Proceedings of the Linked Data in Linguistics Workshop @ LREC2022, pages 17–25
Marseille, 24 June 2022

IMTVault: Extracting and Enriching Low-resource Language Interlinear Glossed Text from Grammatical Descriptions and Typological Survey Articles

Sebastian Nordhoff, Thomas Krämer
Language Science Press, gesis
sebastian.nordhoff@langsci-press.org, thomas.kraemer@gesis.org

Abstract

Many NLP resources and programs focus on a handful of major languages. But there are thousands of languages with low or no resources available as structured data. This paper shows the extraction of 40k examples with interlinear morpheme translation in 280 different languages from LaTeX-based publications of the open access publisher Language Science Press. These examples are transformed into Linked Data. We use LIGT for modelling and enrich the data with Wikidata and Glottolog. The data is made available as HTML, JSON, JSON-LD and N-quads, and query facilities for humans (Elasticsearch) and machines (API) are provided.

Keywords: Interlinear Glossed Text, Extraction, Linked Data, Low-Resource Languages, FAIR, Linguistic Linked Open Data

1. Introduction

There are currently 7616 spoken languages on Earth.[1] Digital resources for these languages are in a very skewed distribution, as surveyed by (Joshi et al., 2020). English has good resources, a few additional languages have satisfactory resources and three other groups of languages can at least list some resources of a certain size or quality. These four groups arrive at 72 languages altogether. The great majority of languages, however, have only minimal resources in both extent and annotation depth, and many languages have no resources available for NLP at all. The latter two groups comprise 93.87% (2 413) of all languages investigated by (Joshi et al., 2020) (Table 1). Beyond that, there are another 5 000 languages which did not even make it into the (Joshi et al., 2020) survey. As we start the International Decade of Indigenous Languages[2] in 2022, this very skewed distribution is concerning.

2. Low Resource Languages and Diversity Linguistics

While the NLP community has not produced structured datasets for these low/no resource languages, structured data does indeed exist within the field of Diversity Linguistics. Diversity Linguistics is the field which concerns itself with the variety of languages spoken in the world. This concerns in-depth treatment of a particular language (grammatical description) as well as large-scale comparison of a given phenomenon (e. g. position of the verb before or after the object) in hundreds or thousands of languages. This comparative work can be found in articles in journals or edited volumes, in monographs, or also in databases. We can name AUTOYP[3] or the CLLD datasets (WALS,[4] APiCS[5]), of which there are 19 as of 2022.

The academic inquiry is complemented by language archives where audiovisual data are stored, some of them transcribed, translated and glossed, in varying percentages. We can name ELAR,[6] AILLA,[7] TLA,[8] Paradisec.[9] See (Nordhoff, 2020a) for a breakdown of their accessible holdings.

These different data sources have been tapped into over time: academic books and articles ((Lewis and Xia, 2010; Xia et al., 2014)), typological databases ((Chiarcos and Ionov, 2019; Ionov, 2021)), and language archives ((Nordhoff, 2020a; Nordhoff, 2020b; von Prince and Nordhoff, 2020)), producing structured data which allows for programmatic and quantitative approaches.

3. The Example Sentence

While the field of Diversity Linguistics is actually quite far from NLP in its practices, it produces nevertheless semi-structured texts. This structure can be exploited to retrieve meaningful elements. The most common datatype is the linguistic example with interlinear morpheme translation (IMT). In this kind of element, we have part-whole relations between morphemes, words and sentences, and translational equivalence relations on the word level and the sentence level between the source language (white) and the translation (grey). This is shown in Figure 1.

From examples like this, we can extract morpheme-to-morpheme translations, which can be used to populate

[1] https://glottolog.org/glottolog/glottologinformation

[2] https://en.unesco.org/idil2022-2032

[3] https://github.com/autotyp/autotyp-data/tree/v1.0.0

[4] http://wals.info

[5] http://apics-online.info

[6] https://www.elararchive.org

[7] https://ailla.utexas.org

[8] https://archive.mpi.nl/tla

[9] https://catalog.paradisec.org.au

		criteria				
	Class	unlabeled data	labeled data	example	# lgs	%
5	winners	good	good	Spanish	7	0.28
4	underdogs	good	insufficient	Russian	18	1.07
3	rising stars	good	none	Indonesian	28	4.42
2	hopefuls	?	smallish sets	Zulu	19	0.36
1	scraping-bys	smallish	none	Fijian	222	5.49
0	left-behinds	none	none	Warlpiri	2 191	88.38

Table 1: Joshi et al's classes

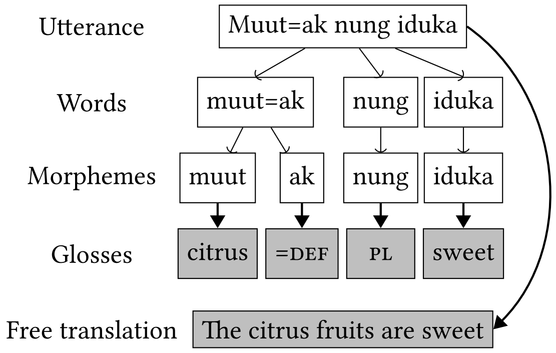


Figure 1: An example of interlinear text (`https://imtvault.org/b/157/ex/wl09-cb9806ea53.htm`, (Klamer et al., 2017)). Light arrows denote part-whole relations; thick arrows denote translational equivalents. Note that there is no translation for the word level.

a dictionary or a word list. The data model used here is discussed in more detail in Section 4. Data sources for interlinearized examples can be found in a variety of places in different formats (see Section 7.4).

