CommonLID: Re-evaluating State-of-the-Art Language Identification Performance on Web Data

Anonymous ACL submission

Abstract

Language identification (LID) is a fundamental step in curating multilingual corpora. However, LID models still perform poorly for many languages, especially on the noisy and heterogeneous web data often used to train multilingual language models. In this paper, we introduce CommonLID, a community-driven, human-annotated LID benchmark for the web domain, covering 109 languages. Many of the included languages have been previously underserved, making CommonLID a key resource for developing more representative high-quality text corpora. We show CommonLID's value by using it, alongside five other common evaluation sets, to test eight popular LID models. We analyse our results to situate our contribution and to provide an overview of the state of the art. In particular, we highlight that existing evaluations overestimate LID accuracy for many languages in the web domain. We make CommonLID and the code used to create it available under an open, permissive license.

1 Introduction

Language technologies should be useful tools for speakers of all languages, but this is far from the case. For example, even the most powerful language models only work well for a small number of languages (namely English and Mandarin Chinese). This is largely due to a lack of available high-quality datasets for many languages (Kreutzer et al., 2022; Foroutan et al., 2025).

Language identification (LID) is one of the first steps in creating larger and better datasets. Despite claims that LID is "solved" (e.g. McNamee, 2005), LID performance is still quite low for many languages including in key domains, such as web data. Even in English, LID models perform worse on social media text and non-standard varieties (Blodgett et al., 2017), which make up a large portion of web-crawl-based pre-training datasets such as those derived from Common Crawl[1].

In addition, many LID systems either do not explicitly support under-served languages or perform poorly on them. For such languages, a majority of the available training data is religious text (Foroutan et al., 2025), meaning that LID models covering these languages are expected to work on both domains and registers on which they were not trained. This limits practitioners' ability to make use of web data as a valuable potential source of training examples, since under most approaches, LID systems will continue to perform poorly on web data for under-served languages. This in turn limits the development of language technologies such as large language models for languages other than English (LOTE), especially low-resource languages. Improving LID models would allow us to improve datasets and in turn improve language model performance for many languages.

To help address these problems, we introduce **CommonLID**, a crowd-sourced LID dataset, which we release under an open permissive license.[2] The dataset is composed of line-level annotations by native speakers on web text coming from Common Crawl, created created in collaboration with over eighty annotators.[3] The dataset includes 109 language varieties, 78 of which contain at least 100 lines. CommonLID is intended as a LID benchmark to highlight where current datasets may be overestimating performance in low-resource, heterogeneous and noisy web contexts. This will help develop more robust models.

To facilitate future research in this topic, we also provide a comprehensive comparison of eight of the most common LID models on LID across multiple domains, including web data. We show that LID is far from solved: using domain-specific

[1] https://commoncrawl.org/

[2] Link to dataset on publication.

[3] All annotators who annotated more than 100 documents were invited to be co-authors on this paper.

evaluation datasets, we see that no existing model performs well across all domains. Developing better LID systems is essential for reducing crosslingual inequity in the field. To do this, we need better domain-appropriate LID evaluation datasets for more languages.

2 Issues Affecting LID datasets

2.1 Training Datasets

Issues with the datasets used to train LID systems allows us to identify the issues with the resulting systems. By identifying weaknesses in LID systems, we can design targeted evaluations for the development of better models.

Lack of Open Data The datasets used to train the most popular LID systems (e.g. CLD2 (Sites et al., 2013), CLD3 (Salcianu et al., 2020), FUN-LangID (Caswell, 2024), and fasttext) are proprietary and therefore not publicly available. Other models (e.g. Okorie (2025)) do not release training data due to legal concerns. Whilst there are models which make their data available (e.g. Franc (Wormer, 2023), GlotLID (Kargaran et al., 2023), and OpenLID (Burchell et al., 2023)), there is still a lack of open data for training SOTA LID models, especially in the long tail of language coverage.

