline(s)

Die Implementierte Textanalyse verläuft in sieben Schnitten. Diese Schritte stellen die Abfolge bzw. die Pipeline der Verarbeitung dar:

1. Auswahl der Files (Corpus) (jedes File repräsentiert einen Text)
2. Normalisierung (Formatanpassung, Zeichenvereinheitlichung, Löschung von Struktur und Metadaten, Maskierung von Wortformen)
3. Zerlegung in Token (Wortformen, Silben, N-Gramme, Zeichen)
4. Zählung / Vektorbildung (01-Kodierung, absolute und relative Häufigkeit, TF-IDF)
5. Auswahl aus den Vektoren (Beschränkung der Häufigkeitsliste, Häufigkeitsfenster, Culling)
6. Anwendung eines Maßes
7. Anwendung einer Cluster-Methode

Funktionsüberblick

In stylo-ah-online sind folgende Funktionen verfügbar (Stand 14.11.2023):

Normalisierung:

Normalform Wortmaskierung (Stoppwörter) Vereinheitichen bestimmter Zeichen UV-Angleich JI-Angleich Markup-Löschen Interpunktion löschen Zeilenumbrüche löschen Elision auflösen Alpha

privativum behandeln Entfernen der Nummerierung Worttrennungen zusammenführen Vereinheitlichung des Iota subscriptum und Iota adscriptum Veränderung des Abschluss-Sigmas Diakritische Zeichen löschen Ligaturen auflösen Kleinschreibung Klammern entfernen

Normalisierung

Zerlegung (Token):

Zusätzlich ohne Konsonanten Zusätzlich ohne Vokale Zusätzlich lediglich kleine Wörter Zusätzlich lediglich große Wörter Zerlegung in Wortformen Zerlegung in Buchstaben-N-Gramme Zerlegung in Buchstaben-N-Gramme der Worformen Zerlegung in Wort-N-Gramme Zerlegung in n-Gramme mit Lücken Zerlegung in Pseudo-Silben Zerlegung in Kopf Körper und Coda Zerlegungen in alle Permutationen von Kopf Körper und Coda

Zerlegung

Zählung:

absolute Häufigkeit relative Häufigkeit 0-1-Codierung TF-IDF Beschränkung/Spannen der Frequenzlisten (nach Rang, min-max-Angabe) Culling

Zählung

Maße:

euclidean, chebyshev, minkowski, manhatten, canberra, soerensen, gower, soergel, lorentzian, intersection, wavehedges, motyka, ruzicka, tanimoto, innerproduct, harmonicmean, cosine, kumar hasse brook, dice, fidelity, bhattacarya 1, bhattacarya 2, hellinger, jensen, jensen shannon, topsoee, kullback divergence, jeffreys, kullback leibler, squared euclidean, pearson chi squared, neyman chi squared, squared chi squared, divergence, clarck, additive symmetric chi squared, eder simple, burrows delta, argamon linear delta, eders delta, argamons quadratic delta, wasserstein 1d

Maße

Gruppierung:

hierarchische Clusterung, multidimensional scaling (MDS), tSNE (t-distributed stochastic neigbor embedding)

Gruppierung

Benutzung

GUI Konzept

Jeder Verarbeitungsschritt wird in einem Konfigurationsschritt eingestellt. Jeder Konfigurationsschritt besitzt ein zusätzliches Kommentarfeld und eine der Anzeige der Zwischenergebnissen (rechte Spalte). Die GUI empfindet die Schritte der Textanalyse nach. Einzige Ausnahme bildet die Auswahl des Corpus; die ist zwar an oberer Stelle angezeigt, aber von der Benutzung her ist das der letzte Schritt! Jedes GUI-Element ist mit einer Beschriftung versehen. Sofern die Funktion einer weiteren Erklärung bedarf, ist zusätzlich eine kursive Erklärung angefügt.

Überblick zur Benutzung

1. Konfiguration des Browsers
2. Konfigurieren der Analyseschritte
3. Erneute Berechnung und Serien

1. Konfiguration des Browsers

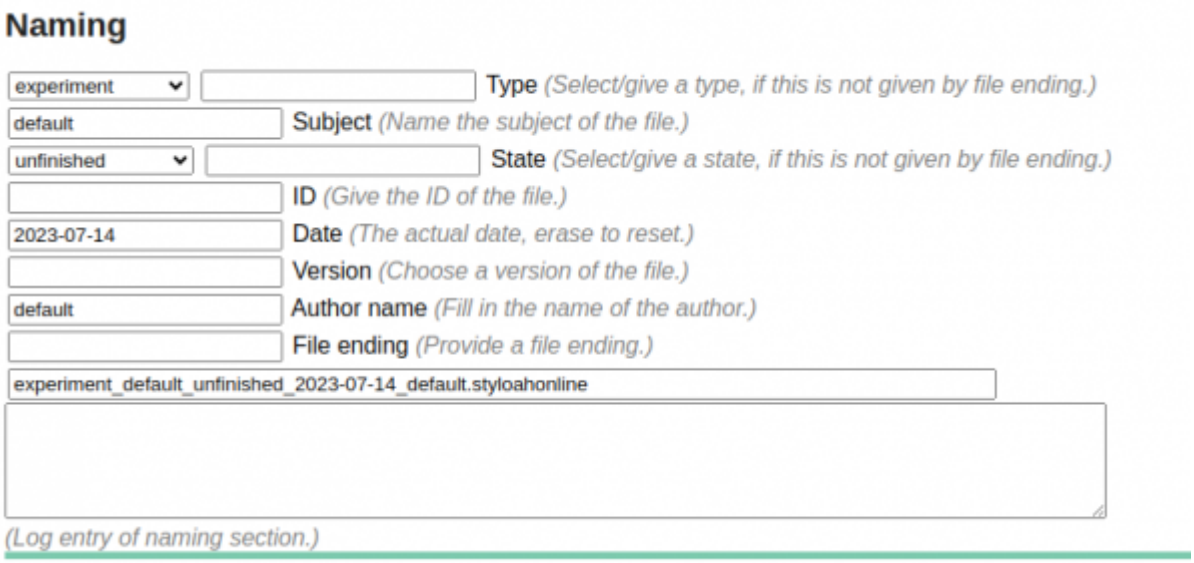
Speicherort: Legen Sie einen Ordner an, in welchem die Ergebnisse von stylo-ah-online gespeichert werden können. Konfigurieren Sie ihren Browser so, dass er die Dowloads in diesem Ordner ablegt. Die Anleitung für Firefox:
https://support.mozilla.org/de/kb/suchen-und-verwalten-heruntergeladener-dateien#w_ziel-ordner-der-heruntergeladenen-dateien-andern