4. Data Modelling

The interlinear sentence has received quite some theoretical treatment. The first technical approach was the implementation in the program Shoebox, which would later become Toolbox.[10] The representation used therein was actually never intended to be used in a productive environment, but turned out to become the mainstay for language documenters for more than two decades. Shoebox/Toolbox was developed by SIL, who discontinued development in favour of FLEx, an XML based tool.[11] In parallel, ELAN[12] ((Wittenburg et al., 2006)) is another XML-based tool for the representation of correpondences and part-whole relations in glossed texts ((Nordhoff, 2020a)). While XML suggest a good perspectives for programmatic extraction of data, (Nordhoff, 2020a) reports that while syntactically valid XML, the ELAN files retrieved from language archives are semantically wildly heterogeneous, making a principled approach very difficult (also compare (Cimiano et al., 2020, 4)).

On a more theoretical level, (Drude, 2002) proposed a very elaborate model with a multiplicity of tiers. The XML Interlinear Glossed Text (XIGT, (Goodman et al., 2015)) format has a recursive structure instead, allowing for an arbitrary number of tiers ((Xia et al., 2014)). (Chiarcos and Ionov, 2019) and (Ionov, 2021) developed a Linked Data version of XIGT, called LIGT, also used in (Nordhoff, 2020a; Nordhoff, 2020b). For the purposes of this paper, a very simple data model distinguishing the tiers of "utterance" and "word", with respective translations, is sufficient; the level of "morpheme" is disregarded. Basic storage is done in JSON, while transformations into JSON-LD, RDF, and CLDF are also made available. An additional morpheme tier could also have been made available, but it was determined that data consumers could easily create such more granular structures easily themselves should the need arise and that it was not necessary to provide an artificially inflated dataset.

5. Data Sources

Extraction of interlinear examples from documents has a comparatively long history. The ODIN project ((Lewis and Xia, 2010; Xia et al., 2014)) [13] crawled the web for pdfs and tried to extract the examples. Copyright problems and the generally poor extraction facilities, however, posed great challenges for this endeavour. While ODIN is still up and running, it uses meanwhile outdated technology (eg HTML framesets), has encoding issues and does not provide dereferenceable URIs for the examples (Figure 2).

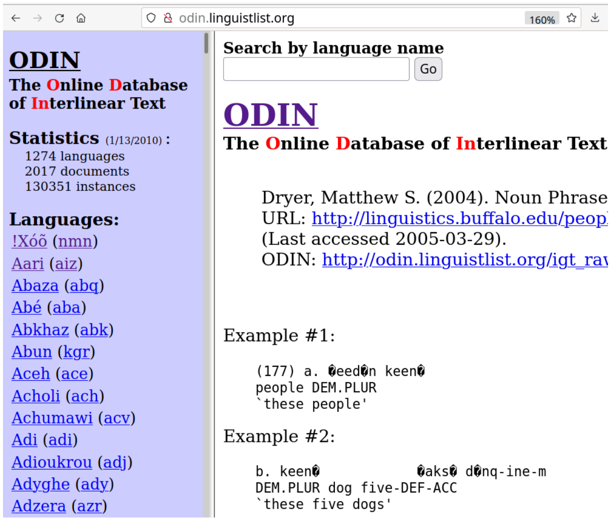
Another source for interlinearized texts are cross-linguistic databases. The Atlas of Pidgin and Creole

[10] `https://software.sil.org/toolbox`

[11] `https://software.sil.org/fieldworks`

[12] `https://archive.mpi.nl/tla/elan`

[13] `http://odin.linguistlist.org`



ODIN
The **O**nline **D**atabase of **In**terlinear Text

Statistics (1/13/2010) :
1274 languages
2017 documents
130351 instances

Languages:
!Xóõ (nmn)
Aari (aiz)
Abaza (abq)
Abé (aba)
Abkhaz (abk)
Abun (kgr)
Aceh (ace)
Acholi (ach)
Achumawi (acv)
Adi (adi)
Adioukrou (adj)
Adyghe (ady)
Adzera (azr)

Search by language name

ODIN
The Online Database of Interlinear Text

Dryer, Matthew S. (2004). Noun Phrase
URL: http://linguistics.buffalo.edu/peop
(Last accessed 2005-03-29).
ODIN: http://odin.linguistlist.org/igt_rav

Example #1:

```
(177) a. �eed�n keen�
people DEM.PLUR
`these people'
```

Example #2:

```
b. keen�             �aks� d�nq-ine-m
DEM.PLUR dog five-DEF-ACC
`these five dogs'
```

Figure 2: A screenshot of the ODIN website, showing an example of the Aari language. Note the URL, which does not give the ID, and the encoding problems. The example given has the “Verified“ rating “highest”. There is also “high”, “auto” and “low”, with presumably worse quality.

Language Structures (APiCS[14], (Michaelis et al., 2013)) offers its example sentences for download in the CLDF format ((Forkel et al., 2018)). These examples were parsed by (Chiarcos and Ionov, 2019), who used them to develop the LIGT format. The APiCS data have the advantage of being available under a free license.

(von Prince and Nordhoff, 2020) and (Nordhoff, 2020a; Nordhoff, 2020b) downloaded data from a variety of language archives, which store ELAN files. ELAN is an XML-format with explicit correspondences between morphemes, words, and sentences. These ELAN files were than converted to the RDF LIGT format, drawing on previous work by (Nordhoff et al., 2016).

Published books, most databases and most of the language archives share the problem of unclear copyright status, which hinders dissemination and reuse. Enter Language Science Press.

6. Language Science Press

Language Science Press is an open-access publisher in linguistics which has published over 180 books (monographs and edited volumes) since 2014. All books are released under a CC-BY license, and the LaTeX source code is available on GitHub. The source code is structured in an identical manner for most books as far as naming conventions and directory structure are concerned, so that a given approach can nicely scale. This is different from, say, the ODIN project or the work on language archives, which had to deal with wildly divergent input data.

For the issue at hand, the task was to retrieve a maximum of interlinear example data from Language Science Press, analyze them, enrich them, and make them available for reuse.