Language Coverage Many of the most popular LID models do not support more than approximately 200 languages, reflecting the coverage of most existing multilingual datasets: e.g. Universal Dependencies (Nivre et al., 2016, 2020) and Europarl (Koehn, 2005; Tiedemann, 2012). Newer corpora have been released aimed at combatting this. One example is The Corpus of Global Language Use (Dunn, 2020), which provides increased representation to regions such as South Asia, Sub-Saharan Africa, and Oceania. Smol (Caswell et al., 2025) is another recent effort, covering 123 languages which are rarely represented in other datasets. There have also been efforts to develop region-specific datasets, such as AfroLID (Adebara et al., 2022; African languages) and Blaschke et al. (2023) (Germanic languages). This can offer targeted improvements for languages in these regions.

Domain and Register For LID systems that cover over a few hundred languages, the primary source of data for most low-resource languages comes from religious texts: for example, the Parallel Bible Corpus (Mayer and Cysouw, 2014) which covers over 1000 languages, and MIL-TALE (Brown, 2014) which covers 2110 languages with the data sourced largely from religious text. Since religious data is all that is available for training for many languages in the so-called "long tail", most of the languages covered by a high-coverage LID model will only have been trained on religious text, limiting domain generalisation for these languages (Costa-Jussà et al., 2022; Goot, 2025). For example, GlotLID has been shown to have a bias towards selecting religious text when classifying heterogeneous web data (Penedo et al., 2025).

Many existing corpora do not include any web data, usually due to its noisy nature (e.g. GlotLID, OpenLID). A notable exception is Smol, which is comprised of professionally translated web data. Instead, many datasets are derived from "cleaner" sources like Wikipedia (e.g. FLORES; NLLB Team et al., 2024) and government documents (Europarl, UDHR), which are distinct from the web domain in style and register.

Human Verification The language labels of many datasets used for training LID models are not human verified, leading to questions around their reliability. For example, two of the largest multilingual datasets in OPUS (Tiedemann, 2012), CCMatrix (Schwenk et al., 2021)) and the Corpus of Global Language Use (Dunn, 2020) were not manually checked. Later work by Kreutzer et al. (2022) found that the automatically-assigned labels in many large-scale datasets are inaccurate, which has a deleterious effect on downstream models.

2.2 Evaluation Datasets

Evaluation sets have their own specific issues in addition to those affecting training data more broadly, which motivate the development of CommonLID.

Web Domain and Noisy Data Most systems are evaluated on clean, high-quality datasets such as FLORES and UDHR (Costa-Jussà et al., 2022; Kargaran et al., 2023; Burchell et al., 2023). Such datasets are professionally translated and represent non-fiction and legal domains. They do not include many of the artifacts of web data, and therefore may not be indicative of performance on such text.

Decreasing Availability Some datasets are not publicly available, such as the AfroLID test split, or were available but are no longer allowed to be used, such as JW300 (Agić and Vulić, 2019) and Twitter (Zubiaga et al., 2016). These cases are part

of a growing trend in which permissions for data usage are being retracted, and is even more common for higher-quality and more frequently used data sources (Longpre et al., 2024). Consequently, it is increasingly important to create and use open evaluation datasets to support replicable research.

Dataset Size Some key datasets only contain a small number of items per language. UDHR, for example, only contains approximately 90 lines per language. Bible data are an exception, as they are large enough, are available and offer excellent language coverage. However, such data is from a limited domain and is often included as training data, especially for under-served languages.

3 Annotation Collection/Dataset Creation

Collecting annotations for the CommonLID dataset was a multi-step, community driven endeavour. The process had three main parts. Firstly, we collected multilingual web data by sampling from recent filtered Common Crawl crawls and MADLAD-400 (Kudugunta et al., 2023). Secondly, we recruited annotators, primarily from the NLP community. Thirdly, we collected annotations via a custom interface, often via hosted hackathons. We discuss the annotation process in more detail below.

