

Annohub Benutzerhandbuch (v. 2.0)

Institut für angewandte Computerlinguistik
Goethe Universität Frankfurt

Frank Abromeit

28. September 2022

Inhaltsverzeichnis

1	Einleitung	2
2	Katalog	3
2.1	Katalogliste	5
2.2	Allgemeine Metadaten	8
2.3	Kommentarfunktion	10
2.4	Suchfunktionen	11
2.4.1	Kombinierte Sprach- und Annotationsmodellsuche	11
2.4.2	Tag Suche	11
2.4.3	URL Suche	12
2.4.4	OLiA Suche	12
2.4.5	Suche in allgemeinen Metadaten	13
2.4.6	Suche in Kommentaren	13
2.4.7	Suche nach Benutzern	14
3	Analyse Editor	14
3.1	Grundlegende Editfunktionen	16
3.1.1	Sprachbearbeitung	17
3.1.2	Modellbearbeitung	19
3.1.3	Abspeichern von Änderungen	22
3.1.4	Löschen von einzelnen Dateien einer Resource	22
3.1.5	Dateisample	22
3.2	Fortgeschrittene Editfunktionen	22
3.2.1	Gleichzeitiges Editieren mehrerer Ressourcen	22
4	Sprachdatenanalyse	23
4.1	Archive	24

4.2	Erneute Analyse	25
4.3	Warteliste	25
4.4	Fehlerprotokoll	26
5	Administration	27
5.1	Benutzerverwaltung	27
5.2	Sprachprofile	29
5.3	Backup	32
5.4	OLiA Manager	33
6	Datenbanken	37
6.1	Registrierungs Datenbank	37
6.2	Modell Datenbank	39
7	Installation und Konfiguration	40
7.1	Vorraussetzungen	40
7.2	Build	40
7.3	Initialisierung	40
7.4	Konfigurationsdatei (FIDConfig.xml)	41
7.5	Konfigurationsparameter	41
8	Kommandozeilen Interface	46
9	Fragen und Antworten	47

1 Einleitung

Annohub ist ein Tool für die Analyse von annotierten Textkorpora auf die in einem Textkorpus verwendeten Sprachen. Ausserdem erkennt das Tool Metainformationen wie benutzte Annotations-Schemata für die Annotation von Syntax und Morphologie und auch Ontologien, die für die Modellierung der Sprachdaten verwendet werden. Es werden zahlreiche Korpusformate unterstützt, darunter RDF (Resource Description Framework), CoNLL (Conference on Computational Natural Language Learning) und spezielle XML kodierte Sprachdaten. Das Tool läuft als Web-Applikation im Browser. Die Benutzeroberfläche enthält sämtliche Funktionen für die Analyse. Nebenbei speichert die Applikation alle Analyseergebnisse und stellt diese in einem Katalog für die Suche bereit. Die einzelnen Komponenten der Software werden in den folgenden Kapiteln beschrieben.

- NLP-Analyse-Tools (RDF, CoNLL, XML-Parser)
- NLP-Backend-Konfiguraton (Sprach- u. Annotationsdefinitionen)
- Benutzeroberfläche (Views & Funktionen)

- Benutzerverwaltung (Accounts, Rechte)
- Katalog (Suche, Export)
- Installation und Konfiguration (auch Backup)

2 Katalog

Die Benutzeroberfläche von Annohub ist auf mehrere Seiten für die jeweiligen Funktionen aufgeteilt. Jedes Benutzerprofil verfügt über einen vordefinierten Funktionsumfang. Im folgenden werden die vorhandenen Funktionen für das Administrator Profil vorgestellt. In der Hauptansicht ist der Katalog aller analysierten Sprachkorpora als Liste dargestellt. Ganz oben im Header wird die Anzahl der aktuell im Katalog enthaltenen Korpora, Lexika und Ontologien angezeigt. Diese Zahlen beziehen sich nur auf Datensätze, die bereits einer Revision unterzogen wurden und für den Export freigegeben wurden. Es können allerdings noch weitere Datensätze im Katalog vorhanden sein, die noch nicht freigegeben sind.¹

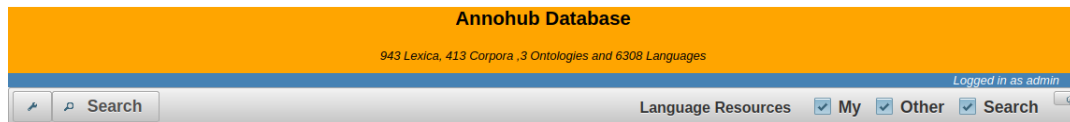


Abbildung 1: Werkzeugtoolbar

Die Werkzeugleiste enthält Reiter für die verschiedenen Funktionsansichten (Werkzeugsymbol) und Suchwerkzeuge.

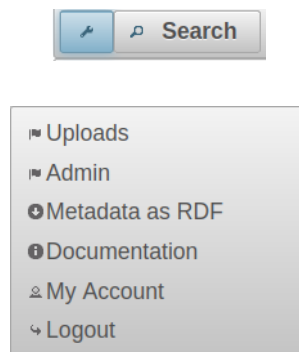


Abbildung 2: Werkzeug Menü

Funktion	Beschreibung
----------	--------------

¹Man erhält die unbestätigten Datensätze mit der Sortierfunktion der Spalte *Approved*

Uploads	Starten der Analyse
Admin	Administration und Konfigurationseinstellungen
Metadata as RDF	Erzeugt eine RDF-Datei mit den Metadaten aller <i>Bestätigten</i> Sprachressourcen aus dem Katalog
Documentation	Zeigt die Dokumentation
My Account	Nutzereinstellungen (wie Passwort, etc.) bearbeiten
Logout	Logout

Tabelle 1: Werkzeuge

Auf der rechten Seite befinden sich die Korpusfilter Checkboxes, mit folgender Funktion



- **My** filtert alle Ressourcen, die mir von einem Benutzer selbst analysiert worden sind. Ausserdem werden solche Sprachressourcen gezeigt, die zuvor mit der *Add* Funktion gemerkt wurden
- **Other** filtert alle Ressourcen, die von anderen Benutzern analysiert worden sind
- **Search** filtert alle Ressourcen, die in der Ergebnisliste der letzten Suche vorhanden sind

Die Kombination mehrerer Filter verhält sich dabei wie die Vereinigung. Der Update-Button (Phi) dient dazu neu verfügbare Analyseergebnisse im Katalog anzuzeigen. Das Neuladen der Seite über den Reload-Button des Web-Browsers kann hierfür nicht verwendet werden.

2.1 Katalogliste

Annohub Database

943 Lexica, 413 Corpora ,3 Ontologies and 6308 Languages

Logged in as admin

Search

Language Resources

☒ My

☒ Other

☒ Search

(1 of 146)

1

2

3

4

5

6

7

8

9

10

>>

<<

Resource	Uploader	Queued / Processed Date	Status	Comments	Metadata	Approved	Type	Online
https://www.clarin.si/repository/vmlu/bistream/hierarchy/11352/2431/PanabMint-LVana.tgz?sequence=31	ub	Fri Jun 03 16:10:58 CEST 2022	FINISHED		USER	yes	CORPUS	<div></div>
https://github.com/acoli-repo/acoli-dicts/raw/master/stable/panlex/panlex-20191001-csv-rdf/rdf/32/326/326.extracted.rdf.gz	ub	Fri Jun 03 15:30:36 CEST 2022	FINISHED		DATAFILE	yes	LEXICON	<div></div>
https://github.com/acoli-repo/acoli-dicts/raw/master/stable/panlex/panlex-20191001-csv-rdf/rdf/33/338/338.extracted.rdf.gz	ub	Fri Jun 03 15:30:03 CEST 2022	FINISHED		DATAFILE	yes	LEXICON	<div></div>
https://github.com/acoli-repo/acoli-dicts/raw/master/stable/panlex/panlex-20191001-csv-rdf/rdf/34/344/344.extracted.rdf.gz	ub	Fri Jun 03 15:29:55 CEST 2022	FINISHED		DATAFILE	yes	LEXICON	<div></div>
https://github.com/acoli-repo/acoli-dicts/raw/master/stable/panlex/panlex-20191001-csv-rdf/rdf/39/391/391.extracted.rdf.gz	ub	Fri Jun 03 15:29:16 CEST 2022	FINISHED		DATAFILE	yes	LEXICON	<div></div>

Abbildung 3: Katalogansicht für Admin-Benutzer

Jede Zeile im Katalog enthält Informationen zu einer Sprachresource.

Spalte	Beschreibung
Resource	URL unter der die Daten einer Sprachresource verfügbar sind
Uploader	Benutzer, der eine Sprachresource hochgeladen hat
Queued/Processed Date	Datum zu dem eine Sprachresource in die Warteschlange eingefügt wurde, bzw. Datum an dem die Analyse gestartet wurde
Status	Verarbeitungsstand einer Sprachresource: WAITING <i>in der Warteschlange</i> INPROGRESS <i>in der Verarbeitung</i> FINISHED <i>vollständig verarbeitet</i> Den SEARCH Status haben solche Sprachresources, die in der Ergebnisliste einer Suche sind.
Comments	Kommentare, die zu einer Sprachresource gemacht wurden

Metadata	Allgemeine Metadaten einer Sprachresource sind der Author, Title, etc. .Diese Spalte zeigt die Quelle dieser Daten (USER DATAFILE CLARIN LINGHUB). Diese können auch in einer Eingabemaske über die Funktion <i>About</i> im Kontextmenü einer Sprachresource manuell eingegeben werden.
Approved	Eine Sprachresource gilt als <i>freigegeben</i> , wenn mindestens eine Datei der Sprachresource den Status <i>Accepted</i> hat. Der <i>Accepted</i> Status muss über die Funktion <i>Accept</i> im Analyseeditor (erreichbar über das Kontextmenü einer Sprachresource) zugewiesen werden
Type	Typ einer Sprachresource (Corpus, Lexicon, Ontology, Wordnet, Unknown, Error)
Online	Die Applikation testet, ob eine Resource online verfügbar ist, indem überprüft wird, ob die URL im Resourcefeld verfügbar ist. Der Test wird zur Serverstartzeit, manuell über die Benutzeroberfläche (über Admin->Configuration->Check broken data links) oder automatisch in einem vorgegebenen Intervall ausgeführt (ebenfalls dort einstellbar). Die Daten, die eine URL bereitstellt, werden jedoch nicht überprüft.

Tabelle 2: Spalten der Katalogansicht

Kontextmenü einer Sprachresource im Katalog Im Kontextmenü einer Sprachresource stehen weitergehende Funktionen zur Verfügung.