7. Data Handling

7.1. Data Source Identification

For this task, we downloaded the source code of free Language Science Press books. LangSci books have an ID, which corresponds to a GitHub repo. For instance, the book *Attributive constructions in North-Eastern Neo-Aramaic* with the catalog page `https://langsci-press.org/catalog/book/123` has the GitHub repo `https://github.com/langsci/123`.

Not all IDs correspond to published books as some submitted books are rejected. Currently, there are 211 titles listed on the catalog page.

7.2. Data Extraction

The highest current ID is 349, so we iterated through the numbers from 1 to 349 and tried to clone the resulting GitHub address. This yielded 173 repositories with usable tex files. For these repositories, we retrieved 3 033 tex files with a total of 25 020 723 words. The content of these tex files was parsed for examples following the gb4e syntax.

This is illustrated in (1) from (Klamer et al., 2017).

(1) Kamang (Schapper, fieldnotes)
Muut=ak nung iduka.
citrus=DEF PL sweet
‘The citrus fruits are sweet.’

The source code for this example is

```
\langinfo{Kamang}{}{Schapper, fieldnotes} \\
\gll  Muut=ak nung iduka. \\
  citrus=\textsc{def} \textsc{pl} sweet   \\
\glt `The citrus fruits are sweet.'
```

The LaTeX markup like `\langinfo`, `\gll` and `\glt` allow us to meaningfully identify the language name (first line), the source line (starting with `\gll`), the interlinear morpheme translation (following the source line), and the translation (following `\glt`). All examples must have the `\gll` and `\glt` parts; the `\langinfo` part is optional, as is citation information (not shown in the example).

The extraction is complicated by a variety of intervening TeX markup, such as `\textsc{}` for small capitals and similar. The raw data for source line, interlinear line, and translation line have thus to be stripped of their TeX markup. After that, the words of the source and interlinear line can be tokenized and matched. Examples which differ in the number of words between the source line and the interlinear line are discarded. This yields 39,352 vanilla examples with a unified structure.

[14] `https://apics-online.info/`

7.3. Data Linking

For the purposes of this paper, we distinguish a `ligt:Utterance`, which contains a `ligt:WordTier`, which in turn has a number of `ligt:Words`.[15] The relation between those items are given in Figure 3. For a more elaborate representation, see (Chiarcos and Ionov, 2019).

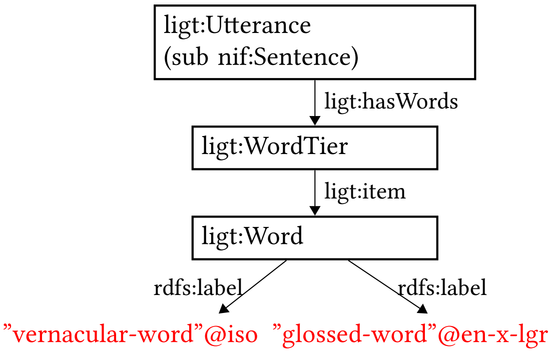


Figure 3: The relevant part of the LIGT model. Note that the predicate `rdfs:label` is assigned twice, but with different language tags. The vernacular label gets the ISO 639-3 code of the language under discussion, while the label containing the glosses gets an RFC 5646 label "en" with a private subtag "-x-lgr" for Leipzig Glossing Rules, following specifications in Section 2.2.7 of the RFC.

We link the extracted examples to Glottolog, the Leipzig Glossing Rules, and Wikidata

7.3.1. Glottolog

Glottolog[16] ((Nordhoff and Hammarström, 2012; Hammarström and Forkel, 2021)) is a knowledge base which contains information about 8,155 languages, some with dialects, and the genealogical classification, amounting to a set of language family trees with more than 25,000 nodes. These nodes have a human readable label (such as "Kamang") and a so-called glottocode with a persistent URL (e.g. `https://glottolog.org/resource/languoid/id/kama1365`). We extracted all language names from the freely available Glottolog dataset ((Hammarström et al., 2021)). and matched the language names we retrieved from our examples to retrieve the corresponding Glottocode. In addition, all examples from books with the title "A grammar of X" were automatically assigned to the language X. If the language name retrieved for a given example could not be matched to a Glottocode in this way, we did a web lookup on glottolog.org with the partial name search. If the result set had the length 1, or if only one result was of type "language" (rather than "dialect" or "family"), this result was retained. Altogether, this yielded 17,425 examples with metadata on source language, for a total of 280 different languages. See Appendix A for a list.

7.3.2. Leipzig Glossing Rules

The Leipzig Glossing Rules are a list of standard abbreviations for grammatical categories such as NOMinative or ACCusative, which are followed by most publications in Diversity Linguistics. We extracted these from the interlinear line and linked them to `https://www.eva.mpg.de/lingua/resources/glossing-rules.php#`.

7.3.3. Named Entity Extraction

There are close to zero NLP tools available for the languages studied in the field of Diversity Linguistics. But fortunately, we have translational equivalents into English for our example sentences. A translation is a faithful rendering of the meaning of a sentence in a given language in another language. Therefore, we can actually use our English translation as a proxy for named entity extraction, as the entities/-concepts should match between the source and the translation. We ran the translation sentences through `https://cloud.science-miner.com/nerd/service/disambiguate`. Upon inspection, a number of the concepts turned out to be misretrievals. For example, translations with "don't" in them are linked to the Wikidata Q17646620, which is about an Ed Sheeran song with the same title. A blacklist was created for these cases.

In a second step, the base concepts were matched with their Wikidata superclasses using the predicates p31 'instanceOf' and p279 'subclassOf'. This allows us to assert that a goat is a mammal is an animal is an organism, greatly enhancing the querying possibilities. This is relevant for instance when linguists want to test hypotheses about certain verbs being sensitive to [±ANIMATE]. Section 7.6 will discuss querying in more detail.