Sampling Common Crawl We sampled data for contributors to annotate using the Ungoliant pipeline (Abadji et al., 2021) from the OSCAR project (Abadji et al., 2022; Ortiz Suárez et al., 2019). We use three LID models for selection: fastText (Joulin et al., 2017; Joulin et al., 2016), OpenLID (Burchell et al., 2023) and GlotLID (Kargaran et al., 2023).

Samples are taken from the WET files (text extracted from HTML) of the `CC-MAIN-2024-22` and `CC-MAIN-2025-05` crawls. We sampled 1,000 documents per language when available from both crawl with each of the three LID models, meaning that a maximum of 6,000 documents were sampled per language. We used the OSCAR automatic text quality annotations introduced in Abadji et al. (2022) to filter out short and noisy documents full of non-linguistic data, since they are based on very simple heuristics that do not target any specific domain or language register.

We also included the noisy splits from MADLAD-400 to create data for annotation in order to increase the amount of data available for some less-represented languages, and because it uses a distinct architecture for LID in web data (Caswell et al., 2020). We sampled 1000 documents by language from the clean and noisy splits of both the original version and version 1.5, resulting in 4000 documents per language.

We note that selecting samples using existing LID systems is a key limitation of our methodology because it means that only data which is already recognised by a LID system will be selected. It also biases the data to genres and language registers which match the model's training data. However, given the scale of web data and the language skew on the web, it would have been unrealistic to ask annotators to check a random sample of web data until they found content in their language, especially for under-represented languages.[4] Pre-annotating data with existing models is thus a necessary compromise to achieve better language representation.

Recruitment Researchers who speak or work with these languages want better LID models, so we invited these members of the NLP community to collaboratively create a dataset that benefits both contributors and the field more broadly. We recruited via a wide range of social media and NLP Discord servers. We also partnered with grass-roots NLP organisations, which focus on driving NLP progress for languages of a particular region. This helped us connect with researchers at different career stages, especially early-career and aspiring researchers, around the world. In collaboration with these grass-roots organisations, we held virtual hackathon events to create a space for annotators to ask questions, get feedback, interact with each other, and set aside time for annotations.

Since contributors were mostly members of the NLP community, we found that they understood the value of annotated data in their languages. This meant that we had good engagement from the community and higher-quality annotations than we might expect from crowd-sourced workers (Fort et al., 2011, 2014).

Annotation Platform We modified the latest version[5] of Dynabench (Kiela et al., 2021) for the annotation interface. In this interface, users first register in order to track their number of annotations. When they start annotating, they choose the language they want to validate data for and are then presented with a series of plain text documents

[4] https://commoncrawl.github.io/cc-crawl-statistics/plots/languages

[5] https://github.com/mlcommons/dynabench/

from the pre-annotated samples in their language of choice. Annotators highlight the sections of the documents that are written in their language of choice and are also instructed to annotate other sections of the document that might be written in other languages (provided they recognize them).

Users are encouraged to highlight entire lines if possible: that is, they should do line-level language identification. The interface also allowed users to highlight or do annotations at the character-level, so that they can complete annotations when the language of a complete line was impossible to determine. Large documents were truncated in order for the interface to be responsive. Screenshots of the interface, as well as the complete annotations guidelines can be found in Appendix C.

Incentives for Participation All contributors who annotated at least 100 documents were offered authorship on the dataset paper. We also offered authorship to contributors all available documents if we had fewer than 100 for that language, which was the case for some of the most low-resource languages. Offering authorship was a key design decision of this project, consistent with the principles of participatory design proposed in Caselli et al. (2021). In particular, we wanted to engage in ethical community collaboration: treating community members with respect, equity and reciprocity. Authorship represents an important way to recognize the intellectual contributions of annotators and increases equity between all contributors of the project. This reflects the goal of improving language representation in NLP datasets—and, as a result, models. But we believe in order to do this responsibly, this requires deep cooperation with language communities and giving contributors ownership over their data.

