

QQ: A Language Metadata Toolkit for Multilingual NLP

Wessel Poelman¹ and Yiyi Chen² and Miryam de Lhoneux¹

¹L^AGOM-NLP, Department of Computer Science, KU Leuven

²AAU-NLP, Department of Computer Science, Aalborg University

wessel.poeleman@kuleuven.be

Abstract

Multilingual NLP research increasingly involves hundreds or thousands of languages across different datasets. Managing, discovering, and reporting language metadata becomes a common hurdle at these scales. We present QQ, a metadata toolkit and browser explorer. QQ compiles language metadata sources into a graph of language varieties, scripts, regions, identifiers, names, and relations, and exposes it through a Python API, a command-line interface, and a browser-based explorer. Users can normalize identifiers, retrieve metadata, traverse relations, and discover which external resources contain which languages. We demonstrate QQ on three workflows: an audit of the HuggingFace Hub, linking resources that use different identifier systems, and generating language-reporting tables. QQ supports FAIR-oriented metadata practices through versioning, open formats, and reusable interfaces.

 Explorer  Source
> _ PyPI  Demo

1 Introduction

The number of languages considered in multilingual NLP research has increased drastically in recent years (e.g., Adebare et al., 2023; Imani-Goghari et al., 2023). Datasets, benchmarks, and models now regularly cover hundreds or thousands of languages. While good news for the field, this also creates a practical problem: language metadata becomes harder to manage consistently.

Additionally, multilingual models are increasingly evaluated on more tasks and in more languages. As more datasets are combined across training and evaluation, a common problem becomes more apparent: datasets do not stick to one convention when using language identifiers. One dataset might list German as `de`, another as `deu`, and a third as `stan1295`. This is easy to handle when dealing with ten or twenty languages,

but with hundreds or thousands of languages, with several identifier types in active use, it becomes cumbersome at best and unmanageable at worst. This issue is amplified since languages need to be tracked across exploration, data cleanup, analysis, and reporting.

Identifiers are only part of the problem. Language metadata matters *a posteriori* when reporting which languages were used in a study (Bender, 2019), and it matters *a priori* when choosing languages based on scripts, geography, families, or other properties. Currently, managing this often means using local mapping files, scattered scripts, and manual browsing across several metadata sources, such as Wikipedia or Glottolog (Hammarström et al., 2024). As the number of languages grows, one-off scripts become increasingly brittle: the same identifiers must be resolved repeatedly across different resources and experimental steps.

Coming back to our German example: OPUS (Tiedemann, 2012) and Universal Dependencies (Nivre et al., 2020) use the ISO 639-1 code `de`; WMT uses the ISO 639-3 code `deu`; older datasets might use ISO 639-2 `ger`; some datasets, such as NLLB (Costa-jussà et al., 2024) and FineWeb-2 (Penedo et al., 2025), use tags such as `deu_Latn`; linguistic datasets often use the Glottocode `stan1295` (Forkel and Hammarström, 2022); and Wikidata uses `Q188`. These different identifiers all refer to the same language.

The same problem appears when trying to find datasets for a particular language. On the HuggingFace Hub (Lhoest et al., 2021), the language filter shows separate entries for what is effectively the same language, such as `el` and `ell` for Modern Greek. Some commonly-used linguistic resources, like WALS (Dryer and Haspelmath, 2013), invent their own identifiers that do not belong to any other standard (`grk` for Modern Greek). This makes the discoverability of which datasets contain which languages more difficult than it should be. This issue

is only getting worse given the increasing number of languages and datasets in multilingual NLP research. Routine tasks like finding datasets for a language, counting language coverage, or combining datasets should be quick and convenient.

To address these issues, we introduce QQ. **QwanQwa** is the phonetic spelling of ቋንቋ, meaning *language* in Amharic. QQ compiles language metadata sources into a versioned, unified database. This database serves as a shared backend for three complementary interfaces: a Python library for simple, programmatic *management*, a command-line interface for simple, *reproducible* workflows, and a browser explorer for interactive *discovery*. QQ resolves identifiers, merges sources, and allows the same data to be reused in scripts, reports, and discovery. Our contributions are:

- We introduce QQ, a language metadata toolkit and browser explorer for multilingual NLP.
- QQ integrates open language metadata resources into one graph over language varieties, scripts, regions, families, identifiers, names, speaker estimates, endangerment statuses, dataset availability, and more.
- We validate QQ on practical workflows: large-scale language-tag auditing, resource linking, and language reporting.
- QQ supports FAIR-oriented metadata practices (*Findable, Accessible, Interoperable, Reusable*; [Wilkinson et al., 2016](#)) through versioning, open formats, and reusable interfaces.

2 QQ

The core of QQ is a compiled database with several interfaces on top. During database construction, QQ imports open language metadata sources, normalizes and resolves their identifiers, and merges partially overlapping records into *a unified metadata graph*. The compiled database ships with the Python package, so common workflows work immediately after installation without downloading or querying each source separately. This also ensures proper versioning and reproducibility. The same snapshot is then used by the Python API, command-line interface, and browser explorer.