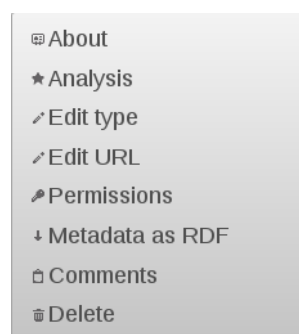


Abbildung 4: Kontextmenü Sprachresource

Funktion	Beschreibung
About	Zeigt die allgemeinen Metadaten wie Author, Titel, etc.. und erlaubt deren Änderung (diese Ansicht ist auch über Doppelklick verfügbar)
Analysis	Öffnet die Analyseeditor Ansicht
Edit Type	Der Typ einer Sprachresource wird automatisch während der Analyse ermittelt. Bei Fehlern kann der Typ hier editiert werden.
Edit URL	Ändern der Daten-URL. Eine URL muss aktiv sein. Die Daten, die eine URL bereitstellt werden nicht überprüft!
Metadata as RDF	Erzeugt eine RDF-Datei, die alle Metadaten einer Sprachresource enthält
Comments	Öffnet den Kommentareditor. Hiermit lassen sich gepostete Kommentare anschauen und neue erstellen
Delete	Löscht eine Sprachresource aus dem Katalog irreversibel . Ressourcen mit dem Status SEARCH können erst gelöscht werden, wenn der SEARCH Filter deaktiviert ist
Add	Eine Resource, die einem anderen Benutzer gehört merken. Diese erscheint dann immer wenn der Filter My gesetzt ist
Permissions	Es können Lese- und Schreibrechte für andere Benutzer vergeben werden

Tabelle 3: Kontextmenü Sprachresource

Ergebnis einer Löschoperation in Abhängigkeit vom Zustand (Status) einer Sprachresource

Resource Status	Besitzer	Delete Operation
<i>WAITING</i>	ja	Entfernt eine Resource aus der Warteschlange
<i>INPROGRESS</i>	ja	Das Abrechnen der Verarbeitung ist z.Z. nicht möglich
<i>FINISHED</i>	ja	Löscht eine Resource irreversibel aus dem Katalog!
<i>SEARCH</i>	ja	Kein Effekt - Zum Löschen Suchfilter deaktivieren!
<i>WAITING</i>	nein	Kein Effekt - Kann nicht Resource eines anderen Benutzers löschen!
<i>INPROGRESS</i>	nein	Kein Effekt - Kann nicht Resource eines anderen Benutzers löschen!
<i>FINISHED</i>	nein	Entfernt eine <i>gemerkte</i> Resource aus der Liste
<i>SEARCH</i>	nein	Kein Effekt - Zum Löschen Suchfilter deaktivieren!

Zugriff auf Sprachressourcen Für jede Sprachressource lassen sich die Zugriffsrechte individuell setzen. Dazu kann der Besitzer unter *Permissions* Leserechte und Editierrechte setzen. Standardmässig sind Leserechte für alle anderen Benutzer einstellen.

Resource Permissions

Other users

☒ Read

☐ Edit

CANCEL **SAVE**

Abbildung 5: Zugriffsteuerung

2.2 Allgemeine Metadaten

Resource Metadata

Title :	Chinese WordNet - Ontolex edition	UB-Title :	
Description:		Keywords:	
Creator:	Christian Chiarcos	Contributor:	
Contact email:		Identifier:	
Webpage:	https://github.com/acoli-repo/acoli-dicts	Type:	
Licence:		Format:	
Rights:		Publisher:	
Year:	2020	Source:	
Location:		Languages:	
Metadata source:	USER		

CANCEL **SAVE**

Abbildung 6: Allgemeine Metadaten

Attribut	RDF	JSON
Title	dc:title	title
Description	dct:description	description
Creator	dc:creator	creator
Contact Mail	metashare:email	contact
Webpage	-	webpage
Licence	dc:licence	-
Rights	dc:rights	licence
Year	-	year
Metadata source	rdfs:comment	-

UB-Title	annohub:ubTitle	-
Keywords	dc:subject	-
Contributor	dc:contributor	contributor
Identifier	dct:identifier	-
Type	-	type
Format	-	format
Publisher	dc:publisher	publisher
Source	dct:source	-
Languages	-	-

Tabelle 4: Serialisierung allgemeiner Metadaten

Einige Attribute werden zwar im Annohub-Interface angezeigt, sind aber für die Annohub Serialisierungen irrelevant. Z.B. werden nur die ermittelten Sprachinformationen der Analyse für die Serialisierungen verwendet und nicht die Information im Feld *Languages*. Diese sollten, falls vorhanden, für die Ergänzung oder Korrektur der Sprachanalyseergebnisse verwendet werden. Ebenso werden für die RDF-Serialisierung Typ und Formatinformationen, die aus der Analyse stammen, direkt verwendet. Die Informationen der Felder *Webpage* und *Year* werden aktuell nicht serialisiert. Die Implementierung in RDF sollte noch nachgetragen werden².

Die allgemeinen Metainformationen können aus unterschiedlichen Quellen stammen.

- **Linghub** Metadaten werden per SPARQL Query aus dem RDF Datendump von Linghub (linghub.org) extrahiert.
- **Clarín** Metadaten werden aus dem XML-Format zuerst in einer Postgres Datenbank gespeichert. Per SQL Query werden Metadaten in Annohub importiert.
- **JSON** Für einzelne Webseiten wie Spraakbanken³ konnten Metadaten als JSON Datei direkt von einer Webseite heruntergeladen werden. Spezifische JSON Parser extrahieren daraus die Metadaten für Annohub.
- **Datei** Einige Sprachressourcen im RDF Format enthalten auch allgemeine Metadaten neben den Sprachdaten. Diese können per SPARQL Query aus einer RDF Datei herausgeparst werden, wenn beim Start der Analyse eine entsprechende Checkbox (siehe Abschnitt 4) markiert ist. Da von vornherein nicht bekannt ist, mit welchen RDF Vokabularen (RDF Attributen) allgemeine Metadaten kodiert sind wird ein generisches Mapping mit Standard-RDF-Attributen verwendet. Als Erweiterung

²dc: <http://purl.org/dc/elements/1.1/>, dct: <http://purl.org/dc/terms/>, rdfs: <http://www.w3.org/2000/01/rdf-schema#>, metashare: <http://purl.org/ms-lod/MetaShare.ttl#>, annohub: <http://acoli.cs.uni-frankfurt.de/annohub#>

³<https://spraakbanken.gu.se/>

der aktuellen Implementierung könnte man im Annohub-Interface eine Vorschau der gefundenen Attribute zeigen und diese dann manuell (zu Titel, Author, ...) zuzuordnen.

- **Manuell** Die allgemeinen Metadaten können ediert werden. Werden Metadaten editiert, dann erhalten sie automatisch das Attribut *Metadata source: USER*.

2.3 Kommentarfunktion

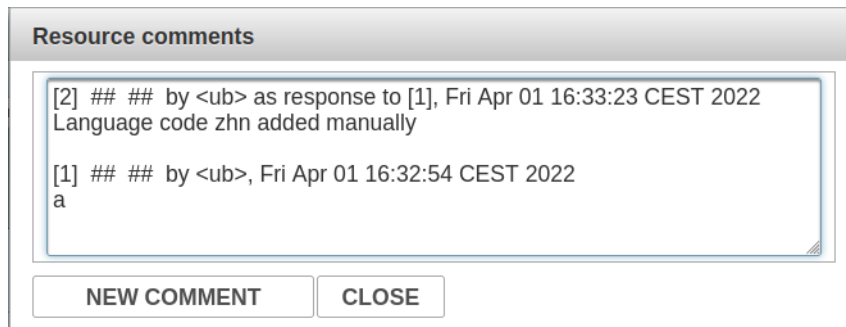


Abbildung 7: Kommentare

Die Kommentarfunktion erlaubt es registrierten Benutzern von Annohub Textbeiträge zu einzelnen Sprachressourcen zu erstellen. Diese sind nur für registrierte Benutzer sichtbar. Jeder Kommentar hat eine ID, der dazu dient um auf einen Post zu antworten. Um einen neuen Kommentar zu erstellen muss ein Kommentartitel und die Antwortreferenz angegeben werden. Jeder Textbeitrag ist auf maximal 200 Zeichen beschränkt.

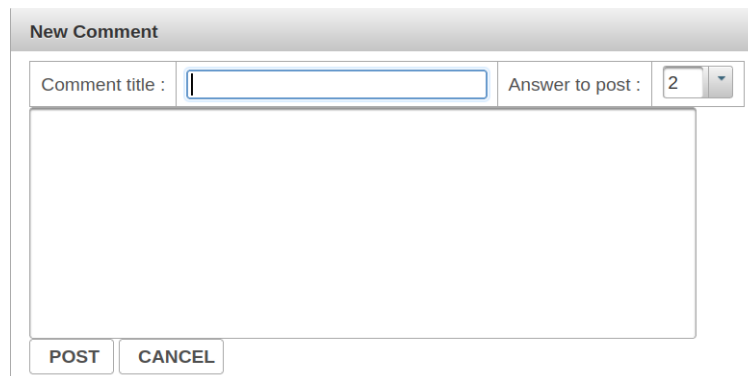


Abbildung 8: Neuen Kommentar erstellen

2.4 Suchfunktionen

Für die Suche nach Sprachressourcen im Katalog stehen verschiedene Suchwerkzeuge zur Verfügung.

2.4.1 Kombinierte Sprach- und Annotationsmodellssuche

The screenshot shows a web-based search interface titled "Query languages & tagsets". It contains several input fields and controls:

- Language (as ISO-639-3 code):** A text input field.
- Annotation tagset:** A dropdown menu currently showing "ALL".
- Resource type:** A dropdown menu currently showing "ALL".
- Resource name filter:** A text input field.
- Logic controls:** Radio buttons for "AND" and "OR" (selected), and a checkbox for "exclusive".
- Buttons:** "ADD" and "CLEAR" buttons are located next to the "Annotation tagset" dropdown.
- Checkbox:** A checkbox labeled "Ignore case" is checked.
- Footer:** A checkbox labeled "Add search results to workspace", and "GO" and "CANCEL" buttons.

Abbildung 9: Kombinierte Suche

1. **Language** Auswahl einer oder mehrerer Sprachen als ISO-639-3 Code
2. **Annotation model** Auswahl eines oder mehrerer Annotationsmodelle
3. **Resource type** Auswahl eines Ressourcetypes *Korpus*, *Lexikon*, *Ontologie*
4. **Ressourcenname** Infixsuche in Dateinamen

Durch Markieren von *Add search results permanently to view* werden die Suchergebnisse permanent in die Liste der eigenen Ressourcen (siehe **My**, Abb. 2) aufgenommen.

2.4.2 Tag Suche

Mit der Tag-Suche kann man nach Ressourcen im Katalog suchen, die eine spezielle Annotation (*Tag*) verwenden. Die Zahl hinter jedem Tag ist Anzahl der Resourcevorkommen.

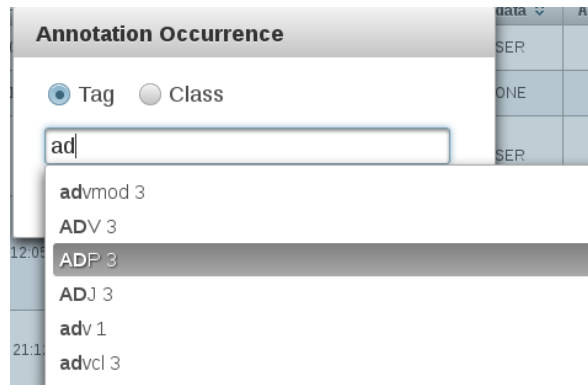


Abbildung 10: Tag Suche

2.4.3 URL Suche

Mit der URL-Suche kann man nach Ressourcen im Katalog suchen, die eine URL enthalten, die eine Referenz für eine Ontologiekategorie ist mit der eine Annotation definiert wurde. Die Zahl hinter jeder URL ist die Anzahl der Resourcevorkommen.

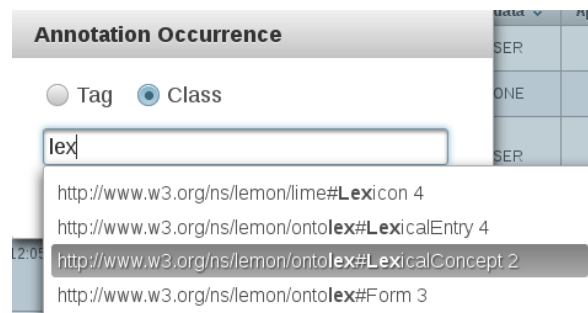


Abbildung 11: URL Suche

2.4.4 OLiA Suche

Mit diesem Tool lassen sich linguistische Annotationen unabhängig von einem Annotationsmodell in Sprachressourcen suchen. Dazu verwendet die Suche die OLiA Ontologieklassen, in denen universelle Wort-Annotationen für Morphologie und Syntax definiert sind. Die OLiA Ontologieklassen sind in einer Hierarchie geordnet. An der Spitze der Hierarchie befinden sich allgemeine Konzepte wie <http://purl.org/olia/olia-top.owl#MorphosyntacticCategory>. Davon abgeleitete Klassen sind spezifischer, z.B. <http://purl.org/olia/olia.owl#Verb>. Die OLiA Suche sucht für Annotationen in Sprachressourcen Klassen in der OLiA Klassenhierarchie, die diesen am besten entsprechen, also in der Klassenhierarchie möglichst weit unten stehen. So werden z.B.

für die Klasse `http://purl.org/olia/olia.owl#Adjective` alle Sprachressourcen gefunden, die Annotationen für 'Adjektive' enthalten. Der Text in der Textbox in Abb. 12 zeigt die Beschreibung einer OLiA Klassendefinition. Die Zahl hinter einer URL bezeichnet die Anzahl der Sprachressourcen im Katalog, für die eine bestimmte OLiA Klasse am spezifischsten ist.