We also incentivized contributors to annotate more documents through a leaderboard displayed on the annotation platform. Each contributor's username was displayed along with the number of documents they annotated. While most contributors annotated between 100 and 200 documents, some participants were far more prolific: 10 annotated over 1,000 documents, and 5 contributors who annotated over 2,000 documents.

4 CommonLID dataset

We finished collecting annotations for the version of CommonLID described in this paper on 25 November 2025. We then processed the collected data to create a corpus annotated with the dominant language variety in each line.

4.1 Dataset preparation

Creating line-level labels Reformatting the annotated data to be labelled at the line level was not straightforward. Firstly, some participants had misunderstood the instructions and either labelled the text at the word level, resulting in multiple labels per line, or labelled all text in the page as one language when in fact multiple languages were present. Secondly, accurate multilingual sentence splitting is not a trivial task. Thirdly, some lines contain content which is hard to classify into any particular language variety (e.g. URLs, proper nouns).

To avoid complications with multilingual sentence splitting, we simply split the text based on new lines. We then removed leading and trailing white space and deduplicated language variety label and line pairs. We used Levenshtein distance between lines to identify lines which were extremely similar but had been assigned different language variety labels to allow for fuzzy splitting.

We noticed many lines with multiple labels were very short to the extent that LID was not a meaningful task, so we filtered out any lines with fewer than 10 characters. We also noticed that many non-English languages contained a significant number of English lines. We therefore used the `langid.py` LID tool[6] to predict whether lines were English, as a fast and simple LID tool. Any short lines predicted to be English or longer lines with predicted probability of being English > 0.7 were filtered out, based on empirical results. We exempted lines labelled as Scottish Gaelic, Irish Gaelic, Nigerian Pidgin, Southern Sotho, and Shona from the English filter since the false positive rate was too high.

Reconciling multiple labels After the initial cleaning, one of the authors (a British English speaker able to read Latin and Arabic scripts) manually inspected all rows with multiple labels. Some lines remained which were actually English but labelled as different distinct languages, or conversely labelled as English but were not in reality. We assumed this was an artifact of over-use of the 'select all' feature in the interface (see Figure 3, top right) and removed these lines. Short lines containing solely non-linguistic content (e.g. numbers, URLs, emoji) were also removed.

[6] https://github.com/saffsd/langid.py

We found two categories of annotation where we could not reconcile the distinct labels. The first category was lines labelled with both the macro- and micro-language label (e.g. Arabic and Egyptian Arabic, Malay and Indonesian). The second category was lines containing multiple languages but without a dominant language. In both cases, we kept multiple copies of the line with the different annotations. The existence of multiple valid labels for a single line is a key limitation of our labelling scheme and should be addressed in future work (Keleg et al., 2023; Burchell et al., 2024a).

Per-language audit For each language class, we checked a random sample of approximately 100 lines to check for language correctness or other quality issues, following the heuristics in Burchell et al. (2023). Three language variety classes were spurious, only containing 1-2 lines which were clearly not in the intended language (`gux`, `swe`, and `tag`). These were removed.

4.2 Dataset description

CommonLID contains 373,230 lines in total with a mean line length of 215.5 characters. This makes CommonLID over ten times larger than Smol. There are 109 language varieties, with the largest containing 43,189 lines (`uzb`), and the 4 smallest containing a single line (`ace`, `pol`, `grc`, `gle`). The macro-average number of lines per class is 3424.1, and 78 classes contain more than 100 examples. Figure 1 illustrates the rough geographical distribution of the languages covered in CommonLID, with the dot size representing the number of lines. Appendix A contains a full breakdown of number of lines and mean line length per language class.

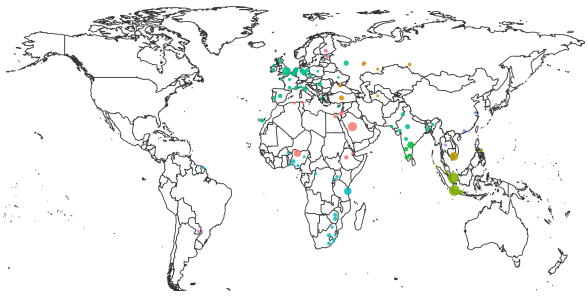

Figure 1: Illustrative geographical distribution of the language varieties in CommonLID. The dot size represents the number of annotated lines, and colour represents language family.