Datenbanken: Teilen sie dem Browser mit, dass er Datenbanken anlegen und nicht löschen soll. Dazu stellen Sie sicher, dass die Chronik angelegt wird:
https://support.mozilla.org/de/kb/firefox-chronik-zeigt-ihre-besuchten-webseiten

Web-Konsole: stylo-ah-online ist in JavaScript geschrieben. Die Web-Konsole bietet eine Darstellung von Meldungen zum Programmablauf und Hinweise auf Fehler. Nach einem Fehler können sie die Seite neu laden und den Ablauf erneut starten. Es bietet sich an die Web-Konsole anzuzeigen. Für Firefox aktiviert man diese so:
https://firefox-source-docs.mozilla.org/devtools-user/tools_toolbox/index.html

2. Konfigurieren der Analyseschritte

2.1 Arbeitsschritt: Benennung



Die Benennung beinhaltet zusammengesetzte Namen für die Experimente. Dabei werden die einzelnen Teile der Benennung durch Unterstriche abgetrennt. Leerzeichen in Einträgen werden durch Bindestriche ersetzt. Die einzelnen Einträge der Benennung erlauben verschiedene Organisationsmuster für Experimente. Das Datum wird beim ersten Programmlauf automatisch

ausgefüllt. Der einmal erstellte Name charakterisiert den Programmablauf und die Einstellungen genau. Änderungen in der Benennung führt auf ein neues Experiment, welches auch separat gespeichert wird. Die Benennung geht in die Config-Datei ein und ein erneuter Aufruf einer Config-Datei stellt alles unter dieser Bezeichnung wieder her. Auf diese Weise können mehrere Konfigurationen unterschieden werden.

2.1.1 Auswahl und Anzahl der Zeichen

- Dateinamen sollten so lang wie nötig und so kurz wie möglich gehalten werden.
- Es werden nur **Kleinbuchstaben [a-z]** und **Zahlen [0-9]** sowie der **Unterstrich [_]** und **Bindestrich [-]** verwendet. Für von der Software automatisch benannte Dateien wird als Ausnahme die Zeichenklasse [A-Z] zugelassen.
- Es dürfen nicht benutzt werden Sonderzeichen, Umlaute, ß.
- Altgriechische Bezeichnungen zu Werken oder Autoren werden ins lateinische transliteriert, falls sie nicht über die entsprechende Corpus-ID kenntlich gemacht werden können. Die Corpora-IDs sind immer den Namen vorzuziehen, weil sie gegebenenfalls einen Verweis auf die benutzte Edition liefern und meist kürzer sind.
- Aus Kompatibilitätsgründen sollte die MAX_PATH Einstellung von 260 bei Windows nicht überschritten werden, dass bedeutet, Ordnerstruktur + Dateinamen sollten maximal aus 256 Zeichen bestehen, da die Laufwerksbezeichnung und das unsichtbar endende Nullzeichen mitgezählt werden. Der maximale Pfad auf einem Windows-Laufwerk D wäre demzufolge „D:\some_256-zeichen-path_string<NUL>“. Bei Unicode-Pfaden sind es gegebenenfalls 255 Zeichen oder weniger. Deswegen wäre eine **Obergrenze von 250 Zeichen** für die gesamte Pfadangabe (Dateiname + Ordnerstruktur) festzulegen.

2.1.2 Mögliche Bestandteile im Dateinamen

Typ des Dokumentes:

beschluss / protokoll / rechnung / notiz / skizze etc. Ein Typ ist immer dann anzugeben, wenn er sich nicht zwangsläufig aus der Dateiendung ergibt. Bei den Endungen jpg oder png beispielsweise den Typ „foto“ voranzustellen, ist überflüssig und widerspricht einer Regel aus Punkt 1 (so kurz wie möglich).

Subjekt:

als Subjekt wird jene Enität verstanden, über die eine Aussage getroffen wird. Eine Rechnung über den bestellten Medion-Rechner bei Saturn würde demzufolge als Subjekt saturn und medion bekommen, die Datei mit rechnung_saturn_medion beginnen.

Status:

der Bearbeitungzustand des Dokumentes, wenn es der stetigen Veränderung unterworfen ist. Bei Listen böte sich zum Beispiel voll oder unvoll an, um zu erklären, ob sie komplett sind. Bei Office-Dokumenten könnte auf final oder unfinal zurückgegriffen werden.

ID:

bei automatisch vergebenen Identifikationsnummern ist es mitunter hilfreich, die ID im Namen zu behalten, um die Datei der Originalversion zuordnen zu können. Zum Beispiel wäre es sinnvoll eine Statistik zu Plutarchs *De tranquillitate animi* aus dem TLG mit tlg0007-096 im Dateinamen zu kennzeichnen.

Datum:

Datumsangaben im Dateinamen sind insbesondere dann sinnvoll, wenn ein Dokument unveränderlich zu Archivierung abgelegt werden soll (protokoll_projektmeeting_replikation_2023-04-28.pdf) oder es zu sukzessiven Änderungen kommt, um die Versionierung übersichtlicher zu gestalten (replikation_stylesheed_2023_v01).

Versionsnummer:

Um die Bedeutung als Versionsnummer klar zu machen wird ein kleines v vorangestellt. Die Anzahl der führenden Nullen ist abhängig von der zu erwartbaren Menge.