QQ represents metadata as a graph consisting of `Languoid`, `Script`, and `Region` nodes. We use “languoid” as a neutral term for language-like entities covering languages, dialects, macrolanguages, and families ([Good and Hendryx-Parker,](#)

2006). Identifiers do not always refer only to individual languages: ISO 639-3 codes and Glotocodes can refer to all four, whereas ISO 639-5 codes strictly refer to families, for example. Edges in the graph connect languoids to scripts, regions, and other languoids. Other metadata such as level (language, dialect, family), identifiers, names, speaker counts, endangerment, external links, and more are stored as node or edge properties.

QQ focuses on sources useful for multilingual NLP workflows. It currently integrates `LinguaMeta` ([Ritchie et al., 2024](#)), `Glottolog` ([Hammarström et al., 2024](#)), `Glotscript` ([Kargaran et al., 2024](#)), Wikipedia and Wikidata metadata, ISO data from the Library of Congress,¹ script information from Unicode,² and deprecated-code information from SIL³ and IANA.⁴ These sources provide overlapping but different information: identifiers, language names, speaker counts, family relations, scripts, regions, Wikipedia editions, and deprecated or replacement identifier codes. The merge step uses shared identifiers to connect partially overlapping source records, as illustrated in [Figure 1](#). After merging, QQ can resolve all identifiers in the connected record to the same languoid. These resources are the *primary* sources for the graph; QQ also supports linking to *external* sources (as seen in [Figure 2](#)). This allows for easy discoverability of resources linked to a particular language, regardless of the identifier convention it uses. Full source details are available in §A.

In Python, QQ supports many common operations through a compact interface:⁵

```
from qq import Database, IdType
db = Database.load()

d1 = db.get("nl")
d1.name
> "Dutch"
d1.iso_639_3
> "nld"
d1.speaker_count
> 24085200
d1.parent.parent
> Languoid("Modern Dutch")

d3 = db.guess("dutc1256")
d2 = db.guess("nld_Latn")
d4 = db.guess("nl-Latn-NL")
d1 == d2 == d3 == d4 # True
```

¹loc.gov/librarians/standards

²unicode.org/ucd

³iso639-3.sil.org

⁴iana.org/protocols

⁵See the [README](#) for much more functionality.

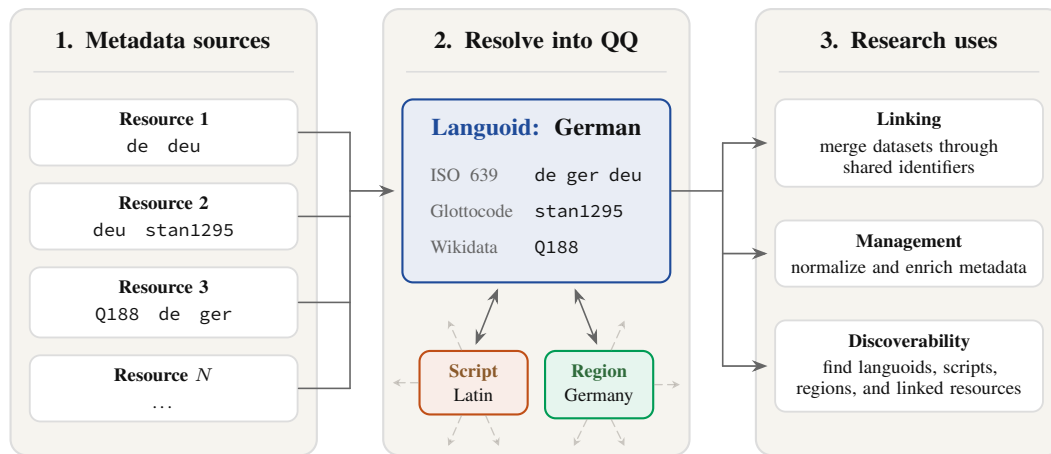


Figure 1 – QQ links metadata sources through shared identifiers. Once records are resolved to the same languoid, the merged representation supports normalization, linking, traversal, reporting, and resource discovery.

Three design goals were especially important. First, QQ ships a packaged snapshot,⁶ so common workflows work immediately after installation. Second, identifier resolving is the central operation, which means the Python API, CLI, browser explorer all use the same database. Third, the graph structure: multilingual NLP workflows often involve multi-hop traversal: from a language to a region, from that region to other languages, and from languages to scripts or linked resources. QQ’s graph makes these relations explicit and accessible.

Interfaces. The interfaces cover different points in a typical workflow. The Python API is meant for analysis scripts, where identifiers need to be normalized before counting, filtering, or joining datasets. The CLI exposes the same database for shell workflows (such as `qq get <id>`). Sources can also be updated through the CLI with a single command. The browser explorer is meant for discoverability: users can search a name or code, check the identifiers and metadata attached to the resulting languoid, explore neighboring nodes in the graph, and find external resources.

Resource discovery and reproducibility. Languoids, scripts, and regions can be searched through identifiers and names (both *exonyms* and *endonyms*). As mentioned, external resources can be attached to a languoid, which allows for easy discovery in external resources. In the current database version, languoids are linked to 7,900+ HuggingFace pages, 6,300+ English Wikipedia articles about a particular languoid, 2,500+ WALS, 2,400+ Grambank, and 2,100+ PHOIBLE links for

typological information. New resources can easily be added through providing where a particular identifier is located and what the URL format is.