Abbildung 12: OLiA Suche

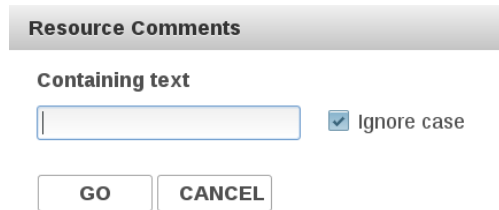
2.4.5 Suche in allgemeinen Metadaten

Damit lassen sich Sprachressourcen nach Author, Titel, etc. (siehe Abschnitt 2.2) finden.

Abbildung 13: Metadaten

2.4.6 Suche in Kommentaren

Sucht in den Kommentaren aller Sprachressourcen im Katalog.



Resource Comments

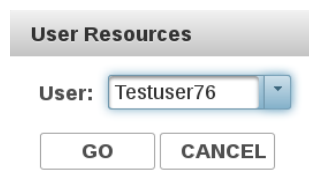
Containing text

☒ Ignore case

Abbildung 14: Kommentare

2.4.7 Suche nach Benutzern

Das ist eigentlich keine Suche. Vielmehr lassen sich damit alle Ressourcen eines Benutzers anzeigen.



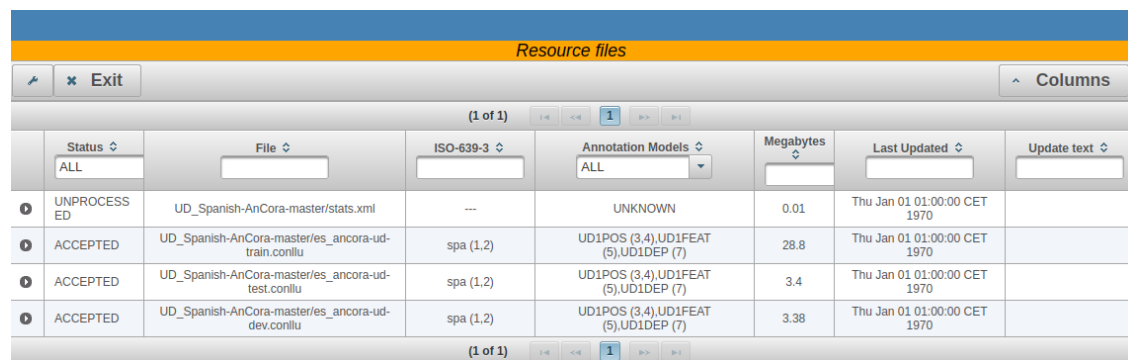
User Resources

User:

Abbildung 15: Benutzer

3 Analyse Editor

Da die Sprach- und Annotationserkennung nicht fehlerfrei ist, sollten alle Analyseergebnisse vor der Veröffentlichung kontrolliert und ggf. korrigiert werden. Die Tabelle im Editor enthält die Analysedetails für jede analysierte Datei einer Sprachresource. Im folgenden werden die Informationen in den einzelnen Spalten erläutert.



Resource files						
Status	File	ISO-639-3	Annotation Models	Megabytes	Last Updated	Update text
UNPROCESSED	UD_Spanish-AnCora-master/stats.xml	---	UNKNOWN	0.01	Thu Jan 01 01:00:00 CET 1970	
ACCEPTED	UD_Spanish-AnCora-master/es_ancora-ud-train.conllu	spa (1,2)	UD1POS (3,4),UD1FEAT (5),UD1DEP (7)	28.8	Thu Jan 01 01:00:00 CET 1970	
ACCEPTED	UD_Spanish-AnCora-master/es_ancora-ud-test.conllu	spa (1,2)	UD1POS (3,4),UD1FEAT (5),UD1DEP (7)	3.4	Thu Jan 01 01:00:00 CET 1970	
ACCEPTED	UD_Spanish-AnCora-master/es_ancora-ud-dev.conllu	spa (1,2)	UD1POS (3,4),UD1FEAT (5),UD1DEP (7)	3.38	Thu Jan 01 01:00:00 CET 1970	

Abbildung 16: Analyse Editor

Spalte	Beschreibung
Status	<i>Analysestatus</i> Processed Ergebnisse gefunden Check Sehr wenige Ergebnisse gefunden (mögl. Fehler) Edited Ergebnisse wurden editiert Accepted Alle Ergebnisse wurden für gut befunden Disabled Ergebnisse als unbrauchbar markiert Excluded Ein Fehler während der Analyse ist aufgetreten
Metadata URL	Ort von dem allgemeine Resouremetadaten stammen
Download URL	Daten URL (identisch mit der Resource URL in der Katalogansicht)
File	Relativer Dateipfad einer Datei in einer Sprachresource
Process Format	Internes Format in dem Daten verarbeitet werden (RDF CONLL)
ISO-639-3	Sprachcodes der gefundenen Sprachen (— für keine)
Annotation Models	Gefundene Annotationsmodelle (Nummern bezeichnen CoNLL Spalten)
Vocabulary	Gefundene RDF-Vokabulare (nur für Lexika)
Metadata Source	Quelle der allgemeinen Metadaten wie Beschreibung, Author, etc. (LINGHUB CLARIN USER)
Metadata State	Qualität der Metadaten : Complete : title, description, creator, year, license, email, webpage; Sufficient title, description, creator, year; Incomplete weniger als <i>Sufficient</i> vorhanden; Empty keine Metadaten vorhanden
Megabytes	Dateigrösse In MB
Type	Automatisch bestimmter Typ einer Sprachresource (Corpus, Lexicon, Ontology, Wordnet, Unknown)
Comment	Kommentar
Processed	Verarbeitungsdatum
Processing Time	Verarbeitungsdauer (z.B. PT30.72S heisst 32.72s)
Accepted	Datum an dem eine Datei als <i>Accepted</i> markiert wurde

Last Updated	Datum des letzten Updates der Analyseergebnisse. Das Sortieren von Ressourcen nach diesem Datum hilft dabei die Änderungen nach der Neuverarbeitung einer Resource oder einem OLiA-Modellupdate zu finden. Das Datum <i>1.1.1970</i> ist ein Dummy-Wert für <i>bisher kein Update durchgeführt</i> .
Update text	Aktion einer Updateaktion. Mögliche Werte sind <i>added</i> , wenn ein neues Modell oder eine Sprache entdeckt wurde oder <i>changed</i> falls ein bisheriges Ergebnis (Modell, Sprache) bei einer Update Operation verändert wurde.

Tabelle 5: Analysespalten

Mit dem *Columns* Knopf in der Werkzeugleiste können einzelne Spalten ein- und ausgeblendet werden.⁴ Nach der erfolgreichen Revision der Ergebnisse sollte zumindest einer Datei der Status **Accepted** zugordnet werden. Das hat zur Folge daß

- eine Resource automatisch im Katalog für alle (auch Gastnutzer) suchbar wird
- eine Resource in den Annohubexport (RDF,JSON) aufgenommen wird

3.1 Grundlegende Editfunktionen

Über das Kontextmenü einer Datei in der Dateiliste und weiter über das Stiftsymbol (siehe Abb. 17) können die Detailergebnisse für Sprach- und Modellergebnisse bearbeitet werden.

⁴Die Spaltenauswahl wird automatisch in der *gui.properties* Datei auf dem Server gesichert. Diese wird bei Serverneustart zurückgesetzt.

Edit Resource

Metadata URL	<input type="text" value="http://fid/metadata/tbc"/>	
Download URL	<input type="text" value="https://github.com/UniversalDependencies/UD_Indonesian/archive/master.zip"/>	
Languages	ind (1,2)	
Models	UD1POS (3),UD1FEAT (5),UD1DEP (7)	
Comments	<input type="text"/>	

CANCEL
SAVE
SAVE To ALL

Abbildung 17: Edit Resource

3.1.1 Sprachbearbeitung

Mit dem Stiftsymbol in der Zeile *Languages* im Dialog (siehe Abb. 17) wird der Spracheditor aufgerufen.

Edit RDF Languages

ISO
Add

(1 of 18)									
ISO639-3	Language	Detected by	From	Property	Probability	Count	Selected	Date	Update text
nld	Dutch	AUTO	LANGTAG	http://www.w3.org/ns/lemon/ontolex#writtenRep	1.0	2	true	2022-04-13	added
nld	Dutch	AUTO	LANGPROP	http://purl.org/dc/elements/1.1/language	1.0	1	true	2022-04-13	added
tur	Turkish	AUTO	LANGTAG	http://www.w3.org/ns/lemon/ontolex#writtenRep	1.0	127	true	2022-04-13	added
tur	Turkish	AUTO	LANGPROP	http://purl.org/dc/elements/1.1/language	1.0	1	true	2022-04-13	added
heb	Hebrew	AUTO	LANGTAG	http://www.w3.org/ns/lemon/ontolex#writtenRep	1.0	4	true	2022-04-13	added

(1 of 18)

Abbildung 18: Spracheditor RDF

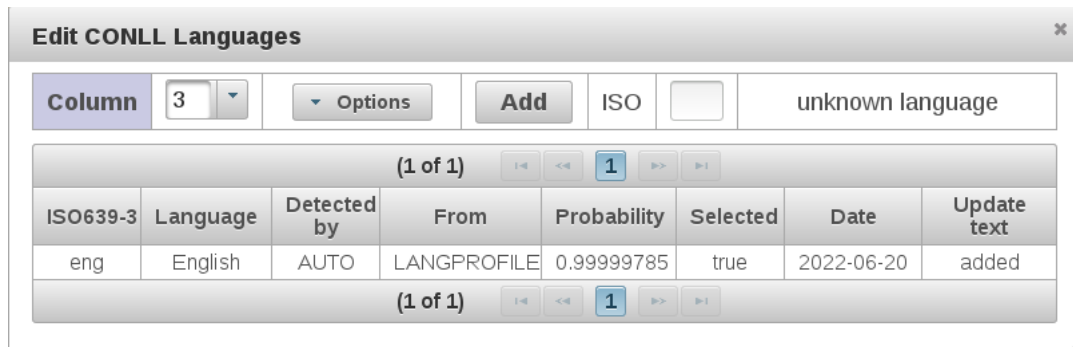


Abbildung 19: Spracheditor CoNLL

Mit dem Spracheditor lassen sich die automatisch erkannten Textsprachen bearbeiten und neue Sprachen (via Add) hinzufügen. Im Kontextmenü jeder gelisteten Sprache kann diese selektiert/deselektiert oder gelöscht werden.

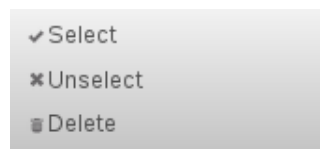


Abbildung 20: Spracheditor Kontextmenü

Im Folgenden findet sich eine Beschreibung des Inhalts der einzelnen Spalten.