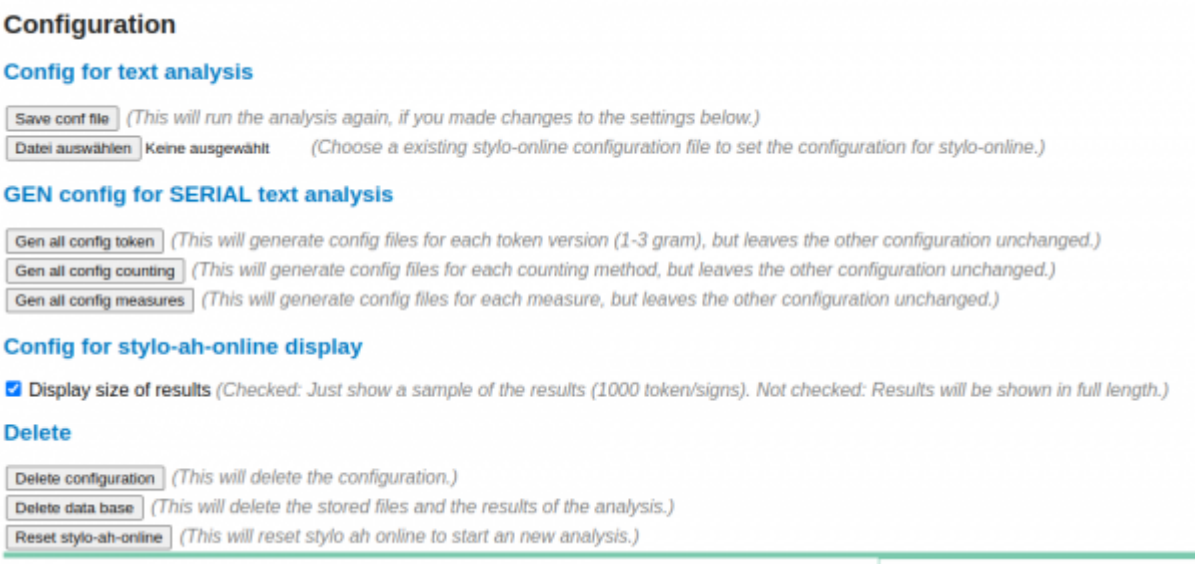
Autorname:

Auf die Benennung des (letzten) Autoren im Dateinamen sollte verzichtet werden außer eine Dateiversion wird parallel und unabhängig voneinander von mehreren Personen bearbeitet und es entstehen dadurch horizontale (anstatt vertikale) Dateiversionen, die später von Hand zusammengeführt werden müssen.

Dateiendung:

Die Dateiendung bezeichnet den Teil, der nach dem abschließenden Punkt folgt. Er verweist in aller Regel auf den Dateityp, also die Art und Weise, wie die Informationen in der Datei kodiert sind.

2.2 Arbeitsschritt: Configurations-Datei / Datenbank



Im Abschnitt „Configuration“ können alle Einstellungen vorgenommen werden, die sich auf die Config-Datei beziehen. stylo-ah-online erlaubt die Speicherung der Konfiguration und genauso ihren erneuten Aufruf durch Eingabe einer Config-Datei. Für das Erzeugen von Serien aus Experimenten erlaubt es stylo-ah-online mehrer Config-Dateien zu erstellen und diese als Serie wieder zu öffnen. Die Ergebnisse werden für jede so gespeicherte Einstellung unter einem extra Namen abgespeichert. Für umfangreichere Corpora ist es nötig die Darstellung von Zwischenergebnissen zu beschränken. Das kann an dieser Stelle mit dem Haken bei „Display size of results“ erzielt werden. stylo-ah-online verfügt über eine auto-save-Funktion, außerdem werden die eingegebenen Daten gespeichert, um diese bei einem erneuten Analysedurchlauf aus der Browser eigenen Datenbank zu holen. Die letzten Buttons des Abschnitts erlauben es, die Einstellungen und die Datenbanken oder beides zurücksetzen.

Input eines Config files

Wenn config files über die „Save config file“- oder „Gen all…“-Buttons erzeugt wurden, so können sie mittels dieses Buttons wieder geladen werden. Das Tool stellt die Konfiguration, dann so ein, wie diese im config file angegeben ist. Alte config files sind mit neueren kompatibel. Das Konfiguration wird stets auf das aktuelle Korpus angewendet.

Mehrer Config files

Werden mehrer config files geladen, dann werden diese nacheinander verarbeitet und die Ergebnisse werden, wie angegeben, abgespeichert. Mehrer Konfigurationen dieser Art werden auf dem aktuellen Korpus ausgeführt.

Serien von config files

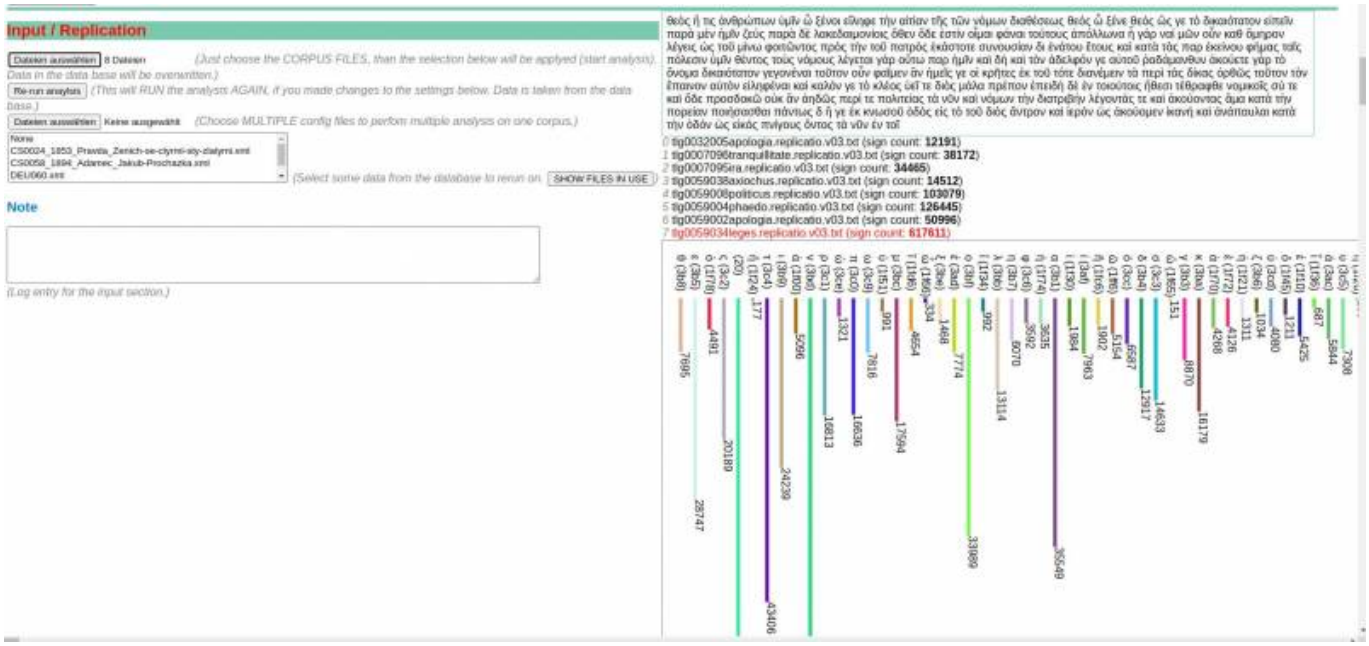
Wenn man die „Gen all…“-Buttons verwendet, dann werden alle Einstellungen die zur Zeit gemacht wurden, übernommen und lediglich die Einstellungen eines Arbeitsschritts werden variiert. Auf diese Weise kann man zu diesem Arbeitsschritt eine Serie von config files erstellen und somit eine Serie von Ergebnissen. Diese Funktion dient der systematischen Untersuchung von Arbeitsschritten in Bezug

auf ein gegebenes Corpus. Die „Gen all“-Funktion steht für die Arbeitsschritte Token-Bildung, Token-Zählung und Anwendung der Maße zur Verfügung.