With QQ, we adhere to and facilitate the FAIR principles⁷ of data management (Wilkinson et al., 2016). The language metadata in QQ is:

- *Findable.* Each release of QQ is deposited on Zenodo under a persistent DOI,⁸ and its entries are indexed and queryable both through the browser explorer and programmatically via the API through IDs and metadata.
- *Accessible.* QQ is retrievable through standard channels, such as by installation via `pip` from PyPI, cloning via GitHub, or downloading from Zenodo. Because the database ships as a self-contained snapshot, it remains fully accessible offline and is unaffected by outages of upstream sources.
- *Interoperable.* QQ serializes its database as compressed JSON with a typed schema. Each entry cross-references identifier schemes, e.g., ISO 639-3, Glottocodes, BCP-47 tags, and Wikidata QIDs, so QQ functions as an interoperability layer across otherwise disconnected identifier systems.
- *Reusable.* QQ is designed for reuse across research workflows (discovery, normalization, and management) through a single unified interface, replacing the ad hoc pattern of consulting multiple sources and writing one-off scripts at each step.

⁶The database is ~3 MiB; lazy-loaded name data ~8 MiB.

⁷go-fair.org/fair-principles/

⁸doi.org/10.5281/zenodo.21219618

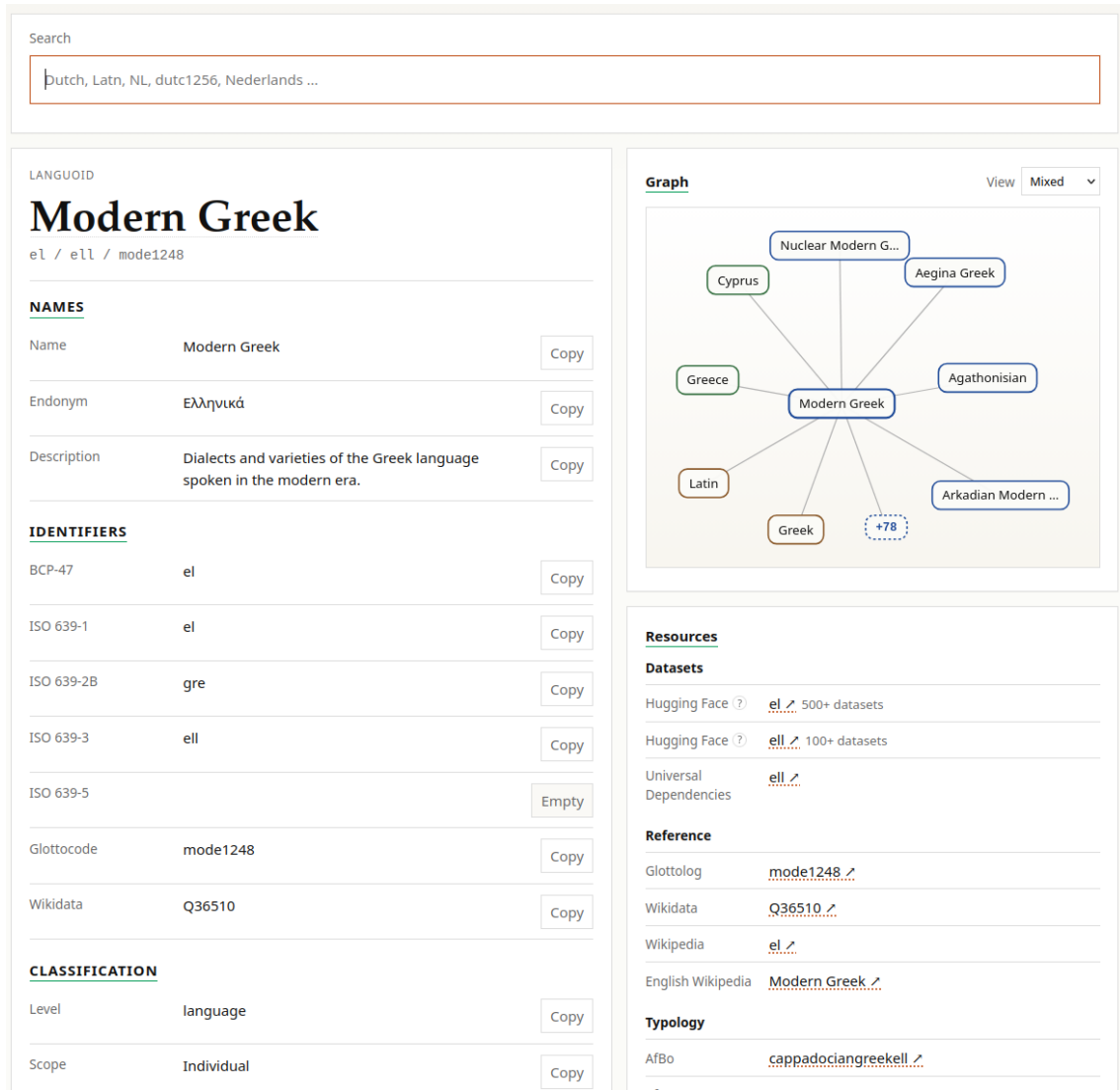


Figure 2 – QQ browser explorer showing identifiers, metadata, graph navigation, and resources for Modern Greek.