5 Evaluation

We analysed the performance of eight widely-used existing LID models on our dataset, and compared the performance of the same models on comparable existing evaluation datasets. This extensive testing situates CommonLID in the context of previous work and demonstrates its value as a web-derived human-annotated evaluation dataset. It also serves as a reference for the state of the art for LID models to aid future research and development. We release all our raw evaluation scores and analysis code.[7]

5.1 Evaluation datasets

The majority of our evaluation datasets are openly available which is our preference. However, we also use two restricted datasets for where we could not find an open alternative: a Bible dataset since it includes an extremely wide range of languages, and a social media dataset since it includes colloquial text often found on the web. We provide further details about all evaluation datasets in Appendix D.

Three of the datasets (SmolSent, Bibles, and social media) had a large variation in the number of examples per class in their original form. This had the double effect of greatly increasing inference times due to the large classes as well as leading to unrepresentative results for the smallest classes. To combat this, we discarded classes in these datasets with fewer than 300 examples, then sampled 300 lines at random from the remaining classes to form our test splits for these datasets.

5.2 LID models

We test eight LID models using the evaluation sets: AfroLID (Adebara et al., 2022), CLD2 (Sites et al., 2013), (NLLB) fasttext (Costa-Jussà et al., 2022), FUN-LangID (Caswell, 2024), pyFranc (Wormer, 2023), CLD3 (Salcianu et al., 2020), GlotLID v4 (Kargaran et al., 2023), and OpenLID-v2 (Burchell et al., 2023). We summarise details of these models in Table 5 in Appendix B. Prior to prediction, we normalise the input text by removing surrounding white space, applying lowercasing, removing non-word characters and digits, and squeezing white space. We found this improved the robustness of LID models to non-standard input without assuming an input language.

[7] Link provided upon publication.

	FLORES+ 209 languages		SmolSent 82 languages		UDHR-LID 418 languages		Bibles 1047 languages		Social Media 66 languages		CommonLID 109 languages	
	all	cov.	all	cov.	all	cov.	all	cov.	all	cov.	all	cov.
AL	15.9	67.9 (49)	42.3	69.4 (50)	15.2	67.6 (94)	12.1	69.8 (181)	1.5	**98.0** (1)	9.2	43.5 (23)
C2	45.0	87.8 (105)	40.6	83.2 (40)	24.0	81.7 (121)	4.4	73.8 (62)	76.8	85.9 (59)	49.5	**79.3** (68)
FT	80.3	92.2 (182)	48.1	83.8 (47)	27.0	68.2 (166)	3.3	43.8 (79)	**83.6**	84.9 (65)	49.3	72.6 (74)
FL	68.8	86.2 (165)	**75.9**	79.8 (78)	59.0	78.2 (314)	71.3	89.1 (837)	58.7	65.5 (58)	46.4	57.8 (86)
Fr	61.9	80.4 (160)	46.4	76.0 (50)	**93.4**	**95.9** (407)	7.6	52.1 (152)	62.4	66.4 (62)	39.3	61.3 (70)
C3	28.1	71.5 (78)	11.9	57.4 (17)	11.0	54.4 (80)	2.1	42.8 (47)	63.1	75.7 (54)	34.3	66.2 (55)
GL	**94.2**	**96.5** (204)	74.7	**91.4** (67)	73.7	84.4 (366)	**82.3**	**93.0** (927)	80.6	85.8 (62)	**60.4**	68.6 (96)
OL	83.5	90.4 (193)	45.4	82.7 (45)	25.1	67.3 (156)	3.3	44.6 (78)	83.4	88.7 (62)	47.4	68.0 (76)

Table 1: Macro-averaged F1 scores achieved by tested models on the evaluation sets: AfroLID (AL), CLD2 (C2), fasttext (FT), FUN-LangID (FL), pyFranc (Fr), CLD3 (C3), GlotLID (GL), and OpenLID-v2 (OL). Scores are calculated over the whole dataset (*all*) and on the subset of language varieties covered by the model (*cov.*). Count of languages in the evaluation set covered by the model in parentheses, highest score per column in **bold**.