Unfortunately, Wikidata does not provide a very clean ontology. Five problems were discovered:

1. misunderstandings of the predicate subclassOf (sweat > excrement > biodegradable waste > waste > bad)

2. useless use of upper ontologies (all sounds are acoustic waves are elastic waves are mechanical waves are waves are oscillations are changes are occurrences are temporal entities are spatio-temporal entities are entities)

3. conjunct categories ("inflammable solid") which needlessly inflate the category count. The categories "inflammable substance" and "solid substance" would have been sufficient.

4. Eurocentrism (housekeeping activities are "activities of households as employers; undifferenti-

[15] Unfortunately, the PURL for LIGT did not resolve while we were writing this paper, so our resources point to a local copy of the LIGT ontology instead.

[16] `https://glottolog.org`

ated goods- and services-producing activities of households for own use" (Q29584238) as part of the Statistical Classification of Economic Activities in the European Community). This is irrelevant in an African context.

5. Other regiocentrisms (all baked items are Bánh; all dairy produce is part of some Russian classification "dairy products and ice cream, as well as services" (Q27149326)).

A blacklist of nearly 1500 entries had to be created to weed out problems caused by the listed shortcomings. Taking into account this second blacklist, we arrive at an augmented count of 28,777 entity tokens (6,773 types). The most frequent concepts are: food (Q2095, 833 instances), organism (Q7239, 788 instances), animal (Q729, 486 instances).

7.4. Data Storage and representation

The extracted examples (see 7.2) are further processed in two forms. One leads to csvw[17]-based CLDF representation[18], another pipeline feeds the IMTVault search and API available at `https://imtvault.org`.

For IMTVault, extracted examples are transformed into plain JSON as well as expanded JSON-LD 1.1.[19] Following the w3c best practises[20], we chose the expanded representation, as no explicit context reference is needed in downstream processing. Additionally, we use the robust titanium-json-ld library for JSON-based Serialization for Linked Data,[21] which provides sound support for transformation from JSON to JSON-LD 1.1, and from JSON-LD to RDF N-Quads.

The plain JSON representation is also used to create a search index based on elasticsearch, which serves the faceted user interface for search available at `https://imtvault.org/search`.

This allows us to present linguistic examples in a suitable way to query for both humans and machines (Figure 4) using either static, referable snapshots of the collection or dynamically via http based retrieval services.

7.5. Minting / URL Resolution

IMTVault has a built-in URL resolver to refer to books and examples, which can be prompted for various formats. The URL pattern includes two dynamic path elements, the book ID (taken over from Language Science Press) and a generated utterance ID.

Resolving utterances Utterances are identified by book ID and an example ID generated as the hexdigest of hashing the sourceline with SHA-256, truncated to 10 digits. The resolver supports four representations: minimal html (appending .htm to the URL pattern), plain JSON (.json), expanded JSON-LD 1.1 (.jsonld) or RDF N-Quads Dataset 1.1 representation (.nq),[22] leading to `https://imtvault.org/b/80/ex/01-9383b907b9.htm`, `https://imtvault.org/b/80/ex/01-9383b907b9.json`, `https://imtvault.org/b/80/ex/01-9383b907b9.jsonld`, and `https://imtvault.org/b/80/ex/01-9383b907b9.nq`, respectively.

Resolving books Without a file ending, the resolver will redirect to the original publication as landing page of a book at LangSciPress (`https://imtvault.org/b/157`). With the file endings .htm or .ld provided, the resolver will generate a list of all examples found in the respective book. `https://imtvault.org/b/157.json` will thus return a json list of all 99 examples from book 157, *The Alor-Pantar languages: History and typology. Second edition.*

7.6. Data Querying

Query search index The elasticsearch index can be queried programmatically. The following curl command executes a query for 'banana' to the IMTVault index of utterances. If not using an API tool such as postman[23] or insomnia,[24] the XSRF-TOKEN value needs to be obtained beforehand.

```
curl 'https://imtvault.org/express/iss/_search'
  -H 'Cookie: XSRF-TOKEN=XXXX'
  --data-raw '{"query": {
                "multi_match": {"query": "banana"}
               }}'
```

The query can be adapted as required, following the elasticsearch query syntax.[25] For users interested in running their queries locally, the CLDF data can be loaded into a SQLite database providing yet another query platform.

8. FAIR language examples

We applied the best practises known as the FAIR data principles[26] in the implementation of IMTVault. Findability, accessibility, interoperability, and reusability of linguistic resources are achieved to varying degrees:

- F1. (Meta)data are assigned a globally unique and persistent identifier. See the patterns in Section 7.5. The identifiers are unique, and persistent.

[17] (Tennison, 2014)