Licensing. The QQ software is released under Apache 2.0, and the packaged database under CC BY-SA 4.0. This separation is important since Creative Commons actively discourages code to be licensed under any CC license.⁹ Our compiled database consists of the *primary* sources, as listed in §A. *External* resources remain under their own licenses; QQ only links to external resources, it does not incorporate their contents.

3 Workflow Evaluation

Existing tools address parts of working with language metadata, such as tag parsing, looking up standards or resource curation (§4). To our knowledge, QQ is the first toolkit that specifically targets common metadata issues encountered in multilingual NLP. Therefore, we evaluate QQ through rep-

resentative workflows, since no benchmark exists for such tasks. Each workflow states the problem, shows QQ’s use, and the result.¹⁰

3.1 Auditing HuggingFace Language Tags

Problem. Suppose we want to count or inspect language coverage in datasets on the HuggingFace Hub (Lhoest et al., 2021). Dataset cards can include `language:` tags, but these tags mix identifier conventions. This means we will encounter the same languoid under different tags, including deprecated or non-standard ones.

Using QQ. We collected dataset metadata from the Hub API and QQ classified each unique tag as valid, deprecated,¹¹ a country code used as a language code, or unknown. In June 2026, the Hub API returned 1,051,735 datasets in total, of which

⁹See the CC FAQ and Lambert and Hartgerink (2024).

¹⁰Scripts: github.com/WPoelman/qwanqwa/case-studies

¹¹Deprecated codes are resolved if a replacement is known.

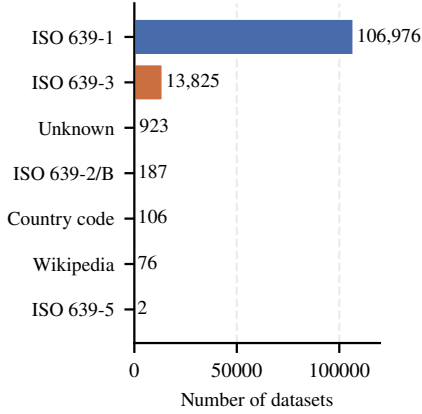


Figure 3 – Identifier types observed in HuggingFace datasets after QQ classification.

Issue	Tags	Examples
Deprecated	34	eml, ajp, iw
Country code	26	gr, cz, jp
Unknown	50	multilingual, code, qaa

Table 1 – Issues found in the HF Hub audit.

119,583 had at least one `language:` tag. Of the 8,237 unique tags, QQ resolves 8,159 tags (99%) to 7,943 unique languoids. Table 1 lists identifier issues; Figure 3 shows which identifiers are in use. **Result.** In the absence of QQ, it seems there are 8,159 unique languages on the Hub. With proper resolving, we see this includes more than 200 duplicates. Similarly, without QQ, normalization is often handled through local mapping files and manual verification. QQ provides a systematic alternative by resolving equivalent codes to a single languoid, which helps when counting, filtering, or comparing datasets. Previous analyses about language resources, like in Joshi et al. (2020) or van Esch et al. (2024), benefit from proper normalization to accurately assess language-coverage. We provide a small, additional analysis about the HF Hub in §A, Figure 5.

3.2 Linking Resources

Problem. A second recurring task is linking resources. Suppose we are evaluating on many different datasets that each use a different identifier convention. Combining these resources is required in evaluation, reporting, exploration, etc. We use lexical resources as an example because they are a good stress test: they describe overlapping sets of languoids using different identifier conventions. We link four resources used in lexical and concept-based analysis: Concepticon Babel-

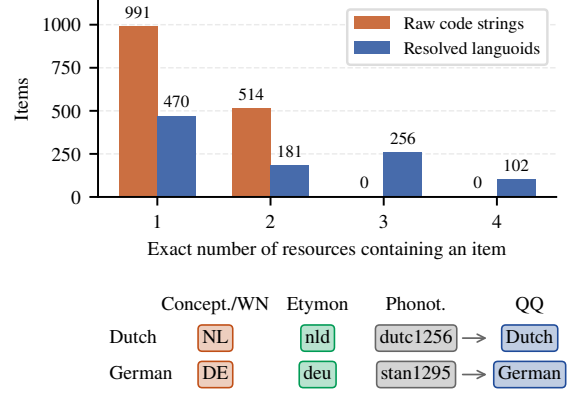


Figure 4 – Linking datasets through resolved languoids.

Net synsets (Navigli and Ponzetto, 2010; List et al., 2016, 2025), WordNet synsets (Miller, 1994), Etymon (de Melo, 2014), and Phonotacticon (Joo and Hsu, 2025). These resources use different identifiers, including ISO 639-2 codes, ISO 639-3 codes, ISO 639-5 family codes, Glottocodes, and a mixture of invalid codes. Just using exact identifiers hides the overlap between the datasets. Figure 4 shows the overlapping sets between the datasets.

Using QQ. The normalization step of QQ resolves different identifiers to a single languoid. In the current snapshot, QQ resolves 592/600 Concepticon codes, 514/520 WordNet codes, 396/397 Etymon codes, and 517/519 Phonotacticon codes. Unresolved codes do not belong to any standard. After resolving, 102 languoids are covered by all four resources.