5.3 Large language models

Furthermore, we evaluate large language models (LLMs) from OpenAI as baselines. Specifically, we compare GlotLID with GPT-4o, GPT-4o-mini, GPT-5, GPT-5-mini in a zero-shot setting. The corresponding LLM prompts are implemented with DSPy (Khattab et al., 2023) without optimisation.

5.4 Label normalisation

The LID datasets and models tested in this work do not share a common language coding schema, meaning that the labels must be normalised to allow large-scale comparison. Our process is as follows:

1. Raw language code strings are trimmed at the first '`-`' or '`_`' character to keep the leading segment (e.g., `en-US` → `en`, `zh_Hant` → `zh`).
2. The trimmed segment is parsed by the Python `iso639-lang` library[8], which provides comprehensive error reporting on invalid and deprecated language codes.
3. The resulting language code is checked for ISO 639-3 compliance, mapping deprecated codes to their modern equivalents where possible. Dataset entries and model outputs which cannot be resolved to a single ISO 639-3 language are treated as undefined and discarded.

Automatic normalisation in this way has the potential to miss labels which could be compared (e.g. macro- and micro- language codes). However, manual reconciliation of so many language labelling schemes would be infeasible.

[8] https://github.com/LBeaudoux/iso639, v2.5.1

5.5 Metrics and scoring

Following previous work (Burchell et al., 2023; NLLB Team et al., 2024; Kargaran et al., 2023), we use F1 score and false positive rate (FPR) as our primary metrics. All calculated averages are macro-averages so that less-resourced language varieties are given the same weight as those with more examples in the evaluation sets.

5.6 Comparing models

Comparing the performance of different LID systems involves comparing between classifiers which output non-overlapping classes. Different LID models cover different language varieties and cannot output the correct label for unseen languages. This means that when taking a simple average over the test set, models with more coverage have an advantage since a model will always score zero for a language it does not include.

That said, higher-coverage models are not always better in practice: low LID accuracy for less-resourced languages results in unacceptably low quality labelled data, hindering performance downstream (Caswell et al., 2020). A more reliable but lower-coverage LID model may be preferable depending on the application.

6 Results

6.1 LID models

Table 1 shows F1 scores achieved by each model on each dataset, macro-averaged over language varieties. Two scores are given: one calculated over the whole evaluation set where languages not included in the model score zero (*all*), and one calculated over the subset of language varieties covered by the model (*cov.*). The number of languages over

which the macro-average is calculated in the latter case is also given. For all models, the difference in these two scores highlights the ongoing issue of determining a fair way of comparing LID models with disparate language coverage. It also provides a fuller picture of the trade-off between coverage and specialisation, which is most apparent for AfroLID.

The results for the CommonLID dataset in Table 1 show that it is a challenging dataset, with most models only achieving F1 scores in the 60s averaged over the languages they cover. We believe the current low scores on CommonLID demonstrate its worth as a novel evaluation dataset, since it covers a different domain compared to other datasets available for under-served languages.

Some of the higher scores for other datasets can be attributed to a likely overlap between the model's training data and the evaluation set: Franc is trained on UDHR, GlotLID uses Bible data as a majority of its training data for many languages in the long tail, and the developers of FUN-LangID were also part of the team behind SmolSent (though the exact training data for both is unknown). The effect on reported scores is exacerbated for high-coverage models, since for these most language classes are in the long tail where training and evaluation data are more similar. Taking the macro-average weights all languages equally, so given the majority of the languages are in the long tail, the average overall score is high. These high scores do not necessarily reflect real-world performance, showing the need for more diverse, independent evaluation data for long-tail languages.