[18] `https://github.com/langsci/imtvault/tree/main/cldf`

[19] `https://www.w3.org/TR/json-ld11/`

[20] `https://w3c.github.io/json-ld-bp/#use-json`

[21] `https://github.com/filip26/titanium-json-ld`

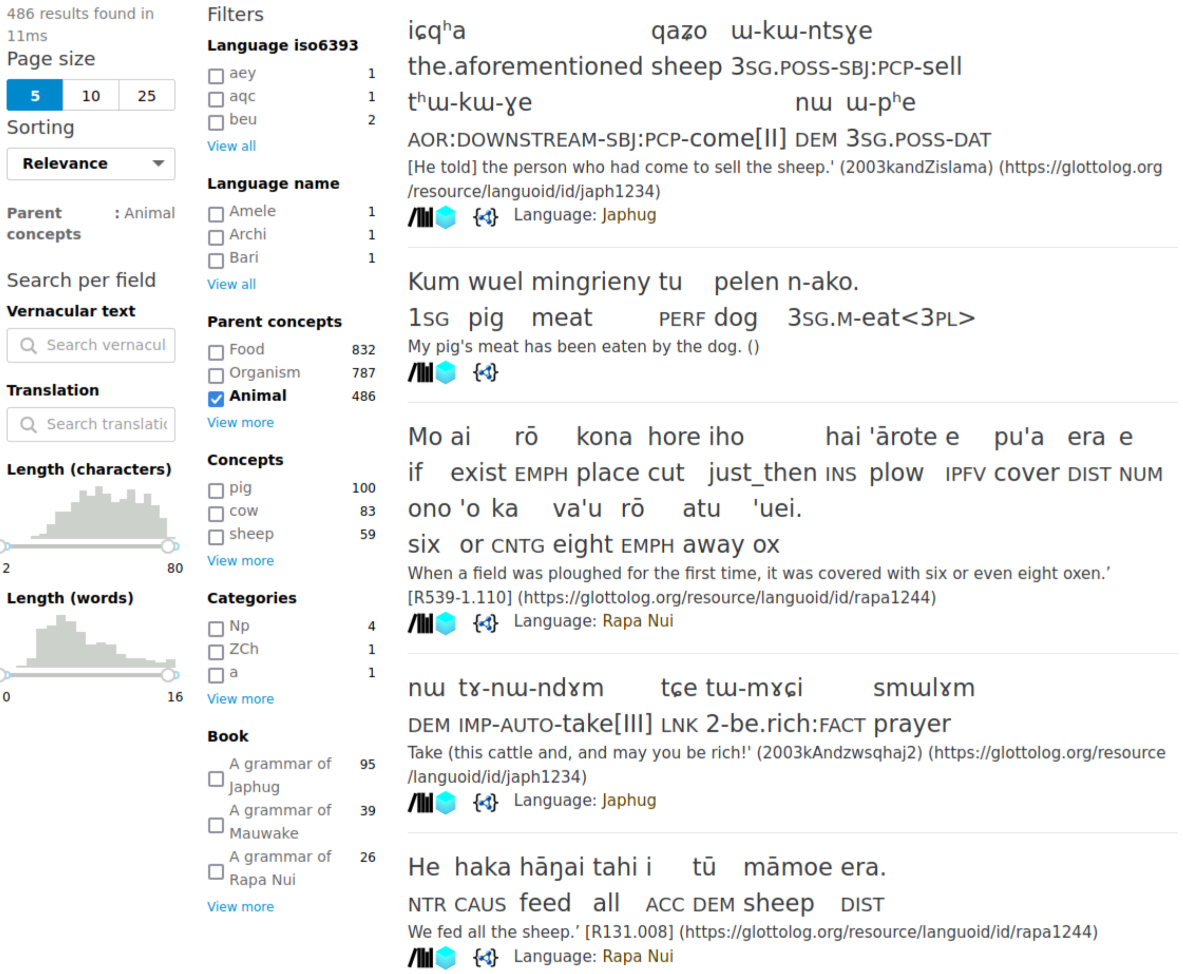
[22] `https://www.w3.org/TR/n-quads/#n-quads-language`

[23] `https://www.postman.com`

[24] `https://insomnia.rest`

[25] `https://www.elastic.co/guide/en/elasticsearch/reference/6.8/full-text-queries.html`

[26] `https://www.go-fair.org/go-fair-initiative/`



486 results found in 11ms

Page size

5 | 10 | 25

Sorting

Relevance ▾

Parent concepts : Animal

Search per field

Vernacular text

Search vernacul

Translation

Search translatic

Length (characters)

2 80

Length (words)

0 16

Filters

Language iso6393

□ aey	1
□ aqc	1
□ beu	2

View all

Language name

□ Amele	1
□ Archi	1
□ Bari	1

View all

Parent concepts

□ Food	832
□ Organism	787
☑ **Animal**	486

View more

Concepts

□ pig	100
□ cow	83
□ sheep	59

View more

Categories

□ Np	4
□ ZCh	1
□ a	1

View more

Book

□ A grammar of Japhug	95
□ A grammar of Mauwake	39
□ A grammar of Rapa Nui	26

View more

iɕqʰa qaʑo ɯ-kɯ-ntsɣe
the.aforementioned sheep 3SG.POSS-SBJ:PCP-sell
tʰɯ-kɯ-ɣe nɯ ɯ-pʰe
AOR:DOWNSTREAM-SBJ:PCP-come[II] DEM 3SG.POSS-DAT
[He told] the person who had come to sell the sheep.' (2003kandZislama) (https://glottolog.org/resource/languoid/id/japh1234)
Language: Japhug

Kum wuel mingrieny tu pelen n-ako.
1SG pig meat PERF dog 3SG.M-eat<3PL>
My pig's meat has been eaten by the dog. ()

Mo ai rō kona hore iho hai 'ārote e pu'a era e
if exist EMPH place cut just_then INS plow IPFV cover DIST NUM
ono 'o ka va'u rō atu 'uei.
six or CNTG eight EMPH away ox
When a field was ploughed for the first time, it was covered with six or even eight oxen.’ [R539-1.110] (https://glottolog.org/resource/languoid/id/rapa1244)
Language: Rapa Nui

nɯ tɤ-nɯ-ndɤm tɕe tɯ-mɤɕi smɯlɤm
DEM IMP-AUTO-take[III] LNK 2-be.rich:FACT prayer
Take (this cattle and, and may you be rich!' (2003kAndzwsqhaj2) (https://glottolog.org/resource/languoid/id/japh1234)
Language: Japhug

He haka hāŋai tahi i tū māmoe era.
NTR CAUS feed all ACC DEM sheep DIST
We fed all the sheep.’ [R131.008] (https://glottolog.org/resource/languoid/id/rapa1244)
Language: Rapa Nui

Figure 4: Querying facilities for humans. The screenshot shows a query for the topic “animal”. The screenshot shows the result list including sentences with interlinear morpheme translation from Japhug and Rapa Nui, covering different kinds of animals (sheep, pig, dog, oxen, cattle).