Result. After resolving, datasets can easily be linked through their shared QQ languoids. This takes away a common hurdle of different identifiers and allows for better interoperability. Reporting which languages were actually used is also straightforward. QQ reduces dataset-specific cleanup and makes the process less error-prone than manual cross-referencing.

3.3 Reporting and Resource Discovery

Problem. A third recurring task is reporting which languages were used in a study. This happens when writing a paper about multilingual NLP, preparing a benchmark, or sanity-checking a language sample before analysis. The work usually combines normalization, metadata lookup, and manual browsing, often with a few intermediate *ad hoc* corrections.

Using QQ. We use QQ to generate a typical LaTeX table containing benchmark results; see Table 4. The output includes relevant metadata such

Tool / resource	ID norm.	Retired IDs	Metadata	Relations	Ext. Links	Python	CLI	Web	Snapshot	Role
Ethnologue	?	?	✓	✓	✗	✗	✗	✓	?	Closed catalogue.
Glottolog	○	✗	✓	✓	✗	○	✗	✓	✓	Classification.
LinguaMeta	○	○	✓	○	✗	✗	✗	✗	✓	Curated metadata.
Glotscript	✗	✗	○	✗	✗	✓	✗	✗	✓	Scripts.
langcodes	✓	✓	○	✗	✗	✓	✗	✗	✓	BCP-47 tags.
pycountry	○	○	○	✗	✗	✓	✗	✗	✓	ISO lists.
DistaLs	✗	✗	○	✗	✗	✓	✓	✓	✓	Distances.
CLDF	✗	✗	○	○	✗	○	✗	○	○	Interchange format.
QQ	✓	✓	✓	✓	✓	✓	✓	✓	✓	Metadata management.

Table 2 – Related tools cover adjacent pieces of the problem; QQ provides easy metadata management and discovery for multilingual NLP. ✓ = yes; ✗ = no; ○ = partially supported; ? = unsure.

as names, BCP-47 and ISO 639-3 codes, canonical scripts, speaker counts, and language families. We have used this workflow in multilingual NLP studies: [Poelman et al. \(2025\)](#) analyze a large number of experimental settings and conditions, each with different, but overlapping sets of languages and datasets; QQ made this process much easier. We also used QQ to generate tables with consistent English names for the languages used. In [Lent \(2025\)](#), QQ is used to find language coverage over datasets using different identifiers. And in [Accou and Poelman \(2026\)](#), QQ is used to generate results and language coverage tables for hundreds of languages across tasks.

Result. Reporting which languages are used becomes much easier with QQ. Instead of manually consulting Wikipedia, Glottolog, and dataset-specific documentation for metadata, researchers can use QQ to inspect, verify, and report languages. With QQ, we have a single, accessible tool that can be used throughout the entire research workflow.

4 Related Tools

[Table 2](#) summarizes how QQ relates to existing tools and resources. QQ is not meant as a replacement for them; in fact, QQ is interoperable with them. QQ facilitates more direct interaction with these resources through easier access, discoverability, and tooling.

QQ addresses a different problem from the sources it builds on. Glottolog ([Hammarström et al., 2024](#)) provides language classifications, phylogenetic data, and Glottocodes ([Forkel and Hammarström, 2022](#)). LinguaMeta ([van Esch et al., 2022](#); [Ritchie et al., 2024](#)) provides curated language metadata, such as endangerment, locales, identifiers, and more for living languages. It is primarily a reference dataset, consisting of 7,000+

JSON files, not a toolkit. Glotscript ([Kargaran et al., 2024](#)) is a system for writing system identification; we use its detailed script metadata. QQ uses these resources as inputs and integrates their fields into shared Languoid, Script, and Region entities.

Ethnologue¹² is a metadata source we do not include since it has a proprietary license.

Two Python libraries have overlapping features: `langcodes`¹³ validates and canonicalizes BCP-47 language tags, and `pycountry`¹⁴ exposes ISO 639 standards. QQ complements these by resolving identifiers across standards to shared Languoids and attaching metadata, relations, and resources.

Some more specialized tools cover related functions: DistaLs ([van der Goot et al., 2025](#)) is a tool for calculating *distances* between languages (defined as ISO 639-3 codes). QQ is interoperable with DistaLs: identifier normalization can be done through QQ and the resulting ISO 639-3 codes can be used to calculate distances. Lastly, CLDF resources (*Cross-Linguistic Data Formats*; [Forkel et al., 2018](#)) provides an interchange format for linguistic datasets. QQ incorporates datasets that use the CLDF format like WALS, PHOIBLE, and Grambank. It is therefore again interoperable; CLDF provides a format for data curation, whereas QQ can be used for interacting with this data.

5 Conclusion

We present QQ, a language metadata toolkit for multilingual NLP. QQ merges open sources into one compiled database and makes the same resolved records available in Python, the CLI, and the browser explorer. The workflows in this paper show how QQ can be used to audit language identifier usage across datasets, link resources with in-

¹²ethnologue.com

¹³github.com/rspeer/langcodes

¹⁴github.com/pycountry/pycountry

compatible identifiers, assist in regular multilingual NLP workflows, and discover related languages by family, script, region, dataset inclusion, and more. QQ offers practical solutions for common metadata issues in multilingual NLP across identifier resolution, metadata lookup, resource discovery, and reporting.