Reset des Tools

Sowohl das Tool, als auch der Browser betreiben ein Caching. Caching bezeichnet die Speicherung von Einstellungen, Daten und Zuständen. Wenn der Button „Delet configuration“ benutzt wird, dann werden alle Einstellungen gelöscht und es werden auch keine default-Einstellungen vorgenommen. Im Anschluss können alle Einstellungen neu vorgenommen werden. Wenn der Button „Delete data base“ benutzt wird, so bleiben die Einstellungen erhalten, allerdings werden alle Eingabedaten und alle Zwischenergebnisse gelöscht. Die Datenbank der Eingabedaten und Zwischenergebnisse wird nicht automatisch gelöscht. Wenn man den Button „Reset stylo-ah-online“ auswählt, so werden die Einstellungen und die Datenbanken gelöscht. Will man die Seite und somit den Zustand (Programmversion) des Tools nicht aus dem Browser-Cach laden, sondern den neusten Code vom Server, dann kann man den Button „Refresh page/tool“ verwenden.

2.3 Arbeitsschritt: Input / Replication



Im Abschnitt „Input / Replication“ geht es um den Aufruf der Textdateien, die analysiert werden sollen. Diese Handlung löst die gesamte / erste Analyse aus und speichert Ergebnisse in der Datenbank. Mit „Re-run“ kann man eine veränderte Konfiguration auf die vorhandenen Daten anwenden. Sollten Serienexperimente vorbereitet worden sein, kann man die dazugehörigen Config-Files hier eingeben und so die Serienverarbeitung auslösen. Dazu werden die Textdaten aus der Datenbank verwendet. Der Arbeitsschritt der Dateieingabe muss der letzte Schritt sein, nachdem alle anderen Konfigurationen vorgenommen wurden.

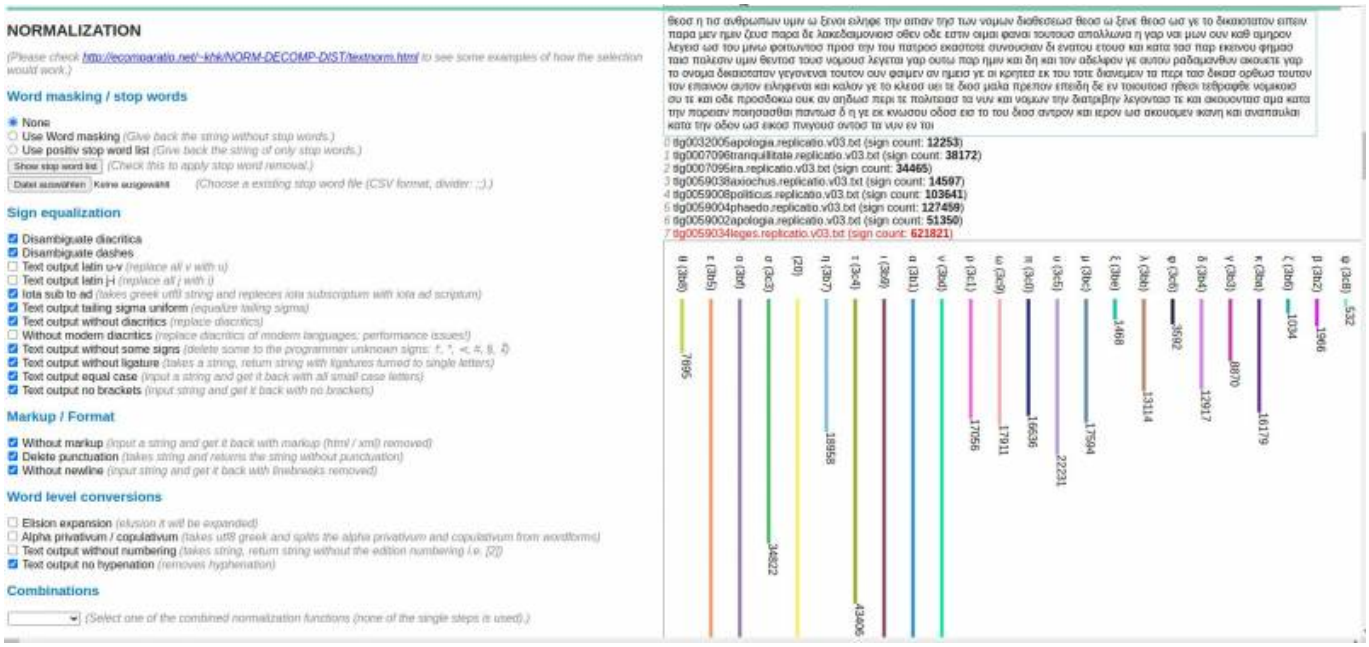
Im Vergleich zu **stylo** ist der Vorgang anders, wenn es um die Auswahl von Corpora geht: In **stylo** genügt es einen Ordner anzugeben, welcher den Ordner „corpus“ enthält. Für die Eingabe in stylo-ah-online muss die gesamte Liste der Texte ausgewählt werden, die das Corpus bilden.

Mit „Select some data from database“ können bereits geladene Dateien aus der Datenbank

ausgewählt und so kann ein neues Corpus zusammengestellt werden.

Zudem wird nun ein Histogramm der absoluten Häufigkeiten der Zeichen der Eingabe angezeigt.

2.4 Arbeitsschritt: Normalisierung (Normalization)



Auf der rechten Seite werden die Zwischenergebnisse des Normalisierungsschritts angezeigt. Man kann somit Einsicht in die Güte der Normalisierung nehmen und gegebenen falls Änderungen vornehmen. Unter dem Anzeigetext befindet sich die Liste der Eingabetexte, die als Menü funktioniert. Es wird außerdem ein Histogramm der absoluten Häufigkeiten der Zeichen in den normalisierten Texten angezeigt.