- F2. Data are described with rich metadata (defined by R1 below): The utterances are described using the relevant metadata schemes and referencing the original publication.
- F3. Metadata clearly and explicitly include the identifier of the data they describe: An identifier for each utterance is generated by IMTVault
- F4. (Meta)data are registered or indexed in a searchable resource : IMTVault provides a user interface for search for humans. The backing index can be queried (Section 7.6).
- A1.1 The protocol is open, free, and universally implementable: HTTP and Elasticsearch/Lucene query language are open standards.
- A1.2 The protocol allows for an authentication and authorisation procedure, where necessary: IMTVault implements authentication and authorisation. While currently all resources are available without restriction, IMTVault could handle embargoes or other types of access control if required.
- A2. Metadata are accessible, even when the data are no longer available : As data are embedded into the metadata, this does not apply.
- I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation. JSON and JSON-LD are W3C recommendations.
- I2. (Meta)data use vocabularies that follow FAIR principles: The vocabularies used (RDF Schema, Dublin Core terms/elements, liodi/ligt) themselves comply with the FAIR principles.
- I3. (Meta)data include qualified references to other (meta)data : We reference Wikidata, Glottolog, and the Leipzig Glossing Rules in a qualified manner.

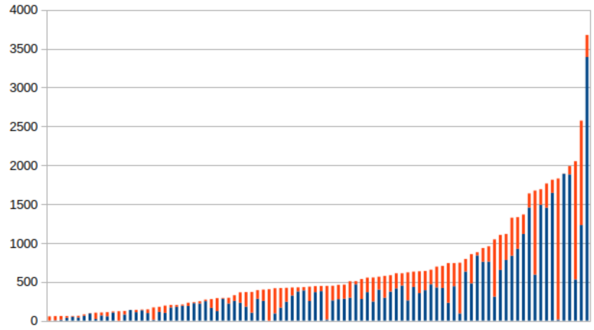


Figure 5: Retrieved examples (blue) vs skipped examples (red) for all books with a quorum of at least 50 examples.

- R1.1. (Meta)data are released with a clear and accessible data usage license : The License is CC BY 4.0 and indicated in both API responses and the user interface for search.
- R1.2. (Meta)data are associated with detailed provenance : The original publication is named and referenced. In addition, the primary citation is given as well if it could be retrieved.
- R1.3. (Meta)data meet domain-relevant community standards, which are Glottolog and LGR in our case.

9. Evaluation

All tex files retrieved from GitHub together contain 60 615 LaTeX commands `\gll` signalling interlinear examples. Of these, 39 352 were retrieved for IMTVault. Figure 5 gives the amount of retrieved and non-retrieved examples per book. The average of examples retrieved per `\gll` passage is 62.91%. If we compute this number per book, we arrive at a median value of 66.46%.

Authors often use `\gll` for certain elements which are not interlinear examples in the strict sense, so there can be good reasons to skip them. We investigated how succesful our algorithm was in sorting the relevant (retain) from the irrelevant (discard) examples. We drew a random sample of 100 passages introduced by `\gll` from the 60 615 and inspected manually whether this passage was correctly/incorrectly retained/discarded as an example. This was done in two steps. At first, a book was drawn at random, then, an example was drawn among the ones present in the book. This was repeated until 100 examples were reached. The reason for this two-tiered approach was that otherwise books with many examples such as *A grammar of Japhung* with over 3500 would have completely dominated the set. For the drawn examples, the pdf, the tex code, and the representation on IMTvault.org were compared. Among the 100 examples drawn, 16 were not good interlinear text and should be discarded. This had been done correctly for all of them. Most often, the reason for this was a missing translation. 84 should have been retained, but this was only the case for 72 of them. 12 were missed, or one in seven. The precision was thus 100% while the recall was 85.7%, giving an aggregate F-score of 91.9%.

Turning to concepts, the sample was extremely sparse. May sentences were of the type *Why read the book?*, which is too short and bland to do meaningful Named Entity Recognition. As such, only 6 concepts were correctly attributed to examples of the sample, while a further 6 were misattributions, often of pop songs with banal titles such as *Tender Years* by George Jones or *Live Life* by the Kinks. We conjecture that concept retrieval might have a very skewed distribution: grammatical descriptions in general have longer and more colourful examples, which are better suited for NER, while more theoretical works tend to have very barren examples, which are boiled down to the minimum, eg *John sees Mary*. If this is the case, we should find more named entities in texts from endangered language archives as well, cf. (Nordhoff, 2020b). Further research will test this hypothesis.

10. Conclusion and Outlook

We started with the observation by (Joshi et al., 2020) that over 90% of the world's languages have no NLP resources. We now provide 40 000 sentences in 280 languages, most of them no/low resource, as a structured dataset under a free license for reuse. The dataset respects the FAIR principles as well as the Linked Data Principles. We have a clearly defined pipeline, a storage format, a query/dissemination platform and consumers downstream. Language Science Press will continue to produce about 30 books a year, but there are other Open Access publishers whose publications could also be crawled to extract interlinear examples. An obvious candidate would be the Diamond-OA journal *Glossa*.[27]

This resources improves on ODIN or the interlinear text extracted from language archives reported in (Nordhoff, 2020a; Nordhoff, 2020b) in that the data are available under an open license and good facilities for querying and dereferencing are in place. As compared to the APiCS set created by (Chiarcos and Ionov, 2019), IMTVault has added about the double the amount of sentences (40k as compared to 18.5k for APiCS) and a more extensive range of formats and querying possibilities.

Integration of the APiCS data by (Chiarcos and Ionov, 2019) is a logical next step, as is the integration of data from endangered language archives ((von Prince and Nordhoff, 2020)), to the extent that the licenses employed there permit this. Further refinement of Named Entity Recognition will be necessary, as well as better algorithms for the identification of the language an example is in based on the surrounding text.