Spalte	Beschreibung
ISO-639-3	ISO-Code einer gefundenen Sprache
Sprache	Sprache aus der ISO Codetabelle
Detected By	AUTO MANUAL automatisch oder manuell
From	<p>LangTag RDF Sprachtag (z.B. "Beispiel"@de, oder <code><rdfs:comment xml:lang=»en»edu</rdfs:comment></code>;</p> <p>LangProp Es gibt spezielle RDF Attribute, die für Sprachinformationen verwendet werden (Beispiel <code>http://purl.org/dc/terms/language</code>);</p> <p>LangProfile Aus einem Textsample wird mit einer n-gram Spracherkennung die Textsprache bestimmt.</p>
Probability	Die Wahrscheinlichkeit ist 1, falls eine Sprache manuell eingetragen wurde oder die Sprachinformation aus LangTag oder LangProp stammt. Für LangProfile stammt die Wahrscheinlichkeit aus dem Tool https://github.com/optimaize/language-detector

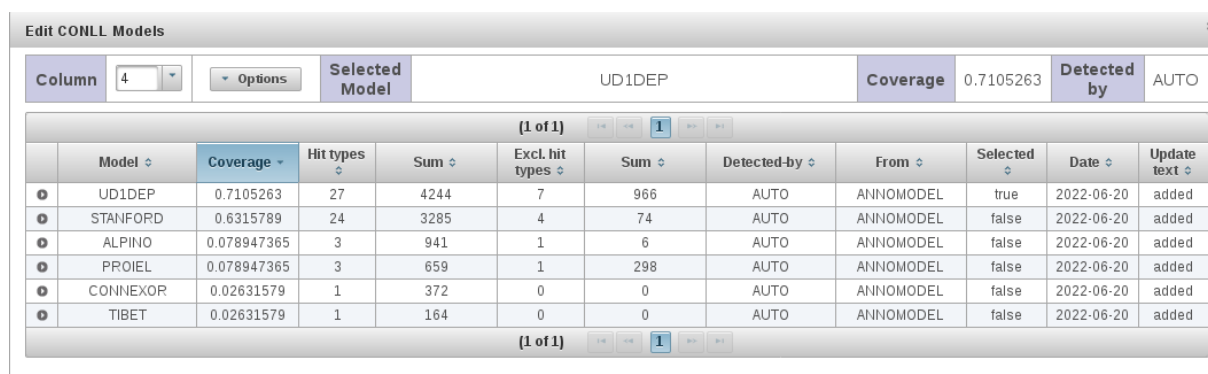
Count	Nur für RDF Ressourcen - Anzahl der Vorkommen
Selected	Eine Sprache kann deselektiert werden. Damit erscheint sie nicht mehr im Ergebnis. Alternativ lässt sich eine Sprache löschen.
Date	Datum der letzten Änderung
Update text	Aktion bei der letzten Änderung

Tabelle 6: Spracheditor

Weitere Informationen finden sich auch in [AFG20].

3.1.2 Modellbearbeitung

Mit dem Stiftsymbol in der Zeile *Models* im Dialog (siehe Abb. 9) wird der Modelleditor aufgerufen. Mit dem Modelleditor lässt sich die Auswahl der automatisch selektierten Annotationsmodelle ändern.



Column	4	Options	Selected Model	UD1DEP	Coverage	0.7105263	Detected by	AUTO
[1 of 1]								
Model	Coverage	Hit types	Sum	Excl. hit types	Sum	Detected-by	From	Selected
UD1DEP	0.7105263	27	4244	7	966	AUTO	ANNOMODEL	true
STANFORD	0.6315789	24	3285	4	74	AUTO	ANNOMODEL	false
ALPINO	0.078947365	3	941	1	6	AUTO	ANNOMODEL	false
PROIEL	0.078947365	3	659	1	298	AUTO	ANNOMODEL	false
CONNEXOR	0.02631579	1	372	0	0	AUTO	ANNOMODEL	false
TIBET	0.02631579	1	164	0	0	AUTO	ANNOMODEL	false
[1 of 1]								

Abbildung 21: Modellanalyse CoNLL

In der Kopfzeile des Modelldialogs (nur für CoNLL Dateien) lässt sich die Spalte einer CoNLL-Datei auswählen.⁵ Rechts daneben wird das aktuell zugeordnete Annotationsmodell für diese Spalte zusammen mit den Parametern *Detected By* und *Coverage* (Erklärung in Tabelle 7) gezeigt. Unter dem Reiter *Options* sind zusätzliche Funktionen zum Löschen und Hinzufügen⁶ von Spalten zu finden. Damit ist es möglich Spalten in einer

⁵Mehr Informationen zur Struktur von CoNLL Sprachdaten sind in [AFG20] zu finden

⁶Z.Z. nicht implementiert

CoNLL Datei, die irrtümlich als Modellspalte erkannt wurden zu löschen.⁷ Jede Zeile in der Modellanalyse entspricht einem Annotationsmodell. Die Spalten der Modellanalyse werden im Folgenden erklärt.

Spalte	Beschreibung
Zeilenexpansion	Zeigt die gefundenen Annotationen (Tags/Klassen) in einem Annotationsmodell
Model	ID des erkannten Annotationsmodells
Property	RDF Attribut mit dem eine Annotation kodiert ist (nur für RDF)
Coverage	Prozentualer Anteil der Annotationen in einer Datei, die einem Modell zugeordnet werden konnten
Hit types	Anzahl der verschiedenen zuordbaren Annotationen
Sum	Summe der zuordbaren Annotationen
Exclusive Hit types	Wie <i>hit types</i> aber nur Annotationen, die exklusiv nur diesem Modell zuordbar waren
Sum	Summe der zuordbaren Annotationen (nur exklusive)
Detected-by	AUTO MANUAL automatisch oder manuell
From	ANNOMODEL SELECTION
Selected	true false
Date	Datum der letzten Änderung
Update text	Aktion der letzten Änderung (added changed)

Tabelle 7: Modellanalyse

Jedes Modell kann über das Kontextmenü selektiert/deselektiert oder gelöscht werden.

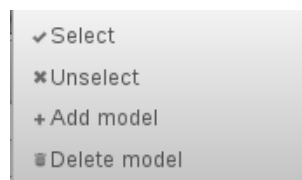


Abbildung 22: Kontextmenü

⁷Das kann z.B. passieren wenn die Textsprache in der Lemma-Spalte nicht erkannt werden konnte. Zur Korrektur muss zuerst die Modellspalte gelöscht werden. Danach kann die Spalte einer Sprache (im Spracheneditor) neu zugeordnet werden. Gelegentlich tritt auch der umgekehrte Fall auf, dass eine Spalte irrtümlich als Textspalte erkannt wurde. Genauso muss dann erst die Spalte im Spracheneditor gelöscht werden bevor der Spalte ein Modell zugeordnet werden kann.

Funktion	Beschreibung
Select	Wählt ein Annotationsmodell aus
Unselect	Deselektion. Alternativ kann ein Modell gelöscht werden.
Add Model	nicht implementiert
Delete Model	Modell löschen
Unselect All	Alle Modelle deselektieren (nur für RDF)

Tabelle 8: Kontextmenü

Die Expansion einer Zeile mit dem Expansionsreiter in der ersten Spalte zeigt die einzelnen zuordbaren Annotationen in einer Datei zu einem Annotationsmodell.

Edit RDF Models

(1 of 1)												
	Model	Property	Coverage	Hit types	Sum	Excl. hit types	Sum	Detected-by	From	Selected	Date	Update text
<input checked="" type="radio"/>	ONTOLEX	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	1.0	4	209848	4	209848	AUTO	ANNOMODEL	true	2022-04-13	added

(1 of 1)		
Found Tag/Class	Matching Tag/Class	Match count
http://www.w3.org/ns/lemon/lime#Lexicon	http://www.w3.org/ns/lemon/lime#Lexicon	44
http://www.w3.org/ns/lemon/ontolex#Form	http://www.w3.org/ns/lemon/ontolex#Form	5037
http://www.w3.org/ns/lemon/ontolex#LexicalConcept	http://www.w3.org/ns/lemon/ontolex#LexicalConcept	5037
http://www.w3.org/ns/lemon/ontolex#LexicalEntry	http://www.w3.org/ns/lemon/ontolex#LexicalEntry	199730

(1 of 1)		
----------	--	--

Abbildung 23: Zeilenexpansion Modellanalyse RDF

In der ersten Spalte der Expansion (*Found Tag/Class*) steht ein in der Datei gefundenes Tag (String) oder eine Ontologiekategorie (URL). In der mittleren Spalte (*Matching Tag/Class*) steht die Zuordnung des Eintrags in der ersten Spalte zu einem Tag oder einer Klasse in einem OLiA Annotationsmodell. In der letzten Spalte (*Match Count*) steht die Zahl der Vorkommen in der Datei des Werts in erster Spalte.

3.1.3 Abspeichern von Änderungen

Alle gemachten Änderungen (Sprachen und Modelle) werden über **SAVE** (siehe Abb. 17) in die Datenbank übernommen. Um das Editieren zu beschleunigen, wenn z.B. die gleichen Änderungen für mehrere Dateien vorgenommen werden sollen⁸ so ist dies über **SAVE to ALL** möglich (siehe Abb. 17). Damit werden die Modell- und Spracheinstellungen der aktuell ausgewählten Datei in allen Dateien einer Resource übernommen, die identische Einstellungen wie die editierte Datei vor deren Änderung haben. Dabei kann zusätzlich ausgewählt werden, ob nur Sprach-/Modell- oder Sprach- und Modelländerungen übertragen werden sollen.

3.1.4 Löschen von einzelnen Dateien einer Resource

Das Löschen einzelner Dateien kann sinnvoll sein, wenn diese keine Sprach- oder Annotationsinformationen enthalten, oder viele Dateien einer Resource identische Analyseergebnisse enthalten. Wenn die letzte Datei einer Sprachresource gelöscht wird, dann wird die Resource automatisch aus dem Katalog **irreversibel** gelöscht. Im allgemeinen ist es sinnvoll die Anzahl an Dateien in der Datenbank zu minimieren damit Updateoperationen, die auf allen Sprachresources ablaufen, z.B. nach einem OLiA-Modellupdate, schneller ablaufen.

3.1.5 Dateisample

In der *Sample* Ansicht wird ein Ausschnitt einer Sprachresource gezeigt. Diese Funktion ist nur für solche Ressourcen verfügbar, die zumindest gefundene Sprachinformationen haben. Es werden für RDF, CoNLL und XML Dateien jeweils 100 Zeilen angezeigt. Bei XML Dateien wird neben dem XML Code zusätzlich der automatisch erzeugte CoNLL Code gezeigt.

3.2 Fortgeschrittene Editfunktionen

3.2.1 Gleichzeitiges Editieren mehrerer Ressourcen

Das Bearbeiten der Ergebnisse von Ressourcen kann mühsam sein, wenn viele Dateien bearbeitet werden müssen. Um das Editieren zu erleichtern wurde zusätzliche Funktionen implementiert.

⁸z.B. bei identischen Fehlern

Funktion	Beschreibung
<i>Mark Accepted</i>	Dateien mit dem Zustand <i>Accepted</i> markieren
<i>Mark Disabled</i>	Dateien mit dem Zustand <i>Disabled</i> markieren
<i>Mark Processed</i>	Dateien mit dem Zustand <i>Processed</i> markieren. (nur für Dateien mit Zustand <i>Disabled</i>)
<i>Delete</i>	Dateien löschen



Abbildung 24: Multi-Edit

Im Dialog *Edit multiple files* wird zuerst die gewünschte Operation ausgewählt. Die URL im Textfeld in der Mitte kann bearbeitet werden, um die Dateien, die bearbeitet werden sollen auszuwählen. Bleibt die URL unbearbeitet dann wird nur die aktuell selektierte Datei verändert. Durch Verkürzen des URL-Strings lassen sich mehr Dateien auswählen.⁹ Nach *OK* kann im nächsten Schritt die Zielfmenge für die ausgewählte Operation überprüft werden und mit *EXECUTE* die ausgewählte Editoperation darauf ausgeführt werden.