2.4.1 Word masking / stop words

Setzt man den Haken in diesem Abschnitt, dann werden die Wortformen auf der Stopp-Wortliste aus den Strings entfernt. Den Vorgang nennt man ebenfalls Maskierung von Wortformen. Man kann sich mit dem Button die aktuelle Stopp-Wortliste anzeigen lassen. Genauso ist es möglich, eine andere Stopp-Wortliste durch die Eingabe einer Datei, die jedes Stopp-Wort durch „;;;“ vom nächsten getrennt enthält, zu nutzen.

2.4.2 Sign equalization

In diesem Abschnitt der Einstellungen geht es um die Behandlung einzelner Zeichen und Zeichengruppen. Die ersten beiden Haken dienen der Vereinheitlichung von diakritischen Zeichen und Strichen. Dadurch wird die unterschiedliche Kodierung visuell gleicher Zeichen vereinheitlicht. Die weiteren Normalisierungen erklären sich (Ersetzung aller u und v durch v, Ersetzung aller i und j durch i, Vereinheitlichung von Schreibweisen Iota und Sigma, Auflösung von Ligaturen, die Lösung einiger Zeichen).

2.4.3 Markup / Format

Vornehmlich geht es um die Lösung zusätzlicher Formatierungsangaben und der Metadaten / Struktur. Sollte die Eingabe in wohlgeformtem XML geschehen, dann löscht der erste Haken dieses unter Verwendung eines XML-Parsers. Sollte dem nicht so sein, dann werden die XML-Tags mittels eines regulären Ausdrucks gelöscht. Hier können zudem Interpunktion und Zeilenumbrüche gelöscht werden.

2.4.4 Word level conversions

Unter die Veränderungen auf Wortformebene zählt alles, was die Wortform als logische, organisatorische Einheit berücksichtigt. Dies betrifft Wortformtrennungen, Nummerierungen, Elisionen und das Alpha privativum.

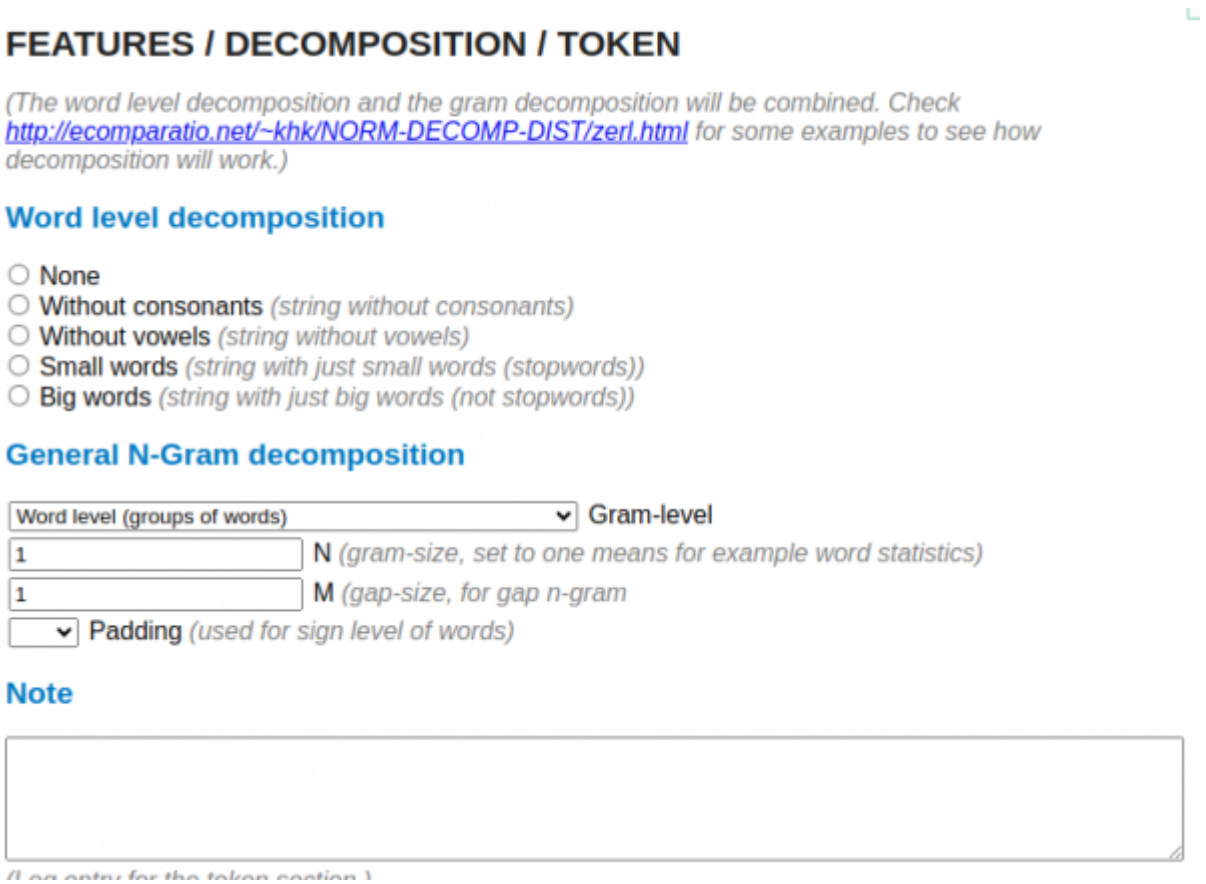
2.4.5 Combinations

Hier können Kombinationen von Normalisierungsschritten gewählt werden. Andere Einstellungen werden dann ignoriert.

2.4.6 Transliteration

Sollte es nötig sein Texte ihrem Zeichenbestand nach zu vereinheitlichen, dann kann diese durch die Transliteration (Griechisch / Latein) geschehen.