[27] `https://www.glossa-journal.org`

11. Bibliographical References

Chiarcos, C. and Ionov, M. (2019). Ligt: An LLOD-native vocabulary for representing interlinear glossed text as RDF. In Maria Eskevich, et al., editors, *2nd Conference on Language, Data and Knowledge (LDK 2019)*, number 70 in OpenAccess Series in Informatics (OASIcs), pages 3:1–3:15, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Cimiano, P., Chiarcos, C., McCrae, J. P., and Gracia, J. (2020). *Linguistic Linked Data: Representation, Generation and Applications*. Springer, Cham.

Drude, S. (2002). Advanced glossing: A language documentation format and its implementation with Shoebox. In Peter Austin, et al., editors, *Proceedings of the International LREC workshop on Resources and Tools in Field Linguistics*.

Forkel, R., List, J.-M., Greenhill, S. J., Rzymski, C., Bank, S., Cysouw, M., Hammarström, H., Haspelmath, M., Kaiping, G. A., and Gray, R. D. (2018). Cross-linguistic data formats, advancing data sharing and re-use in comparative linguistics. *Scientific Data*, 5:180205.

Goodman, M. W., Crowgey, J., Xia, F., and Bender, E. M. (2015). Xigt: extensible interlinear glossed text for natural language processing. *LREC*, 49(2):455–485.

Hammarström, H. and Forkel, R. (2021). Glottocodes: Identifiers linking families, languages and dialects to comprehensive reference information. *Semantic Web Journal*.

Hammarström, H., Forkel, R., Haspelmath, M., and Bank, S. (2021). *Glottolog 4.5*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Ionov, M. (2021). APiCS-Ligt: Towards semantic enrichment of interlinear glossed text. In Dagmar Gromann, et al., editors, *3rd Conference on Language, Data and Knowledge (LDK 2021)*, volume 93 of *Open Access Series in Informatics (OASIcs)*, pages 27:1–27:8, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Joshi, P., Santy, S., Budhiraja, A., Bali, K., and Choudhury, M. (2020). The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, page 6282–6293.

Klamer, M., Schapper, A., and Corbett, G. (2017). Plural number words in the alor-pantar languages. In Marian Klamer, editor, *The Alor Pantar languages*, number 3 in Studies in Diversity Linguistics, page 365–403. Language Science Press, Berlin.

Lewis, W. D. and Xia, F. (2010). Developing ODIN: A multilingual repository of annotated language data for hundreds of the world's languages. *Journal of Literary and Linguistic Computing (LLC)*, 25(3):303–319.

Susanne Maria Michaelis, et al., editors. (2013). *Atlas of Pidgin and Creole Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. http://apics-online.info.

Nordhoff, S. and Hammarström, H. (2012). Glottolog/langdoc: Increasing the visibility of grey literature for low-density languages. In *Proceedings of LREC 2012*.

Nordhoff, S., Tuttle, S., and Lovick, O. (2016). The Alaskan Athabascan Grammar Database. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, 5. European Language Resources Association (ELRA).

Nordhoff, S., Krämer, T., and Forkel, R. (2022). IMT Vault (v1.0). Data set.

Nordhoff, S. (2020a). From the attic to the cloud: mobilization of endangered language resources with linked data. In *Proceedings of the Workshop about Language Resources for the SSH Cloud*, pages 10–18, Marseille, France, May. European Language Resources Association.

Nordhoff, S. (2020b). Modelling and annotating interlinear glossed text from 280 different endangered languages as linked data with LIGT. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 93–104, Barcelona. Association for Computational Linguistics.

Tennison, J. (2014). a primer, CSV on the web. W3C working group note. Cambridge: W3C.

von Prince, K. and Nordhoff, S. (2020). An empirical evaluation of annotation practices in corpora from language documentation. In *Proceedings of LREC 2020*. LREC, Marseille.

Wittenburg, P., Hennie, B., Russel, A., Klassmann, A., and Sloetjes, H. (2006). *ELAN: A Professional Framework for Multimodality Research*.

Xia, F., Lewis, W. D., Goodman, M. W., Crowgey, J., and Bender, E. M. (2014). Enriching ODIN. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, number 2014.

A. Appendix

This is a list of languages and their glottocodes for which at least one interlinear example could be retrieved.