4 Sprachdatenanalyse

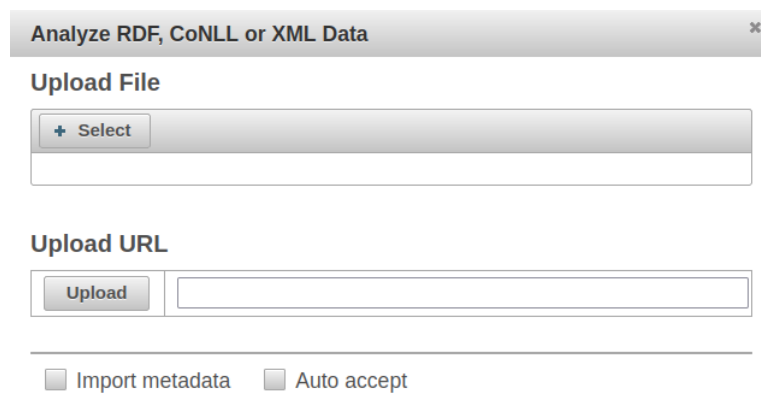


Abbildung 25: Bereitstellen von Sprachressourcen für die Analyse

Die Sprachdaten für die Analyse können als lokale Datei oder mittels URL zur Verfügung gestellt werden. Unterstützte Dateiformate sind alle RDF (rdf,owl,nt,n3, etc.), CoNLL (conll, conllu, conllx) und XML Dateiformate. Ebenso können Archive (zip,rar,tar,gz,

⁹Die Syntax für reguläre Ausdrücke kann verwendet werden. Der * Operator wird automatisch angefügt

etc.) verarbeitet werden. Für Archive können weitere Einstellungen für die Zahl der zu bearbeitenden Dateien eingestellt werden (siehe Abschnitt 4.1). Ein Archiv repräsentiert so wie Einzeldateien eine Sprachresource und wird im Katalog entsprechend dargestellt. Das kann genutzt werden um mehrere Einzeldateien in einer Sprachresource zusammenzufassen. Die Verarbeitung von URL-Listen funktioniert mit einer Textdatei vom Typ (tsv). In der Datei stehen URLs jeweils in einer Zeile. Zusätzliche Informationen in weiteren Spalten sind nicht erforderlich. Am unteren Ende des *Upload* Dialogs befinden sich zwei Checkboxes. Mit *Import metadata* können allgemeine Metadaten wie Author, Titel, etc., die zusammen mit den Sprachdaten in der gleichen Datei enthalten sind geparkt werden. Dazu müssen diese als RDF-Information mit entsprechenden RDF-Vokabularen notiert sein (Dublin Core, DCMI (dublin core terms)). Mit *Auto accept* wird den analysierten Sprachdaten automatisch der Zustand *Bestätigt* zugewiesen. Das ist sinnvoll, wenn z.B. URL Listen verarbeitet werden, von denen bekannt ist, dass die enthaltenen Daten valide sind, und nicht überprüft werden müssen.

4.1 Archive

Bei der Verarbeitung von Archivdateien (zip,rar,etc.) werden standardmässig alle in einem Archiv enthaltenen Dateien (RDF, CoNLL, XML) analysiert. Da es Distributionen von Sprachdaten gibt, die tausende von Einzeldateien enthalten, welche jedoch vom Inhalt mehr oder weniger dieselben Metadaten zu Sprachen und Annotationschemata haben, kann es sinnvoll sein die Analyse nur auf einen Teil aller vorhandenen Dateien zu beschränken, um Zeit zu sparen. Im Dialog (Abb. 26) können Obergrenzen für die Verarbeitung einzelner Dateitypen eingestellt werden.

Parse options			
	RDF	XML	CoNLL
max sample files	50	15	15
activationThreshold	20	10	20
thresholdForGood	20	10	15
thresholdForBad	15	10	15
CLOSE			

Abbildung 26: Analyseparameter

In der ersten Zeile werden mit dem Parameter *max sample files* die maximale Anzahl der zu verarbeitenden Dateien für jeden Dateityp RDF,XML und CoNLL eingestellt (0 = keine Beschränkung). Die eingestellten Werte in Zeile 1 haben jedoch nur einen Effekt, wenn ein Archiv eine bestimmte Mindestanzahl eines Dateityps enthält (siehe Zeile 2 *activationThreshold*). In den Zeilen 3 (*thresholdForGood*) und 4 (*thresholdForBad*) werden Abbruchbedingungen für die Verarbeitung festgelegt. Dabei bezieht sich *Good* auf

Dateien, die Analyseergebnisse geliefert haben (*Bad* keine Analyseergebnisse). Für die Beispieleinstellungen in Abb. 26 kann das an einem Beispiel erklärt werden. Ein Archiv enthält 100 RDF Dateien. Der Aktivierungsthreshold (20) ist überschritten. Daraus folgt das maximal 50 RDF Dateien verarbeitet werden. Sollten zuerst 20 RDF Dateien Analyseergebnisse liefern, dann bricht die Verarbeitung nach 20 positiv analysierten Dateien ab. Im anderen Fall bricht die Verarbeitung nach 15 negativ analysierten Dateien ab.

4.2 Erneute Analyse

Wenn ein Benutzer versucht eine Sprachresource, die bereits im Katalog ist, erneut zu analysieren, gibt es zwei Fälle. Falls die Sprachresource dem Benutzer nicht gehört, dann kann er die Resource nicht erneut analysieren. Stattdessen wird diese automatisch 'gemerkt' und wird fortan in der Liste seiner Sprachresources gezeigt. Ansonsten hat der Benutzer die Option eine erneute Analyse zu starten. Das kann sinnvoll sein, wenn sich eine Resource geändert hat oder Annothub um neue Sprach- oder Annotationsmodelle erweitert wurde. Für die Neuverarbeitung gibt es einige Regeln:

- Wenn eine Resource sich nicht verändert hat dann findet keine Neuverarbeitung statt.
- Es werden nur die Dateien einer Sprachresource neuverarbeitet, die bisher verarbeitet wurden, d.h. bereits in der Datenbank sind. Für eine vollständige Neuverarbeitung muss die Resource vorher gelöscht werden. Andererseits können nicht einzelne Dateien einer Sprachresource neu verarbeitet werden.
- Die Neuverarbeitung aktualisiert gefundene Sprachen und Annotationsmodelle, wobei jedoch zuvor manuell selektierte/deselektierte Sprachen und Modelle beibehalten werden. Wenn z.B. ein Model (Sprache) automatisch vom Parser ausgewählt wurde und danach manuell deselektiert wurde, dann wird diese Auswahl durch eine später folgende Neuverarbeitung nicht verändert.

4.3 Warteliste

Die zur Analyse eingestellten Dokumente werden der Reihe nach abgearbeitet. Der jeweilige Bearbeitungszustand kann in der Katalogliste in der Spalte *Status* abgelesen werden.¹⁰

¹⁰Die Wiederherstellung der Warteliste bei Serverneustart ist implementiert. Die Neustartfunktion vom GUI aus ist noch nicht fertig implementiert.

4.4 Fehlerprotokoll

Nach der Analyse werden nur Dokumente in die Katalogsicht aufgenommen, für die Sprach- und/oder Modellmetadaten generiert werden konnten. Alle anderen Dokumente werden separat im *Error Log* gelistet.¹¹

The screenshot shows a window titled "Uploads with errors or no results" with a close button (X). Below the title bar is a table with 7 columns: Data, File, Format, Processed, Processing time, Result, and Error. The table contains 5 rows of data. At the bottom of the window is a "CLEAR" button.

Data	File	Format	Processed	Processing time	Result	Error
http://resources.mpi-inf.mpg.de/yago-nlp/yago3-1/yagoTypes.ttl.gz	yagoTypes.ttl.rdf	RDF	Sun Apr 21 04:25:27 CEST 2019	PT20.246S	ERROR	Premature end of file.
https://indat.mff.cuni.cz/repository/xmlui/bitstream/handle/11858/00-097C-0000-0001-48FE-9/morce_1.0_linux.tgz?sequence=4	t.dct	RDF	Sun Apr 21 03:32:49 CEST 2019	PT2.354S	ERROR	Failed to determine the content type: (URI=file:/tmp/fid/0/morce_1.0_linux/trained/t.dct : stream=nu
https://indat.mff.cuni.cz/repository/xmlui/bitstream/handle/11858/00-097C-0000-0001-48FE-9/morce_1.0_windows.tgz?sequence=5	t.dct	RDF	Sun Apr 21 01:08:28 CEST 2019	PT1.002S	ERROR	Failed to determine the content type: (URI=file:/tmp/fid/0/morce_1.0_windows/trained/t.dct : stream=
http://hdl.handle.net/11041/ralpe-000853/all-prune-091128.zip	all-prune-091128.zip	UNKNOWN	Sun Apr 21 01:08:20 CEST 2019	PT-3H-41M-34.119S	ERROR	The file type could not be handled
https://indat.mff.cuni.cz/repository/xmlui/bitstream/handle/11858/00-097C-0000-0001-48FE-9/styx-0.9.4-src.tar.bz2?sequence=12	styx-0.9.4-src.tar.bz2?sequence=12	UNKNOWN	Sat Apr 20 22:34:56 CEST 2019	PT-1H-8M-10.172S	ERROR	The file type could not be handled

Abbildung 27: Fehlerprotokoll

Spalte	Beschreibung
Data	Download-URL der Sprachresource
File	Datei in der Sprachresource
Format	Erkanntes Dateiformat
Processed	Datum der Verarbeitung
Processing time	Verarbeitungsdauer (PT-H-M-S)
Result	Analyseergebnis
Error	Fehlermeldung

Tabelle 9: Spalten im Fehlerprotokoll

Das Fehlerprotokoll dient dazu solche Sprachressourcen zu erfassen, die keine Analysere-sultate geliefert haben. Bei der Fehleranalyse kann man grob drei Fälle unterscheiden:

¹¹Mit *CLEAR* wird die Liste geleert. Die Einträge gehen aber nicht verloren (werden bei Serverneustart wieder hergestellt)

- Es konnten keine Sprach- oder Annotationsinformationen ermittelt werden
 - da eine Resource keine Sprach- oder Annotationsinformation enthält
 - da das Format in dem Sprach- oder Annotationsinformation vorliegen nicht erkannt wurde
- Die URL einer Sprachresource liefert keine Daten
 - weil eine Resource offline ist (z.B. Fehler 404)
 - weil die Daten einer Resource wegen eines Fehlers nicht erhältlich sind (Fehlermeldung in der Spalte Error)
- Ein Fehler während der Verarbeitung aufgetreten ist (z.B. Java-Fehlermeldung)

Die hier gelisteten Informationen können auch dazu genutzt werden, um Fehler in der Applikation oder der Konfiguration zu finden.

5 Administration

5.1 Benutzerverwaltung

Die Benutzerverwaltung dient zum Erstellen und Konfigurieren von Benutzerkonten. Es gibt drei verschiedene Kontotypen:

Account Type	Eigenschaften
Guest	Der öffentliche Zugang ist über user=acoli, password=guest möglich. Der Funktionsumfang ist auf die Suche im Katalog und das Herunterladen der Metadaten von Einzelressourcen beschränkt.
Member	Der Funktionsumfang erlaubt das Einstellen von Ressourcen zur Analyse sowie das Editieren der Analyseergebnisse (auch von Sprachressourcen anderer Benutzer, die freigegeben sind). Ausserdem ist die Kommentarfunktion aktiv, die es erlaubt Textbeiträge zu einzelnen Ressourcen zu erstellen, und die Kommentare anderer Benutzer zu lesen.
Admin	Keine Einschränkungen

Retired	Ein aktives Benutzerkonto lässt sich (auch vorübergehend) deaktivieren. Danach kann sich der Benutzer nicht mehr einloggen, jedoch sind alle Sprachressourcen, die zu dem Konto gehören, weiterhin im Katalog verfügbar
----------------	---

Tabelle 10: Kontotypen

Allgemein umfasst ein Benutzerkonto die Rechte zum Ausführen bestimmter Funktionen (z.B. Hochladen von Ressourcen) und auch die Sprachressourcen, die von einem Benutzer analysiert wurden. Ein registrierter Benutzer hat das exklusive Recht (neben dem Admin) :

- die Analyseergebnisse seiner Sprachressourcen zu editieren
- die vom ihm hochgeladenen Sprachressourcen aus dem Katalog zu löschen
- seine Sprachressourcen für andere Benutzer zum Editieren freizugeben

In der Benutzerverwaltung (Abb. 28) stehen Funktionen zum Erstellen neuer Benutzerkonten zur Verfügung. In der Kopfzeile des Fensters findet man eine Übersicht aller existierenden Benutzer. Alternativ lassen sich Benutzerkonten auch über die Kommandozeile (Server) erstellen. So können Administrator Konten nur von dort aus erstellt und auch gelöscht werden.