2.5 Arbeitsschritt: Zerlegung (features / decomposition / token)



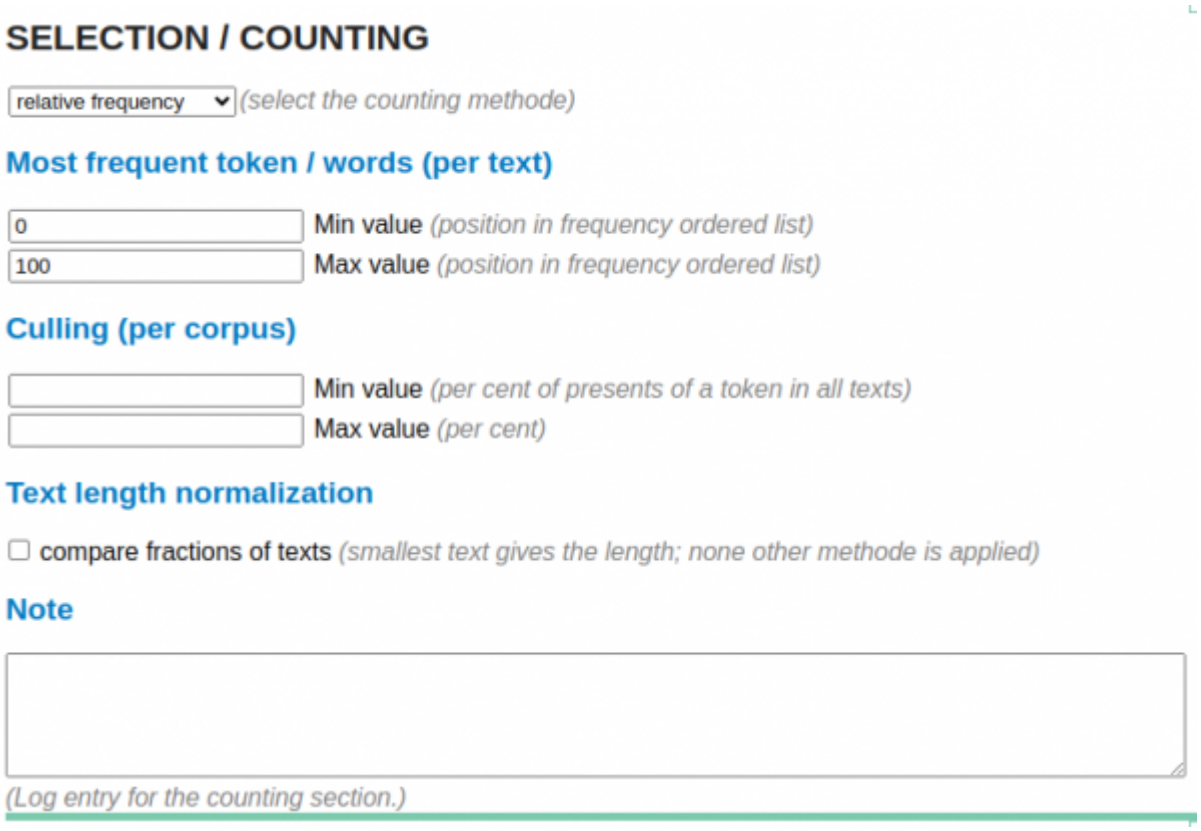
2.5.1 Word level decomposition

Die Einstellungen setzt die Vorstellung von Wortformen um, aus denen ein Text besteht. Jede Auswahl tut etwas mit der Wortform. Es kann nur ein Verfahren gewählt werden. Sollen lediglich Wortformen als Token behandelt werden, dann muss dies über die N-Gramm-Einstellung geschehen.

2.5.2 General N-Gram decomposition

Mittels dieser Einstellung kann der String in N-Gramme (Aufteilungen der Länge N, „N“ und „n“ sind Synonyme und stehen für eine gewählte natürliche Zahl) zerlegt werden. Dabei gibt die Auswahl im Pulldown-Menü die Zerlegungsebene an. Die zusätzlichen Zahlen müssen ausgefüllt werden, um das „N“ der Zerlegung oder der Lücken angeben zu können. Für Skip-Gramme spielt nicht nur das „N“ (Länge) der Teilung, sondern auch die Länge der Auslassung eine Rolle. Mit der Auswahl „Word level“ zerlegt man den String in „N“ lange Gruppen von Wortformen, die dann als Token ausgezählt werden. Mit der Einstellung „sign level of words“ teilt man die Zeichen der Wortformen auf. Durch die Auffüllung können Wortendungen und -anfänge kodiert werden. Die Wortübergänge werden nicht kodiert. Mit der Auswahl „signs of whole string“ wird die Aufteilung kontinuierlich auf dem String vorgenommen. Dabei werden Wortformübergänge (ab n = 3) berücksichtigt. „Gap-ngram“ ist die Skip-Gramm-Implementierung, hier muss man zusätzlich die Länge der Lücke angeben. Für lateinische und griechische Texte steht die Zerlegung in Silben zur Verfügung. Abschließend gibt es noch die Zerlegung der Wortformen in drei Abschnitte. Die erste dieser Einstellungen teilt zu gleichen Teilen auf, die zweite teilt jede Wortform so auf, dass alle Partitionen des Strings entstehen.

2.6 Arbeitsschritt: Zählung (Selection / Counting)



Im oberen Pulldown-Menü des Abschnitts kann man sich für die Zählweise entscheiden, die auf die Menge aller Token (Wortformen) angewendet werden soll. Die absolute Häufigkeit, ist die Anzahl des Auftretens eines Token. Die relative Häufigkeit ist die Auftretenszahl eines Token geteilt durch die Textlänge. Das schwächt den Einfluss der Textlänge ab. Allerdings überdeckt dies das Problem im Corpus sehr kurze Texte mit sehr langen Texten zu vergleichen. Die Kodierung des Auftretens oder Nichtauftretens in einem Text, kann mittels 0 und 1 geschehen. Dann wird lediglich die Existenz von Token untersucht. Abschließend stehen zwei Zählungen bereit, die eine Beziehung zum Gesamtkorpus herstellen. Die TF-IDF stellt den Bezug zwischen relativer Häufigkeit in einem Text eines Token zur Häufigkeit in Texten vertreten zu sein her. Der Quotient aus relativer Häufigkeit pro Text und relativer Häufigkeit im Corpus mindert die Funktion der relativen Häufigkeit ab und sollte Werte ergeben, die sich zwischen der absoluten Häufigkeit und der relativen Häufigkeit bewegen.

Um dem Umstand verschieden langer Texte noch auf eine andere Weise zu begegnen, existiert die Einstellung „Text length normalization“. Setzt man hier den Haken, dann wird jeder Text in Teile der Länge des kürzesten Textes im Corpus zerlegt. Will man diesen Vorgang rückgängig machen, so muss man die Texte erneut laden und den Haken zuvor entfernen. Bei umfangreicheren Corpora und sehr unterschiedlich bemessenen Textlängen ist Vorsicht angebracht. Vergleicht man ein Fragment weniger hundert Wortformen mit einem Buch, so wird das Buch in viele tausend Teile zerlegt. Die Funktion steht nur beim Laden eines Corpus zur Verfügung und kann nicht mit dem „Re-run“ Button ausgelöst werden.