abui1241	Abui
adan1251	Adang
afri1274	Afrikaans
agua1253	Aguaruna
akan1250	Akan
alam1246	Alamblak
alba1267	Albanian
aleu1260	Aleut
amel1241	Amele
amis1246	Amis
anti1246	Antioch
arab1395	Arabic
arch1244	Archi
assa1263	Assamese
awad1243	Awadhi
awji1241	Awjilah
awtu1239	Awtuw
bamu1253	Bamun
bari1284	Bari
bari1286	Bariai
basq1248	Basque
bath1244	Baṭḥari
bava1246	Bavarian
beja1238	Beja
bena1259	Bena
beng1280	Bengali
berb1260	Berber
bero1242	Berom
bezh1248	Bezhta
bilo1248	Biloxi
bium1280	Biu-Mandara
blag1240	Blagar
bong1285	Bongo
bong1298	Bongor
bora1263	Bora
braj1242	Braj
braz1246	Brazilian Portuguese
budu1265	Buduma
bukh1238	Bukharic
buku1249	Lubukusu
buna1278	Bunaq
bund1253	Bundeli
buru1296	Burushaski
cant1236	Cantonese
capp1239	Pharasiot
cayu1261	Cayuga
cent1972	Central Kurdish
chum1261	Chumburung
coos1249	Hanis Coos
copt1239	Coptic
cusc1236	Cuzco Quechua
cypr1249	Cypriot Greek
dadi1249	Dadiya
dani1285	Danish
dido1241	Tsez
digo1243	Digo
dink1262	Dinka
doma1258	Jerusalem
dutc1256	Dutch
dyir1250	Dyirbal
efik1245	Efik
egyp1253	Egyptian Arabic
elem1253	Eleme
enga1252	Enga
ewee1241	Ewe
faro1244	Faroese
fefe1239	Fe'efe'e
fern1234	Pichi
finn1318	Finnish
fore1270	Fore
fuli1240	Fuliiru
furu1242	Furu
fyam1238	Fyem
gaaa1244	Ga
ghod1238	Godoberi
gida1247	Gidar
gree1276	Greek
guja1252	Old Gujarati
gunn1250	Gungbe
guro1248	Guro
gyel1242	Gyele
hait1244	Haitian Creole
hali1245	Coastal Marind
halk1245	Halkomelem
hass1238	Ḥassāniyya
hind1269	Hindi
hung1274	Hungarian
hunz1247	Hunzib
hupd1244	Hup
icel1247	Icelandic
ikkk1242	Ik
inan1242	Inanwatan
indo1316	Indonesian
indo1319	Indo-European
inui1246	Inuit
iraq1241	Iraqw
ital1282	Italian
itza1241	Itzá
japh1234	Japhug
jara1276	Jarawara
jita1239	Jita
kabw1241	Kabwa
kaby1243	Kabyle
kaer1234	Kaera
kagf1238	Ut-Ma'in
kala1372	Kalasha
kama1365	Kamang
kava1241	Kavalan
kelo1247	Klon
keng1240	Kenga
khez1235	Khezha Naga
khuz1234	Khuzestan
kild1236	Kildin Saami
kili1267	Kilivila
kima1244	Kimaragang
kips1239	Kipsigis
klao1243	Klao
kohu1244	Kohumono
komi1268	Komi-Zyrian
kore1280	Korean
kulu1253	Tibeto-Burman
kuma1276	Nêlêmwa-Nixumwak
kumz1235	Kumzari
kway1241	Kwaya
kwom1262	Kwoma
laca1243	Lacandón
lako1247	Lakota
lamm1241	Western Pantar
late1256	Late Egyptian
lati1261	Latin
latv1249	Latvian
lavu1241	Lavukaleve
lele1264	Lelemi
lese1243	Lese
lezg1247	Lezgian
limb1268	Limbum
loni1238	Loniu
lugb1240	Lugbara
lule1254	Lule Saami
mait1250	Maithili
mako1251	Makonde
mala1464	Malayalam
malt1254	Maltese
mamm1241	Mam
mang1381	Mangarrayi
mang1394	Mangbetu
mani1292	Meithei
maoo1244	Mao
maor1246	Maori
mapu1245	Mapudungun
mauw1238	Mauwake
maya1287	Mayan
mayo1261	Mayogo
mege1234	Megeb
mehr1241	Mehri
meje1239	Meje
mesk1242	Meskwaki
mian1256	Mian
midd1317	Middle English
midd1321	Middle Dutch
midd1343	Middle High German
mina1268	Minangkabau
mira1254	Miraña
misk1235	Mískito
mofu1248	Mofu-Gudur
moha1258	Mohawk
molo1266	Moloko
mopa1243	Mopán Maya
moro1292	Moroccan Arabic
mwan1247	Mwani
nalc1240	Nalca
ndem1249	Ndemli
ndut1239	Ndut-Falor
nezp1238	Nez Perce
noma1260	Nomaande
nort2641	Northern Kurdish
nort2671	North Saami
nort3139	North Levantine Arabic
nort3142	Sason
norw1258	Norwegian
nubi1253	Nubi
nucl1301	Turkish
nucl1302	Georgian
nucl1328	Wambaya
nucl1417	Igbo
nucl1622	Marind
nucl1630	Barai
nupe1254	Nupe-Nupe-Tako
nyan1308	Nyanja
oksa1245	Oksapmin
olde1238	Old English
olde1242	Old Egyptian
oldf1239	Old French
oldj1239	Old Japanese
oldr1238	Old Russian
olds1249	Old Spanish
omah1247	Omaha-Ponca
oman1238	Omani
onei1249	Oneida
oroc1248	Oroch
paam1238	Paamese
papu1250	Papuan Malay
paum1247	Paumari
phal1254	Palula
pipi1250	Pipil
pnar1238	Pnar
polc1243	Polci
poli1260	Polish
rapa1244	Rapanui
rash1249	Rashad
roma1327	Romanian
russ1263	Russian
ruul1235	Ruuli
safa1245	Safaitic
sans1269	Sanskrit
sant1410	Santali
sanz1248	Sanzhi
sara1340	Saramaccan
savo1255	Savosavo
scot1245	Scottish Gaelic
shua1254	Shua
siee1239	Sie
sigi1234	Sigidi
sino1245	Sino-Tibetan
siwi1239	Siwi
skol1241	Skolt Saami
sout2674	South Saami
sout2789	Central Dagaare
sout2969	Southern Paiute
sran1240	Sranan Tongo
stan1288	Spanish
stan1289	Catalan
stan1290	French
stan1295	German
suba1252	Suba-Simbiti
suda1236	Sudanese Arabic
surs1245	Sursilvan-Oberland
swah1253	Swahili
swed1254	Swedish
swis1247	Swiss German
taga1270	Tagalog
tago1246	Tagoi
taji1245	Tajik
tama1365	Tamasheq
tari1263	Tarifiyt
taro1263	Tarok
teiw1235	Teiwa
teop1238	Teop
tian1238	Tianjin Mandarin
toab1237	Toqabaqita
tobe1252	Tobelo
tokp1240	Tok Pisin
toto1304	Totoli
tuar1240	Tuareg
tuka1247	Tukang Besi
udih1248	Udihe
uduk1239	Uduk
unaa1239	Una
uppe1455	Upper Guinea Crioulo
viet1252	Vietnamese
vlaa1240	Western Flemish
waim1252	Waima'a
wapp1239	Wappo
wara1294	Komnzo
waya1269	Wayana
weno1238	Wobé
wers1238	Wersing
yace1238	Yatye
yagu1244	Yagua
yima1243	Yimas
yiwo1237	Yiwom
yong1288	Yongning Na
yoru1245	Yoruba
yura1255	Yurakaré
zand1248	Zande
zena1248	Zenaga