Limitations

QQ depends on the sources it integrates. If a source is missing information or contains an error, QQ may inherit that issue. Merging sources can also introduce ambiguity when sources disagree. In practice, this is rare for essential fields: we encountered two conflicting Glottocodes, which were resolved manually. QQ provides different ways of dealing with such conflicts, see §B.

QQ represents a snapshot of language metadata. It includes historical entities and deprecated identifiers when they are present in the sources, but it does not model how each metadata field changes over time. Time-aware modeling of names, scripts, regions, and political boundaries would require a different data model and would go more in the direction of language-change research, rather than metadata management.

The HuggingFace audit uses dataset-card metadata returned by the Hub API. It does not inspect dataset contents or language columns inside individual datasets. The results should therefore be interpreted as an audit of Hub metadata, not a complete audit of every dataset.

The workflow evaluations are representative rather than exhaustive. They show how QQ supports multilingual NLP metadata tasks, but they do not cover every interface, resource type, or downstream use case.

Acknowledgments

In alphabetical order, we thank Stef Accou, Thomas Bauwens, Artur Kulmizev, Esther Ploeger, and Kushal Tatariya for valuable suggestions and for enduring bugs in the development phase. We also thank Kaja Dobrovoljc, Daan van Esch, Amanda Kann, Marie-Catherine de Marneffe, Tanja Samardžić, and Agata Savary for their insightful feedback.

This work received support from the CA21167 COST action UniDive, funded by COST (European Cooperation in Science and Technology). W.P. is funded by a KU Leuven Bijzonder Onderzoeks-

fonds C1 project with reference C14/23/096. Y.C. is funded by the Carlsberg Foundation (project no. CF21-0454) and the Novo Nordisk Foundation (grant no. NNF24OC0092972).

Use of AI

We used AI assistance for creating the web interface, for reviewing “human-generated” code for the main library, and for checking the grammar and readability of the (human-generated) paper. AI was not used to generate ideas or create other substantial content in this paper. All the content has been verified and reviewed by the authors, who also take full responsibility.

References

- Stef Accou and Wessel Poelman. 2026. [TIPA: Typologically Informed Parameter Aggregation](#). In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 2253–2267.
- Ife Adebbara, AbdelRahim Elmadany, Muhammad Abdul-Mageed, and Alcides Alcoba Inciarte. 2023. [SERENGETI: Massively Multilingual Language Models for Africa](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1498–1537.
- Emily Bender. 2019. [The #BenderRule: On Naming the Languages We Study and Why It Matters](#). *The Gradient*.
- Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Mailard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, and 1 other. 2024. [Scaling neural machine translation to 200 languages](#). *Nature*, 630(8018):841–846.
- Gerard de Melo. 2014. [Etymological Wordnet: Tracing the history of words](#). In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 1148–1154.
- Matthew S. Dryer and Martin Haspelmath, editors. 2013. [WALS Online \(V2020.3\)](#). Zenodo.
- Robert Forkel and Harald Hammarström. 2022. [Glottocodes: Identifiers linking families, languages and dialects to comprehensive reference information](#). *Semantic Web*, 13(6):917–924.
- Robert Forkel, Johann-Mattis List, Simon J. Greenhill, Christoph Rzymski, Sebastian Bank, Michael Cysouw, Harald Hammarström, Martin Haspelmath, Gereon A. Kaiping, and Russell D. Gray. 2018. [Cross-Linguistic Data Formats, advancing data sharing and re-use in comparative linguistics](#). *Scientific Data*, 5(1):180205.