2.6.1 Most frequent token / words (per text)

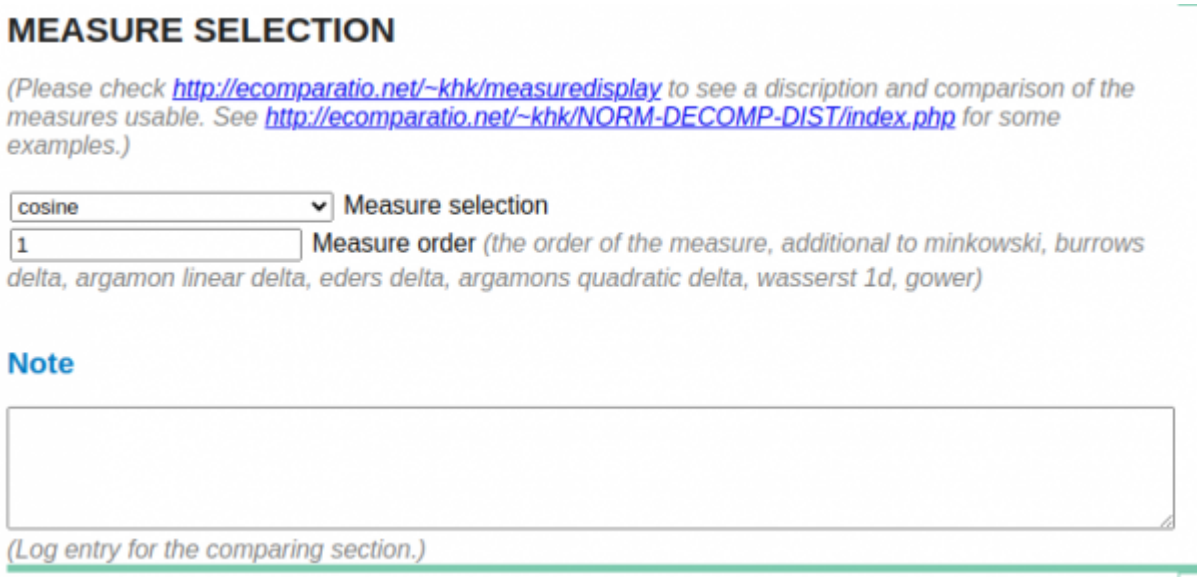
Mit den beiden Zahlenangaben kann der Frequenzrang angegeben werden, der im Profil Berücksichtigung finden soll. Sollen lediglich die Werte der 100 häufigsten Token Betrachtung finden,

dann muss man den niedrigsten Rang 0, als minimal zu berücksichtigenden Rang, und den Wert 100, als höchsten zu berücksichtigenden Rang, angeben.

2.6.2 Culling

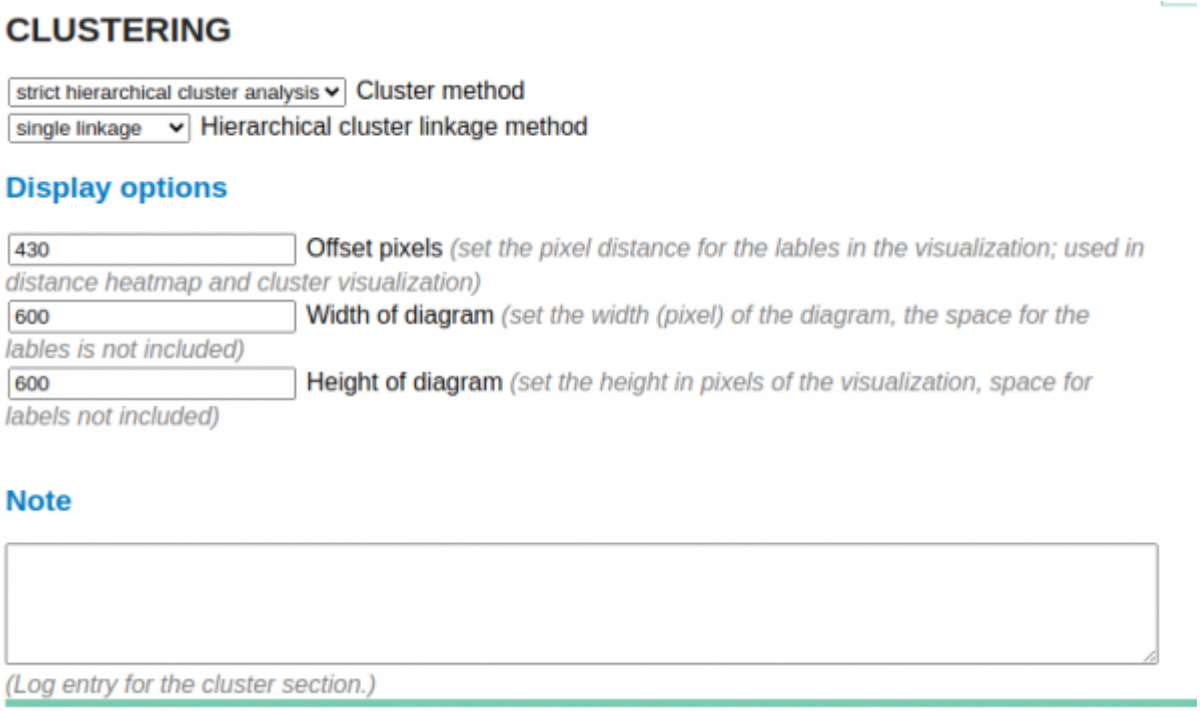
Mit den beiden Werten kann man angeben, in wie viel Prozent der Texte ein Token mindestens und maximal auftreten soll, um im Profil Berücksichtigung zu finden. Im Kontrast zu Most frequent words oder TF-IDF kann man dabei nicht absehen, wie die Profile beeinflusst werden. Es kann durchaus sein, dass damit null-besetzte Bereiche vermindert werden, allerdings kann auch das Gegenteil eintreten. Hier sollte man die Profile genau untersuchen.

2.7 Arbeitsschritt: Maßanwendung (Measure selection)



Aus dem Pulldown-Menü können verschiedene Distanzmaße ausgewählt werden, die zum Vergleich zwischen der Token-Profilen benutzt werden sollen. Die Anwendung der Maße auf die Profil-Vektoren ergibt die Distanzmatrix. Die Distanzmatrix ist Grundlage der Gruppierung der Texte untereinander. Manche Maße erfordern die Angabe eines Zahlenwerts. Was dieser bedeutet, ist von Maß zu Maß verschieden. Man sollte die Dokumentation der Maß, wie sie in stylo-ah-online verlinkt ist, konsultieren.