The screenshot shows a 'User Management' window. At the top, it displays statistics: 'Users : Admins : 1 Members : 0 Guests : 0 Total : 1 Retired : 0 Online : 1'. Below this is a 'Selected User' dropdown menu currently showing 'ub', with buttons for 'Save User', 'New User', and 'Delete User'. The main section is divided into 'User data' and 'Quota'. Under 'User data', there are fields for 'Login' (containing 'ub'), 'New Password', 'Repeat Password', and 'Account Type' (set to 'ADMIN'). Under 'Quota', there are three spinner controls for 'Max resource uploads' (1000), 'Max total resource files' (1000), and 'Max upload filesize [in MB]' (1500). A 'CLOSE' button is at the bottom left.

Abbildung 28: Benutzerverwaltung

Funktion	Beschreibung
Select User	Wählt einen Benutzer aus
Save User	Speichert Änderungen ab. Wird auch zum Erstellen eines neuen Benutzerkontos verwendet.
New User	Erstellen eines neuen Benutzerkontos
Delete User	Löscht ein Benutzerkonto. Es ist zu beachten, dass damit auch alle Ressourcen, die zu einem Konto gehören irreversibel gelöscht werden. Falls das nicht gewünscht ist, sollte das Benutzerkonto mit <i>Retired</i> deaktiviert werden. Es kann danach später wieder aktiviert werden, indem man den Kontotyp wieder auf <i>Member</i> setzt.

Tabelle 11: Verwaltungsfunktionen

In der **Quota** Sektion kann die Zahl der Sprachressourcen, die ein Benutzer hochladen darf, eingestellt werden. Der Parameter **Max resource uploads** bestimmt die maximale Anzahl hochladbarer Ressourcen. Der Parameter **Max total resource files** bezieht sich auf die in Archiven enthaltenen Dateien. Zuletzt kann mit **Max upload file size** die Grösse hochladbarer Sprachressourcen beschränkt werden.

5.2 Sprachprofile

Die Spracherkennung verwendet sog. Sprachprofile für das Erkennen der Textsprache(n) in Sprachressourcen. Dazu stehen Profile für ca. 500 Sprachen bereits zur Verfügung. Um die Spracherkennung um neue Sprachen zu erweitern oder die Spracherkennung für Sprachen zu verbessern, für die schon ein Profil existiert, stehen umfangreiche Funktionen zur Verfügung.

Language Profile Manager				
✕ EXIT				
(1 of 52) 1 2 3 4 5 6 7 8 9 10 >> >				
ISO-639 ▲	Language ⇅	1-grams ⇅	2-grams ⇅	3-grams ⇅
aai	Arifama-Miniafia	50	523	2214
acf	Saint Lucian Creole French	54	608	2475
acr	Achi	60	693	2781
af	Afrikaans	71	547	1525
agr	Aguaruna	58	708	3025
agu	Aguacateco	62	936	4450
agw	Kahua	40	391	2353
aii	Assyrian Neo-Aramaic	24	486	3974
ake	Akawaio	33	319	1628
alb	unknown	54	705	3468
(1 of 52) 1 2 3 4 5 6 7 8 9 10 >> >				

Abbildung 29: Sprachprofile

Spalte	Beschreibung
ISO-639	Der ISO-639 Sprachcode. Bis auf wenige (ca. 70) Profile, die als ISO-639-2 Code gelistet sind, sind alle Sprachen als ISO-639-3 Code angegeben.
Language	Der englische Sprachname (aus den ISO-639 Code-Tabellen).
n-grams	N-Gramme sind hier Zeichenketten der Länge n, die im Trainingstext vorkommen. Aus deren Verteilung lässt sich letztendlich eine Wahrscheinlichkeit über die Textsprache berechnen. Für die Erstellung eines Sprachprofils ist es wichtig einen geeigneten Trainingstext zu verwenden, damit man eine repräsentative Verteilung der n-Gramme einer Sprache erhält.

Tabelle 12: Sprachprofil Attribute

Über das Kontextmenü eines Profils stehen Funktionen zum Testen, Löschen oder Erstellen eines neuen Profils zur Verfügung.

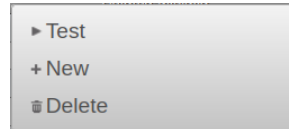


Abbildung 30: Kontextmenü

Um ein neues Sprachprofil zu erstellen muss ein Trainingstext (als Textdatei) hochgeladen oder in das Textfeld eingefügt werden. Nach der Analyse wird die Anzahl der gefundenen n-Gramme in der Tabelle, zusammen mit einem Dummy-Sprachcode ausgegeben. Anschliessend lässt sich das Profil mit **Save** unter Angabe eines ISO-639-3 Sprachcodes abspeichern.

Abbildung 31: Sprachprofil erstellen

Ein Sprachprofil lässt sich mit einem beliebigen Text testen. Das Ergebnis der Evaluation sind die Sprachprofile, die am besten zur Testeingabe passen. Die Evaluationsergebnisse umfassen die Anzahl der Sätze für die ein Sprachprofil als zutreffend assoziiert wurde. Die Werte von *Min-Prob* und *Max-Prob* bezeichnen die kleinste/grösste gemessene Wahrscheinlichkeit für diese Sätze. Der Wert von *Avg-Prob* macht eine Aussage über die Wahrscheinlichkeit einer Sprache für den ganzen Text. Es ist $\text{avg-prob} = \text{sum}(\text{prob}_i) / (\text{\#Sätze im Text})$, wobei prob_i die Wahrscheinlichkeit für das Sprachprofil in einem assoziiertem Satz ist.

ISO-639	Language	1-grams	2-grams	3-grams
aal	Aritama-Miniafia	50	523	2214

Test Profile [v] Upload Evaluation Data Start evaluation Save

Input Text Sample

Selected	Language	ISO-639	#sentences	min-prob	max-prob	avg-prob
No records found.						

Abbildung 32: Sprachprofil testen

5.3 Backup

Mit dem Backup-Manager lassen sich Backups erstellen und wiederherstellen. Die Daten, die gesichert werden sind die zwei Datenbanken in denen alle Annohub Daten gespeichert sind. Da die eingesetzten Neo4j Instanzen ein online-Backup nicht unterstützen muss dafür der Annohub Server heruntergefahren, Kopien der Datenbankverzeichnisse erstellt, und danach der Server neugestartet werden. Die vollständige Automatisierung dieses Prozesses ist komplex und umfasst das Ausloggen aller Benutzer und das Sichern/Wiederherstellen der Warteschlange. Ebenso dauert der Serverneustart einige Stunden, da beim Start alle Analyseergebnisse neugeladen werden. Die Backupfunktion über den Backup Manager ist z.Z. in der Testphase, und sollte deswegen nicht verwendet werden. Ein manuelles Backup ist jederzeit auf dem Server mit dem Sichern der Datenbankverzeichnisse *Databases.Registry.Neo4jDirectory* und *Databases.Data.Neo4jDirectory* möglich.

Name	Date	Gremlin version	Reg-DB version	Data-DB version
No records found.				

Abbildung 33: Backup Verwaltung

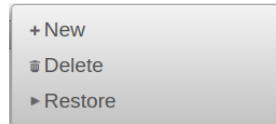


Abbildung 34: Kontextmenü

Für das Erstellen eines neuen Backups muss ein Name vergeben werden. Alle anderen Angaben sind optional. Diese sind *Gremlin-Version*, die Version der verwendeten Tinkerpop-Instanz, *Registry/Data-DB version* ist die Version des Datenbanktreibers für Neo4j (Gremlin-Plugin) und schliesslich ein Textkommentar.

 A dialog box titled 'Create New Backup'. It contains five input fields arranged vertically: 'Backup name', 'Gremlin version', 'Registry DB version', 'Data DB version', and 'Backup Comment'. At the bottom of the dialog are two buttons: 'CANCEL' and 'START BACKUP'.

Abbildung 35: Neues Backup erstellen

5.4 OLiA Manager

Vor der ersten Verwendung von Annohub muss die Modelldatenbank mit den OLiA-Core, Linking und Annotationsmodellen initialisiert werden. Dazu werden die Ontologiedateien eingelesen, die in der Modelldefinitionsdatei *ModelDef.json* gelistet sind, und in einer Graphdatenbank abgespeichert. Danach verwendet Annohub diese Definitionen, solange bis die Modelldatenbank aktualisiert wird.

OLiA Modelle Die für die Analyse verwendeten OLiA Modelle werden in einer Konfigurationsdatei definiert.

```
{
  "modelID": "BLL",
  "documentationUrl": "https://data.linguistik.de/bll/index.html",
  "niceName": "Bll",
  "namespaces":
  [
    "http://data.linguistik.de/bll/bll-ontology"
  ],
}
```

```

"files":
[
  {
    "url": "http://purl.org/olia/bll-link.rdf",
    "modelUsage": "LINK",
    "active": true,
    "documentationURL": "https://data.linguistik.de/bll/index.html"
  }
]
}

```

Listing 1: Auszug aus ModelDef.json

Attribut	Beschreibung
modelID	Der interne Modelname wird in Java als ID verwendet. Dieser sollte nur Grossbuchstaben enthalten und nicht länger als 5 Zeichen sein.
documentationUrl	Allgemeine Dokumentation zu einem Modell
niceName	Wird u.a. für den JSON Export verwendet
namespaces	Wird für die Modellerkennung verwendet. Namespaces sind URL-Präfixe die spezifisch für die Klassen einer Ontologie verwendet werden
files	
url	URL unter der eine Ontologie verfügbar ist
modelUsage	SYSTEM LINK ANNOTATION <i>SYSTEM</i> wird nur für die OLiA-Core Ontologien verwendet. Entsprechend <i>LINK</i> für Linkingmodelle und <i>ANNOTATION</i> für Annotationsmodelle
active	true false
documentationURL	Dokumentation

Tabelle 13: Attribute einer Modelldefinition

Die aktuell verwendeten Modelldefinitionen aus *ModelDef.json* lassen sich im Ontology-Manager ablesen.