- Jeff Good and Calvin Hendryx-Parker. 2006. [Modeling contested categorization in linguistic databases](#). In *EMELD Workshop 2006: Digital Language Documentation; Tools and Standards; The State of Art*.
- Harald Hammarström, Robert Forkel, Martin Haspelmath, and Sebastian Bank. 2024. [Glottolog 5.0](#). Max Planck Institute for Evolutionary Anthropology.
- Ayyoob ImaniGooghari, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. 2023. [Glot500: Scaling Multilingual Corpora and Language Models to 500 Languages](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117.
- Ian Joo and Yu-Yin Hsu. 2025. [Phonotacticon: A cross-linguistic phonotactic database](#). *Linguistic Typology*, 29(2):405–431.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 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*, pages 6282–6293.
- Amir Hossein Kargaran, François Yvon, and Hinrich Schütze. 2024. [GlotScript: A Resource and Tool for Low Resource Writing System Identification](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 7774–7784.
- Bernd Kortmann, Kerstin Lunkenheimer, and Katharina Ehret, editors. 2020. [eWAVE](#). Max Planck Institute for the Science of Human History.
- Joshua Lambert and Chris Hartgerink. 2024. [Dual licensing R packages with code and data](#).
- Heather Lent. 2025. [Beyond Weaponization: NLP Security for Medium and Lower-Resourced Languages in Their Own Right](#). *Preprint*, arXiv:2507.03473.
- Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, and 1 other. 2021. [Datasets: A Community Library for Natural Language Processing](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184.
- Johann-Mattis List, Michael Cysouw, and Robert Forkel. 2016. [Concepticon: A Resource for the Linking of Concept Lists](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 2393–2400.
- Johann Mattis List, Annika Tjuka, Frederic Blum, Alžběta Kučerová, Carlos Barrientos Ugarte, Christoph Rzymiski, Simon Greenhill, and Robert Forkel, editors. 2025. [CLLD Concepticon 3.4.0](#). Max Planck Institute for Evolutionary Anthropology.
- Susanne Maria Michaelis, Philippe Maurer, Martin Haspelmath, and Magnus Huber, editors. 2013. [APiCS Online](#). Max Planck Institute for Evolutionary Anthropology.
- George A. Miller. 1994. [WordNet: A Lexical Database for English](#). In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 8-11, 1994*.
- Steven Moran and Daniel McCloy, editors. 2019. [Phoible 2.0](#). Max Planck Institute for the Science of Human History.
- Pieter Muysken, Harald Hammarström, Neele Müller Olga Krasnoukhova, Simon van de Kerke Joshua Birchall, Loretta O’Connor, Swintha Danielsen, Rik van Gijn, and George Saad. 2016. [South American Indigenous Language Structures \(SAILS\)](#). Max Planck Institute for the Science of Human History.
- Roberto Navigli and Simone Paolo Ponzetto. 2010. [BabelNet: Building a Very Large Multilingual Semantic Network](#). In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, Daniel Zeman, and 8 others. 2020. [Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043.
- Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. 2025. [FineWeb2: One Pipeline to Scale Them All — Adapting Pre-Training Data Processing to Every Language](#). In *Second Conference on Language Modeling*.
- Wessel Poelman, Thomas Bauwens, and Miryam de Lhoneux. 2025. [Confounding Factors in Relating Model Performance to Morphology](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7273–7298.
- Sandy Ritchie, Daan van Esch, Uche Okonkwo, Shikhar Vashishth, and Emily Drummond. 2024. [LinguaMeta: Unified Metadata for Thousands of Languages](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 10530–10538.
- Frank Seifart and Francesco Gardani, editors. 2026. [AfBo 2.0: A World-Wide Survey of Affix Borrowing](#). Zenodo.

- Hedvig Skirgård, Hannah J. Haynie, Damián E. Blasi, Harald Hammarström, Jeremy Collins, Jay J. Latache, Jakob Lesage, Tobias Weber, Alena Witzlack-Makarevich, Sam Passmore, Angela Chira, Luke Maurits, Russell Dinnage, Michael Dunn, Ger Reesink, and 1 other. 2023. [Grambank reveals the importance of genealogical constraints on linguistic diversity and highlights the impact of language loss](#). *Science Advances*, 9(16):eadg6175.
- Jörg Tiedemann. 2012. [Parallel Data, Tools and Interfaces in OPUS](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 2214–2218.
- Rob van der Goot, Esther Ploeger, Verena Blaschke, and Tanja Samardzic. 2025. [DistaLs: A Comprehensive Collection of Language Distance Measures](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 307–318.
- Daan van Esch, Tamar Lucassen, Sebastian Ruder, Isaac Caswell, Clara Rivera, others, and 1 other, and 1 other. 2022. [Writing System and Speaker Metadata for 2,800+ Language Varieties](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5035–5046.
- Daan van Esch, Sandy Ritchie, Sebastian Ruder, Julia Kreutzer, Clara Rivera, Ishank Saxena, and Isaac Caswell. 2024. [Connecting Language Technologies with Rich, Diverse Data Sources Covering Thousands of Languages](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 3729–3746.
- Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, and 1 other. 2016. [The FAIR Guiding Principles for scientific data management and stewardship](#). *Scientific Data*, 3(1):160018.

A Additional Figures

Resource	QQ use	License
Primary Resources		
Glottolog (Hammarström et al., 2024)	Classification, genealogy, Glottocodes.	CC BY 4.0
LinguaMeta (Ritchie et al., 2024)	Names, speakers, endangerment, scripts.	CC BY-SA 4.0
Glotscript (Kargaran et al., 2024)	Script metadata.	CC BY-SA 4.0
SIL ISO 639-3	ISO 639-3 identifiers and retired codes.	Custom free use with attribution
Library of Congress	ISO 639-1 and ISO 639-2 mappings.	LOC public data / terms
IANA	BCP-47, script, region, and deprecations.	Public / Internet Standard
Wikidata ISO 639-5	ISO 639-5 family links.	CC0
Wikidata EN sitelinks	English Wikipedia sitelinks.	CC0
Wikistats	Wikipedia edition statistics.	CC BY-SA 4.0
Unicode Character DB	Script aliases, code-point ranges.	Unicode License v3
External Resources		
HF Datasets (Lhoest et al., 2021)	Search links; audit.	Mixed; Hub terms
Glottolog (Hammarström et al., 2024)	Reference links.	CC BY 4.0
Wikidata	Reference links.	CC0
English Wikipedia	Article links.	CC BY-SA 4.0
WALS (Dryer and Haspelmath, 2013)	Typology links.	CC BY 4.0
Grambank (Skirgård et al., 2023)	Typology links.	CC BY 4.0
PHOIBLE (Moran and McCloy, 2019)	Phonology links.	CC BY-SA 3.0
Universal Dependencies (Nivre et al., 2020)	Treebank links.	UD v2.18 License Agreement
AfBo (Seifart and Gardani, 2026)	Resource links.	CC BY 4.0
APiCS (Michaelis et al., 2013)	Resource links.	CC BY 3.0
eWAVE (Kortmann et al., 2020)	Resource links.	CC BY 3.0
SAILS (Muysken et al., 2016)	Resource links.	CC BY-NC-ND 2.0 DE
CLLD Meta	Resource links.	CC BY 4.0

Table 3 – Primary and external sources used in the construction of QQ. Names are clickable links.