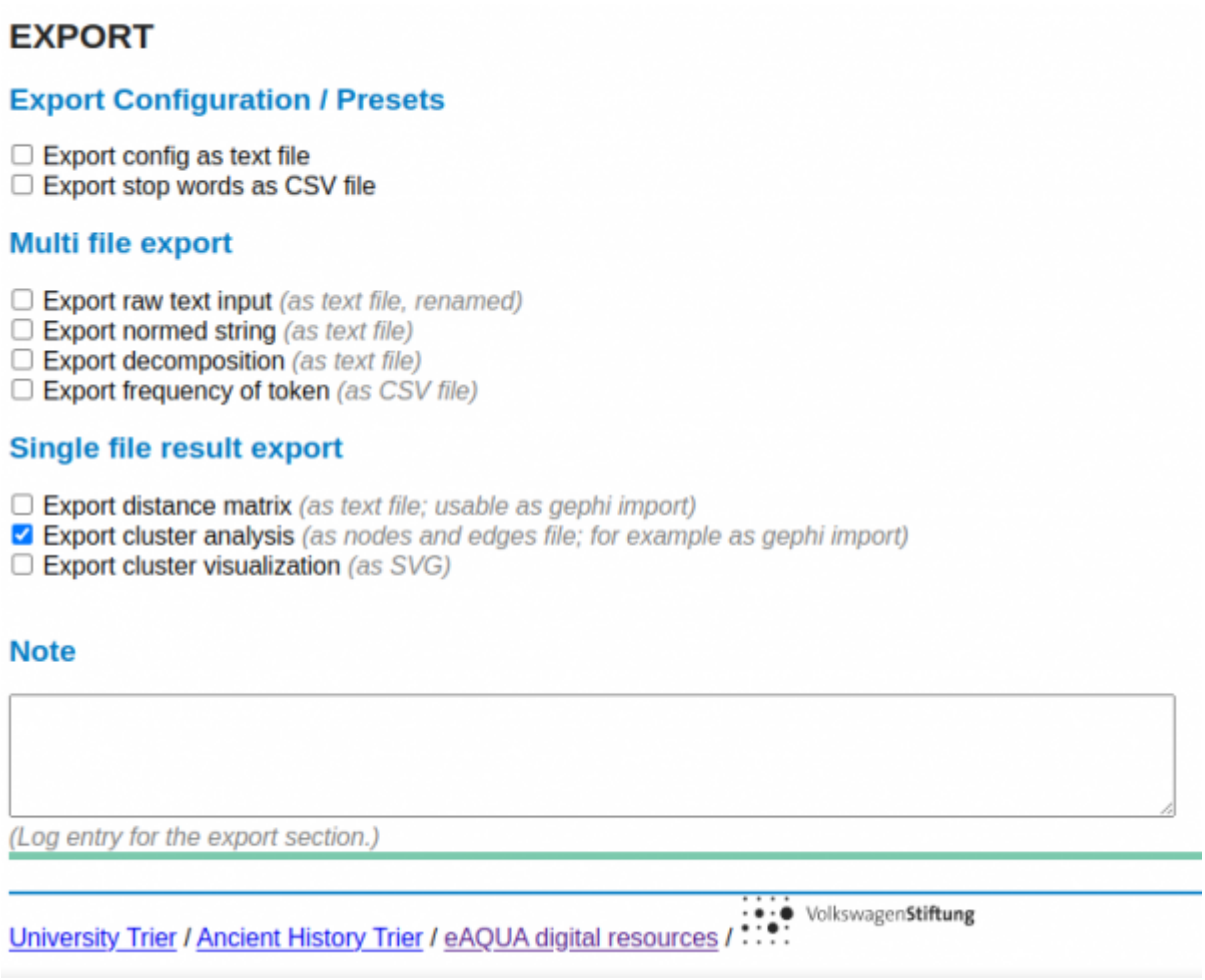
2.8 Arbeitsschritt: Gruppierung (Clustering)



Zur Zeit stehen noch nicht alle Cluster-Verfahren, die **stylo** verwendet zur Verfügung. **stylo** nutzt vornehmlich die Implementierungen Dritter, um die Funktionalität anzubieten, diesen Vorteil haben wir für die Implementierung von stylo-ah-online nicht genutzt. Zur Zeit steht die hierarchische Clusterung in zwei verschiedenen Darstellungen zur Verfügung. Ebenso die tSNE, von der, als optimale Einbettung, allerdings als Cluster-Methode für den gesamten Gegenstand der Textanalyse abzuraten ist. Auch steht eine alternative Variante des MDS (multi dimensional scaling, Anwendung einer radialen Funktion) zur Verfügung. Die Implementierung wird noch komplettiert.

Wichtig für die graphische Darstellung sind die Offset-Werte (Bereich für Beschriftungen) und die Größenangabe der Grafik. Die Grafiken können als SVG gespeichert werden und in Veröffentlichungen Einsatz finden. Die Werte beziehen sich auf alle erstellten Grafiken.

2.9 Arbeitsschritt: Export



Im Abschnitt „Export“ können, durch das Setzen von Haken, verschiedene Dateien exportiert werden. Diese werden bei jedem Programmdurchlauf geschrieben. Im Abschnitt wird nach drei Typen von Export unterschieden: Dem Export grundsätzlicher Dateien, wie der Config-Datei, dem Export von Zwischenergebnissen (die dann auch für jeden eingegebenen Text existieren) und dem Export von Ergebnissen (die für das ganze Corpus gelten). Die Dateien werden im angewählten Download Ordner gespeichert.

3. Erneute Berechnung und Serien

stylo-ah-online speichert das Corpus, welches aktuell bearbeitet wird. Es speichert auch Zwischenergebnisse. Werden Änderungen der Konfiguration vorgenommen, dann müssen nicht unbedingt alle Arbeitsschritte neu ausgeführt werden. Verwenden Sie den „Re-run“ Button, um ein neue Konfiguration auf das gespeicherte Corpus anzuwenden. Man profitiert damit von Vorberechnungen.

Sollen mehrere verschiedene Konfigurationen ausgeführt und die Ergebnisse exportiert werden, dann legen sie mehrere Config-Dateien an und öffnen diese als Datei-Liste. Jede Konfiguration wird dann auf die gespeicherten Daten des Corpus angewendet. Auch hier kann man von der Wiederverwendung von unveränderten Zwischenergebnissen profitieren.