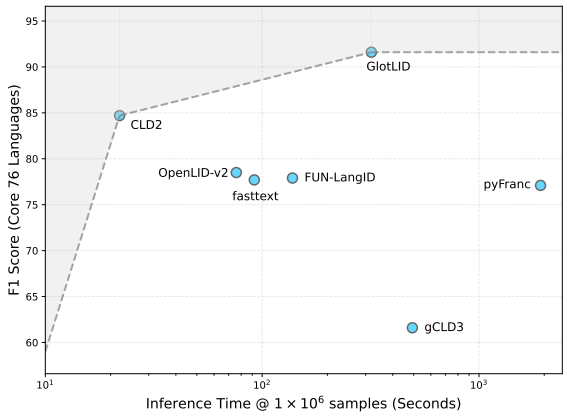

Figure 2: Inference speed vs. F1 (76 core languages, combined datasets)

GlotLID shows strong performance for many evaluation sets and has the highest reported language coverage of the models we test. This comes at a cost of slower inference: Figure 2 plots inference speed versus F1 score for all models but AfroLID on a core set of 76 shared languages. CLD2 and GlotLID are on the Pareto frontier of compute-performance trade-off, as GLotLID's improved performance comes at the cost of significantly slower inference. Table 7 contains full details of the inference speed for all models.

	High-coverage 1351 languages		Afro* 294 languages		Core 76 languages	
	F1 ↑	FPR ↓	F1 ↑	FPR ↓	F1 ↑	FPR ↓
GL	87.8	0.006%	88.7	0.02%	91.6	0.1%
FL	76.3	0.02%	73.5	0.08%	77.9	0.2%
AL	—	—	71.8	0.7%	—	—
Fr	—	—	—	—	77.1	0.4%
FT	—	—	—	—	77.7	0.5%
OL	—	—	—	—	78.5	0.5%
C2	—	—	—	—	84.7	0.2%
C3	—	—	—	—	61.6	1.4%

Table 2: Performance for all test data combined for the subset of languages supported by all compared models. We give results for GlotLID (GL), FUN-LangID (FL), AfroLID (AL), pyFranc (Fr), fasttext (FT), OpenLID-v2 (OL), CLD2 (C2), and CLD3 (C3). Performance should only be compared within each column.

As mentioned in Section 5.6, it is often difficult to make a direct comparison between LID models which cover different sets of language varieties. We decided that the fairest way to compare models directly was to evaluate them on test sets containing only the languages which were supported by all models. Table 2 shows the results for this type of evaluation. "High-coverage" compares results for GlotLID and FUN-LangID alone as the highest-coverage models, "Afro*" compares these models with AfroLID for coverage of the languages of Africa, and "Core" compares all models but AfroLID on a core set of 76 language varieties all share.[9] Note that only metrics in the same column can be compared to each other. These results show that GlotLID performs strongly across all comparisons both in terms of F1 and FPR.

6.2 LLMs for LID

Given that LLMs are state-of-the-art for many NLP tasks, we report the results of four OpenAI GPT models based on three language groups in Table 3. The language groups are the same as in Table 2. To reduce LLM inference costs, we down-sample the evaluation datasets (15k samples in total). The

[9] AfroLID's focus on the languages of Africa means that only one language, Afrikaans, is shared between all models.

	High-cov. 1351 langs.		Afro* 294 langs.		Core 76 langs.	
	F1 ↑	FPR ↓	F1 ↑	FPR ↓	F1 ↑	FPR ↓
GlotLID	90.0	0.01%	90.6	0.05%	93.5	0.13%
OpenAI GPT models						
4o-mini	48.5	0.09%	43.6	0.69%	81.0	0.54%
4o	62.0	0.07%	57.3	0.46%	89.0	0.33%
5-mini	60.4	0.06%	55.1	0.43%	76.4	0.53%
5	70.3	0.05%	66.6	0.29%	91.8	0.22%

Table 3: A comparison of large language models from OpenAI's GPT family (4o-mini, 4o, 5-mini, 5) and GlotLID (best performining LID model). We report performance on the combined but down-sampled test data (15k test samples), subsets as in Table 2.

best performing LID model, GlotLID, is used as a baseline. Despite requiring magnitudes more resources, the GPT models are outperformed by GlotLID. The performance gap is smaller for the core languages (-1.8% F1 with GPT-5) and larger for the African languages (-30% F1 with GPT-5).