Ontology Manager				
<div> <div> <div></div> <div>Exit</div> </div> <div> <div>(1 of 11)</div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>6</div> <div>7</div> <div>8</div> <div>9</div> <div>10</div> </div> </div> </div>				
Status ▾	URL ▾	Type ▾	Content ▾	Documentation ▾
BROKEN	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/stable/ubyCat.owl	UBYCAT	ANNOTATION	http://www.acoli.informatik.uni-frankfurt.de/resources/olia/html/stable/ubyCat.html
BROKEN	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/stable/ubyCat-link.rdf	UBYCAT	LINK	http://www.acoli.informatik.uni-frankfurt.de/resources/olia/html/stable/ubyCat-link.html
UP2DATE	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/experimental/univ_dep/all_from_rdfa/ud-dep-all.owl	UD1DEP	ANNOTATION	
UP2DATE	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/experimental/univ_dep/all_from_rdfa/ud-dep-all-link.rdf	UD1DEP	LINK	
UP2DATE	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/experimental/univ_dep/all_from_rdfa/ud-feat-all.owl	UD1FEAT	ANNOTATION	http://ginter.github.io/docs/u/feat/all.html
UP2DATE	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/experimental/univ_dep/all_from_rdfa/ud-pos-all.owl	UD1POS	ANNOTATION	http://ginter.github.io/docs/u/pos/all.html
UP2DATE	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/experimental/univ_dep/all_from_rdfa/ud-pos-all-link.rdf	UD1POS	LINK	
OUTDATED	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/stable/alpino.owl	ALPINO	ANNOTATION	
OUTDATED	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/stable/alpino-link.rdf	ALPINO	LINK	
OUTDATED	https://raw.githubusercontent.com/acoli-repo/olia/master/owl/stable/ancorra.owl	ANCORRA	ANNOTATION	http://www.acoli.informatik.uni-frankfurt.de/resources/olia/html/stable/ancorra.html
<div> <div>(1 of 11)</div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>6</div> <div>7</div> <div>8</div> <div>9</div> <div>10</div> </div> </div>				

Abbildung 36: OLiA Ontologie Verwaltung

Spalte	Beschreibung
Status	Zeigt an, ob die aktuell verwendeten Modelldefinitionen noch aktuell sind. Mögliche Zustände sind <i>UP2DATE</i> , <i>OUTDATED</i> und <i>BROKEN</i> .
URL	URL der aktuell verwendeten Ontologiedatei
Type	Entspricht der <i>modelID</i> in der Modeldefinitionsdatei
Content	Entspricht <i>modelUsage</i> in der Modeldefinitionsdatei
Documentation	Entspricht <i>files:documentationURL</i> in der Modeldefinitionsdatei

Tabelle 14: Ontology Manager Infos

Im Kontextmenü eines Modelleintrags stehen Funktionen zum Bearbeiten und Löschen zur Verfügung. Damit kann z.B. die URL einer OLiA Modelldatei angepasst werden, falls sich diese geändert hat.

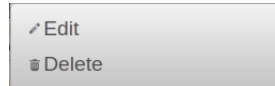


Abbildung 37: Kontextmenü

Unter dem Reiter Werkzeuge gibt es weitere Funktionen:

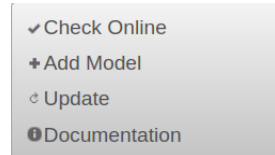


Abbildung 38: Werkzeuge

Funktion	Beschreibung
Check Online	Überprüft den Online-Status aller Modelldateien
Add Model	Hinzufügen eines neuen OLiA Annotationsmodells (siehe Abb. 39)
Update	Startet den Update Prozess
Documentation	Dokumentation

Tabelle 15: Werkzeuge

Add new model definition

URL

Documentation

Model ID

ADD

Type

ANNOTATION

CANCEL

SAVE

Abbildung 39: Neues Modell erstellen

Aktualisierung Bei einer Aktualisierung (**Update**) werden alle Modelle neu in die Graphdatenbank geladen und es werden alle Verknüpfungen zu bisherigen Treffern (gefundenen Annotationen) aktualisiert. Im Idealfall wird man eine Aktualisierung nur vornehmen, um verbesserte Ontologiedefinitionen auf die Treffermenge anzuwenden. Sollten jedoch bei einer Aktualisierung Modelldefinitionen fehlen, die bisher vorhanden waren, dann werden möglicherweise bisher erkannte Annotationen nicht mehr als Treffer in der Analyse erscheinen und es werden im schlechtesten Fall die falschen Modelle ausgewählt. Um solchen Fehlern vorzubeugen wird vor jeder Aktualisierung automatisch ein Backup durchgeführt, so dass der alte Zustand vor der Aktualisierung wieder hergestellt werden kann.

6 Datenbanken

Annohub verwendet zwei Datenbanken, eine für die Informationen über verarbeitete Sprachressourcen (REG-DB) und eine zweite für die Berechnung der Annotationsmodelle (MOD-DB). Hierzu wird das Apache-Tinkerpop¹² Framework für Graphdatenbanken eingesetzt, das es erlaubt aus verschiedenen Graphdatenbank-Implementierungen auszuwählen. Annohub verwendet *Neo4j* für beide Datenbanken. Im einzelnen wird die Registrierungsdatenbank über einen Webserver (gremlin Server) betrieben, d.h. Queries werden über das Http-Protokoll verarbeitet. Hingegen wird für die Modelldatenbank die *embedded* Version von Neo4j verwendet, so dass Anfragen direkt von Java aus stattfinden.

6.1 Registrierungs Datenbank

Die Registrierungsdatenbank hat Informationen über jede Sprachresource, die die NLP-Pipeline durchlaufen hat.

- HTTP-Header Information, wie z.B. der MIME Typ, Grösse einer Resource, *last-modified-date*, etc.
- Informationen über alle verarbeiteten Dateien einer SprachresourceInfo wie z.B. Dateigrösse /-Typ/-Format, etc.
- Die Ergebnisse der NLP-Analyse, die die gefundenen Annotationsmodelle, Sprachen und RDF-Vokabulare, jedoch nicht Detailinformation wie die einzelnen erkannten Annotationen beinhalten.

In REG-DB wird jede Sprachresource mit einem Resourceknoten repräsentiert, der mit einem oder mehr Dateiknoten verbunden ist. Diese sind wiederum mit Modell-, Sprach- oder Vokabularknoten verbunden. Allgemeine Metadaten sind bibliographische Daten,

¹²<http://tinkerpop.apache.org/>

wie Author, Titel, etc., die zu jeder Sprachresource gehören. Diese Informationen werden in Metadatenknoten abgespeichert, die mit den Resourceknoten verbunden sind.

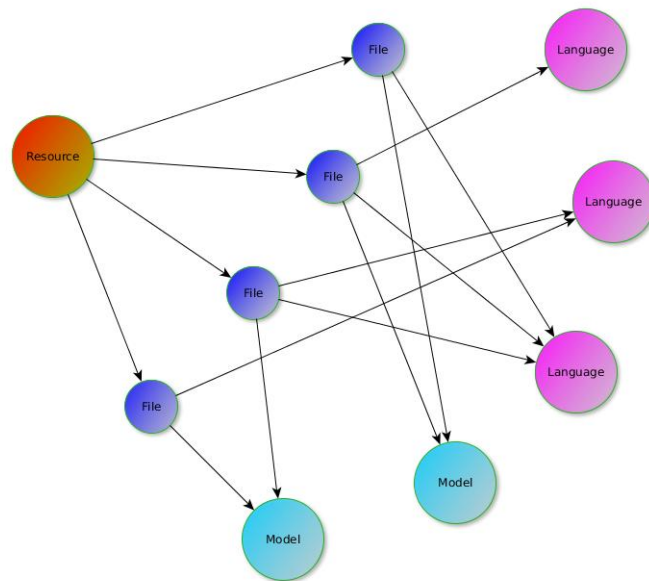


Abbildung 40: Grundlegende Struktur der REG-DB

6.2 Modell Datenbank

Die Modelldatenbank (MOD-DB) wird dazu benutzt, um die eigentliche Berechnung der Annotationsmodelle für jede Sprachresource durchzuführen. Die Klassenrelationen von OLiA-Annotationsmodellen (z.B. in <http://purl.org/olia>) sowie der BLL Ontologie¹³ werden in der MOD-DB auf einen Graph abgebildet. Dieser Graph wird vervollständigt mit Knoten für alle gefundenen Annotationen in analysierten Sprachresources, die mit einem OLiA Annotationsmodell identifiziert werden können, bzw. allen Annotationen aus CoNLL Sprachresources.

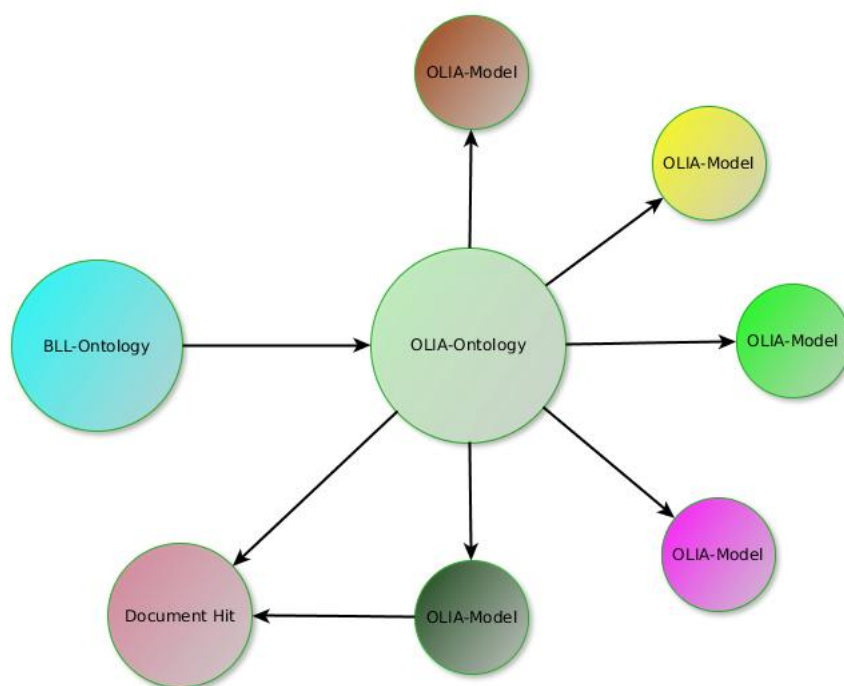


Abbildung 41: Grundlegende Struktur der Modelldatenbank

OLiA modelliert linguistische Annotationen aus den Bereichen Syntax und Morphologie in einer hierarchischen Klassenstruktur mit RDF. Dazu werden RDF-Attribute wie *subClassOf*, *intersection*, *union*, *complement* und *equivalentClass* verwendet. Diese Relationen werden im Graph auf Kanten zwischen OLiA Klassen abgebildet. Die Annotationen (Tags oder URLs) der Sprachresources werden über Kanten mit allen treffenden OLiA-Klassen verbunden. Durch die Traversierung des dadurch entstandenen Graphs kann eine Prognose über die für eine Resource verwendeten Annotationsmodelle gemacht werden. Als zweites Ergebnis können gefundenen Annotationen BLL-Konzepte der BLL-Ontologie zugeordnet werden.

¹³<https://www.linguistik.de/de/lod/>

7 Installation und Konfiguration

Für die Installation ist es nicht erforderlich die Neo4j Datenbank nativ zu installieren. Stattdessen muss lediglich ein Neo4j-Treiber für das Tinkerpop Framework installiert werden.

7.1 Voraussetzungen

- Linux/Unix Distribution
- Java ≥ 1.8 (tested)
- Apache Tinkerpop $\geq 3.3.3$ (getestet)
- Apache TomEE ≥ 7.1 (getestet)
- Blazegraph $\geq 2.0.1$
- 7z (7za) Archivierungstool
- Rapper RDF Tool (<http://librdf.org/raptor/>)
- maven (<https://maven.apache.org/>)

7.2 Build

Build mit maven : *maven install clean*. Für den Start der Web-Applikation muss die von maven erzeugte WAR Datei im Web-Server deployed werden.

7.3 Initialisierung

Bevor Annohub verwendet werden kann müssen die Datenbanken initialisiert werden. Dies geschieht mit dem Kommando *fid -init*. Dadurch werden die Ontologieninformationen aus OLiA Ontologien und der BLL Ontologie, deren URLs in der Datei ModelDef.json definiert sind in die MOD-DB geladen. Die Initialisierung löscht dabei alle vorherigen Informationen in der REG-DB und MOD-DB.

7.4 Konfigurationsdatei (FIDConfig.xml)

Es gibt verschiedene Möglichkeiten für die Bereitstellung der Konfigurationsdatei

1. Durch Setzen der Umgebungsvariable FID_CONFIG_FILE
2. Beim Start von der Kommandozeile kann die -CX Option verwendet werden
3. Die Web-Applikation verwendet standardmäßig die Datei /WEB-INF/classes/FIDConfig.xml

Die Tabelle unten zeigt eine Liste aller Konfigurationsparameter.