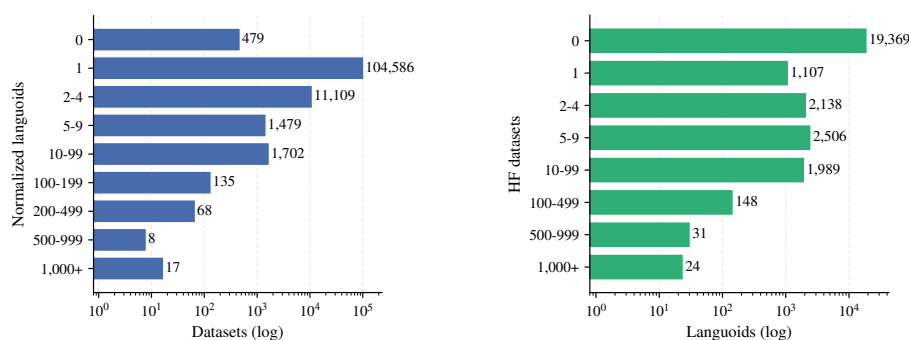


Figure 5 – HuggingFace languoid and dataset coverage. The left plot shows, for each HuggingFace dataset with language metadata, how many unique QQ languoids it contains. Most datasets are monolingual; only a couple are “massively multilingual.” The right plot shows, for each QQ languoid, how many HuggingFace datasets contain it. Many languoids appear in only a handful of datasets. Put differently: most datasets cover very few languages, most languages appear in very few datasets or none, and a small number of languages and datasets dominate the multilingual landscape.

B Merging

Since QQ combines many different sources into a single format, we have to choose what to do when sources provide conflicting information. The most important fields to resolve correctly are the different identifiers. Luckily, in all the primary sources we merge, we only encountered two conflicting identifiers (a duplicate Glottocode in one of the sources), which was easily resolved.

For other fields, QQ provides two options: source priority or manual resolving. With priority, each field can specify its preferred source in case of a conflict. If source *A* with priority 1 lists the English name for the Han̄ script as *Han* and source *B* with priority 2 as *Han (Traditional variant)*, we select source *A*. With manual selection, the user gets prompted in the command line with all conflicting options for a single field, and can pick which one should be kept. The results of this operation are stored, so they can be re-used for the next database rebuild.

Language	BCP-47	ISO 639-3	Script	Speakers	Family
English	en	eng	Latn	1.3B	Indo-European
French	fr	fra	Latn	203.2M	Indo-European
German	de	deu	Latn	91.7M	Indo-European
Spanish	es	spa	Latn	468.4M	Indo-European
Portuguese	pt	por	Latn	225.8M	Indo-European
Italian	it	ita	Latn	65.3M	Indo-European
Dutch	nl	nld	Latn	24.1M	Indo-European
Russian	ru	rus	Cyrl	170.2M	Indo-European
Mandarin Chinese	cmn	cmn	Hans	1.3B	Sino-Tibetan
Japanese	ja	jpn	Jpan	119.0M	Japonic
Korean	ko	kor	Hang	75.0M	Koreanic
Standard Arabic	arb	arb	Arab	—	Afro-Asiatic
Hindi	hi	hin	Deva	544.0M	Indo-European
Bengali	bn	ben	Beng	266.0M	Indo-European
Tamil	ta	tam	Taml	81.5M	Dravidian
Telugu	te	tel	Telu	95.0M	Dravidian
Thai	th	tha	Thai	55.0M	Tai-Kadai
Vietnamese	vi	vie	Latn	85.0M	Austroasiatic
Standard Indonesian	id	ind	Latn	171.0M	Austronesian
Standard Malay	zsm	zsm	Latn	—	Austronesian
Swahili	swh	swh	Latn	—	Atlantic-Congo
Yoruba	yo	yor	Latn	28.0M	Atlantic-Congo
Igbo	ig	ibo	Latn	28.0M	Atlantic-Congo
Hausa	ha	hau	Latn	28.2M	Afro-Asiatic
Amharic	am	amh	Ethi	36.0M	Afro-Asiatic
Burmese	my	mya	Mymr	36.0M	Sino-Tibetan
Central Khmer	km	khm	Khmr	15.0M	Austroasiatic
Lao	lo	lao	Lao	5.1M	Tai-Kadai
Georgian	ka	kat	Geor	3.4M	Kartvelian
Eastern Armenian	hy	hye	Armn	3.0M	Indo-European

Table 4 – Language metadata table generated with the help of QQ ([source-script](#)).