7.5 Konfigurationsparameter

Databases		
GremlinServer.home	Ordner	Home Verzeichnis der Tinkerpop Installation
GremlinServer.conf	Pfad	Pfad zur Gremlinserver Konfigurationsdatei (gremlin-server-neo4j.yaml)
Registry.Neo4jDirectory	Ordner	Verzeichnis der Neo4j Registrierungs-Datenbank (REG-DB)
Data.Neo4jDirectory	Ordner	Verzeichnis der Neo4j Modell-Datenbank (MOD-DB)
GremlinServer.port	Integer	Gremlin-Server [8182] für Zugriff auf Neo4j Datenbanken
Blazegraph.loadProperties	Pfad	Blazegraph Datenbank Konfigurationsdatei (blazegraph.properties)
Postgres.usePostgres	Boolean	[false] Postgres Datenbank verwenden
Postgres.keyFile	File	SSH key Datei (nur für remote Zugriff)
Postgres.remoteHost	String	URL falls Postgres nicht lokal
Postgres.database	String	Name der Datenbank
Postgres.databaseUser	String	Postgres Benutzer
Postgres.databasePassword	Boolean	Postgres Passwort
Postgres.sshUser	String	SSH user
deleteRdfDataAfterIndex	Boolean	[false]
retryUnsuccessfulRdfData	Boolean	[false]
restartTimeoutInMilliseconds	Integer	[10000]
RunParameter		
downloadFolder	Ordner	Ordner in dem hochgeladene Sprachressourcen temporär abgelegt werden
htmlFolder	Pfad	nicht verwendet
urlSeedFile	Pfad	[/tmp/urlSeedFile]
urlPoolFile	Pfad	[/tmp/urlpool]
urlFilter	String	CONLL,RDF,ARCHIVE

updatePolicy	String	[UPDATE_NEW]
threads	Integer	[1] Workerthreads (z.Z. wird nur ein Thread unterstützt)
decompressionUtility	Pfad	Pfad zu Linux-Archivtool 7z oder 7za
RdfPredicateFilterOn	Boolean	[false] (immer aus)
ExitProcessDiskSpaceLimit	Integer	[1000] MB, Bei weniger freiem Speicherplatz stoppt die Verarbeitung automatisch
MaxArchiveFileCount	Integer	[30000] Maximal erlaubte Dateianzahl eines Archivs
compressedFileSizeLimit	Byte	[2048576000] Maximal erlaubte Grösse eines Archivs (1 GB = 1073741824 Bytes)
uncompressedFileSizeLimit	Byte	[2048576000] Maximal erlaubte Grösse einer Sprachdatei
isoCodeMapDirectory	?	TODO set priority
XMLParserConfiguration. matchingMeasurement	String	[RECALL] (intern)
XMLParserConfiguration. sampleSentenceSize	Integer	[10] (intern)
startExternalQueue	Boolean	[true] (nicht verändern)
OptimaizeExtraProfiles Directory	Ordner	Ordner für Sprachprofile (ca. 500 St.)
OptimaizeManualProfiles Directory	Ordner	Ordner für über Annohub-Interface erstellte Sprachprofile
OptimaizeAnnotationModels ProfilesDirectory	?	?
LexvoRdfFile	File	
RdfExportFile	File	[/tmp/FidExport.rdf] Exportierte RDF Datei
JsonExportFile	File	[/tmp/JsonExport.json] Exportierte JSON Datei
AnnohubRelease	File	[/tmp/AnnoHubDataset.rdf] Exportierte RDF Datei
RdfPredicateFilterOn	Boolean	false ??
useBllOntologiesFromSVN	Boolean	false
BLLontologiesDirectory	String	false
convert2RdfXmlScript	String	/bash/convert2RdfXml
debugOutput	Boolean	[true] Speichere Debug Information in Tomcat catalina.out
guiPropertiesFile	?	Speichert sichtbare Spalten in Annohub Editoransicht

ServiceUploadDirectory	Ordner	[/tmp] Ordner in dem hochgeladene Sprachressourcen temporär abgelegt werden
defaultResourcePermissions	Integer	701 (owner/group/world), 1=Read,2=Edit,4=Export
cached	Boolean	true (Muss immer an sein)
loadUnsuccessfull	Boolean	false
initRdfExporterAtServerStart	Boolean	true
checkBrokenLinksAtServerStart	Boolean	[false] Testet auf kaputte Daten URLs aller Ressourcen im Katalog
exportBrokenLinks	Boolean	[false] Gibt an, ob Ressourcen, die aktuell nicht mehr online sind im Export erscheinen
checkBrokenLinksInterval	Tage	[0] Zeit nach der Sprachressourcen im Katalog überprüft werden
publishRDFExportInterval	Tage	[0] Zeit nach der der Export automatisch stattfindet
JavaHome	Ordner	Java Homeverzeichnis
QueueBackupFile	Pfad	Sichert die Warteschlange in Datei. Dient zur Wiederherstellung der Warteschlange beim Serverneustart
Quotas		
maxResourceUploads	Integer	[10] Erlaubte Zahl von Ressourcen
maxResourceFiles	Integer	[100] Erlaubte Zahl von hochgeladenen Dateien
maxResourceUploadSize	Integer	[200] in MB, Erlaubte Grösse von hochgeladenen Dateien
Linghub		
linghubDataDumpURL	URL	URL des Linghub Datendump http://linghub.org/linghub.nt.gz
resourceQueries	?	linghubResourceQueries
metadataQueries	?	linghubMetadataQueries
statusCodeFilter	?	?
useQueries	Boolean	false
enabled	Boolean	false
forceUpdate	Boolean	false
Backup		
autobackupInterval	Tage	[0] manuell oder Tage zwischen Backups
ActiveMQ		
brokerUrl	URL	<code>tcp://localhost:61616</code>
OWL		

BLL.BllOntology	URL	https://valian.uni-frankfurt.de/svn/repository/intern/Virtuelle_Fachbibliothek/UB/OWL/BLLThesaurus/bll-ontology.rdf
BLL.BllLink	URL	https://valian.uni-frankfurt.de/svn/repository/intern/Virtuelle_Fachbibliothek/UB/OWL/BLLThesaurus/bll-link.rdf
BLL.BllLanguageLink	URL	https://valian.uni-frankfurt.de/svn/repository/intern/Virtuelle_Fachbibliothek/UB/OWL/BLLThesaurus/bll-language-link.ttl
modelUpdateMode	String	manuell
modelUpdateHitDeletePolicy	String	manuell
checkModelsOnlineAtStartup	Boolean	false
checkModelsOnlineAtStartup StopOnFail	Boolean	false
Sampling		
maxSamples	Integer	(-1 für unbegrenzt) Maximale Samples (über alle Archivunterordner)
thresholdForGood	Integer	Abbruch, sobald thresholdForGood Dateien mit Ergebnis gefunden
thresholdForBad	Integer	Abbruch, sobald thresholdForBad Dateien ohne Ergebnis gefunden
activationThreshold	Integer	Threshold, ab dem gesampelt wird
Rdf.maxSamples	Integer	[100]
Rdf.activationThreshold	Integer	[50]
Rdf.thresholdForGood	Integer	[20]
Rdf.thresholdForBad	Integer	[10]
Xml.maxSamples	Integer	[15]
Xml.activationThreshold	Integer	[10]
Xml.thresholdForGood	Integer	[3]
Xml.thresholdForBad	Integer	[2]
Conll.maxSamples	Integer	[15]
Conll.activationThreshold	Integer	[20]
Conll.thresholdForGood	Integer	[3]
Conll.thresholdForBad	Integer	[3]
Processing		
ConllParser.conllFileMinLine Count	Integer	[10] Überspringe CoNLL Dateien mit weniger als 10 Zeilen

ConllParser.conllFileMaxLineCount	Integer	[-1] (unbegrenzt)
ConllParser.maxSampleSentenceSize	Integer	[100]
ConllParser.modelSampleSentenceMinTokens	Integer	[40]
ConllParser.languageSampleSentences	Integer	[15]
ConllParser.languageSampleSentencesMinTokenCount	Integer	[10]
GenericXmlFileHandler.xmlValueSampleCount	Integer	[10]
GenericXmlFileHandler.makeConllMode	String	[sample] (auto sample full), Falls makeConllMode=auto dann verwende volle Konversion für Dateien, die grösser als makeConllAutoMaxFileSize MB sind, sonst Sampling
GenericXmlFileHandler.makeConllSampleSentenceCount	Integer	[5000] Sätze
GenericXmlFileHandler.makeConllAutoMaxFileSize	Integer	[5] MB
GenericXmlFileHandler.makeConllConverterChoice	String	[generic]
XMLAttributeEvaluator.processDuplicates	Boolean	[false]
ModelEvaluator.autoDeleteConllModelsWithTrivialResults	Boolean	[false]
AccountProperties		
uploadResourceCountLimit.MEMBER	Integer	[50]
uploadResourceCountLimit.ADMIN	Integer	[-1] (unbegrenzt)
uploadResourceFileCountLimit.GUEST	Integer	[20]
uploadResourceFileCountLimit.MEMBER	Integer	[50]
uploadResourceFileCountLimit.ADMIN	Integer	[-1] (unbegrenzt)
uploadTotalSizeLimit.GUEST	Integer	[500] Uploads in MB
uploadTotalSizeLimit.MEMBER	Integer	[5000]
uploadTotalSizeLimit.ADMIN	Integer	[-1] (unbegrenzt)
Clarín		

clarinQueries	String	SELECT title, description, resource_type, date, author, licence, publisher, language from metadata where link = 'ACCESSURL';
---------------	--------	--

8 Kommandozeilen Interface

```
usage: program [-CU <userName> <password>] [-CX <configfile>] [-DU
<userName> <password>] [-EX] [-h] [-IN] [-PA] [-SD <seedfile>] [-SP
<userName> <rights>]
```

```
-CU,--create-user <userName> <password>    Create user
-CX,--config-file <configfile>              Provide configuration file
-DU,--delete-user <userName> <password>     Delete user
-EX,--execute                                Run
-h,--help                                    Show this help
-IN,--init                                   Initialize application -
                                              deletes all data !
-PA,--database-patch                         Run a database patch - see
                                              code in
                                              Executer.executePatch()
-SD,--seed-file <seedfile>                  Provide seed file with URLs to
                                              be processed
-SP,--set-privileges <userName> <rights>    Set privileges
```

Option	Beschreibung
-CX, -config	Angabe der Konfigurationsdatei FidConfig.xml
-IN, -init	Initialisierung der Applikation (löscht alle Daten)
-SD, -seed	Angabe einer Datei mit Daten-URLs, die verarbeitet werden sollen
-CU, -create-user	Benutzerkonto erstellen
-DU, -delete-user	Benutzerkonto löschen
-SP, -set-privileges	Rechte für Benutzerkonto setzen
-EX, -execute	Start der Applikation (ohne Webserver)
-PA, -database-patch	Datenbank-Patch ausführen (Aufruf der Methode Executer.executePatch())

Um Sprachressourcen zu analysieren gibt es folgende Möglichkeiten:

- Hochladen von Sprachressourcen in der Web-Applikation
- Starten lokal mit *fid -SD Datei*

Nachdem die Verarbeitung beendet ist können die Ergebnisse (nur) in der Web-Applikation angeschaut werden.

9 Fragen und Antworten

Problem	Antwort
Das GUI-Fenster reagiert nicht	Timeout ist abgelaufen - bitte neu einloggen

Literatur

- [AFG20] ABROMEIT, Frank ; FÄTH, Christian ; GLASER, Luis: Annohub – Annotation Metadata for Linked Data Applications Data. In: *Proceedings of the Thirteenth International Conference on Language Resources and Evaluation (LREC 2020)*. Marseille, France : European Language Resources Association (ELRA), 2020