7 Discussion

Creating CommonLID We created CommonLID in collaboration with many native speakers. The result significantly expands the availability of LID web-domain evaluation data, especially for low-resource languages. Our work complements other collaborative efforts, like Smol, though unlike previous efforts we annotate texts originally in the target language rather than translated texts.

The creation of the dataset was constrained by the data available for annotation. We used existing LID models and datasets to select texts to annotate, meaning it was extremely difficult to collect texts in languages not supported by these models or datasets for contributors to annotate. During our hackathons, some participants were unable to annotate significant amounts of data because we had few documents available. This highlights one of the challenges of LID model development: it is not simply a matter of gathering data. All existing tools and methods to *begin* the annotation process break down for all but the highest resource languages. In the future, we hope to work with native speakers to collect textual resources in their languages to help improve language coverage in NLP technologies and as a step towards annotating such data.

Comparing LID Models Given a range of LID models, it is unclear how to compare them fairly. Each model has a different number of language labels it is trained to support. Unifying the language labels across models is far from straightforward, and even delineating boundaries between some language varieties is contentious (Burchell, 2024).

For many of the languages in the highest-coverage models like GlotLID and FUN-LangID, there is no test split distinct from the training data for many languages, since data for many languages in the long tail is already so sparse and usually limited to religious text. Therefore, it is currently impossible to evaluate LID performance for the lowest resource languages in a meaningful way. This is one way in which current evaluation practices over-estimate LID performance.

State-of-the-Art LID Considering the trade-off between coverage and inference speed, our results show that CLD2 and GlotLID perform best overall. CLD2 out-performs GlotLID in some contexts, but does not support lower-resource languages. GlotLID performs best on FLORES and Bibles, which represent some of the cleanest evaluation datasets. In particular, GlotLID's high F1 on Bible data reflects the fact that for most long-tail languages Bible data is the only LID training data for GlotLID. It is likely that performance for these languages would be more limited if the evaluation data covered a wider range of domains and were more distinct from the training data.

That said, there is no clear top-performing model given the complexity of meaningful comparison and evaluation. What is clear is that LID, especially in the long tail, remains challenging. There is no model that achieves >75% F1 across all evaluation datasets, even when coverage is taken into account. There is a clear need for better LID models. CommonLID helps to select and develop models that will work best on web data.

8 Conclusion

We presented CommonLID, a LID evaluation dataset in the web domain. We detailed our highly collaborative data collection and annotation process, in which we worked with contributors from multiple language communities. To show CommonLID value as a benchmark, we used it to test eight popular LID models alongside other common LID evaluation datasets. We analysed our results to provide researchers with an overview of the state of the art in this task and thus support further work. Our data and code are made available to the community under an open, permissive license.

Limitations

As mentioned in Section 3, we used three existing models and one existing dataset in select samples to be validated and annotated by contributors. This inherently biases our dataset to the languages, genres and language registers supported by these models and dataset. Moreover, these three models were used in our evaluation, which could have potentially skewed the results. We aimed to mitigate this by evaluating CommonLID along with other existing datasets.

Other concerns are inherent to the task of language identification: for example, the existence of multiple labels for a given single line and the existence of macro- and micro- language. These can significantly complicate the annotation task, even for native speakers. We aimed to reduce the ambiguity by presenting full documents to the annotators so that they could have more context, but this does not remove all of the uncertainty inherent in the task. Moreover, as mentioned in Section 5.6 and Section 7, evaluating and comparing models remains challenging, especially as we had to account for issues such as deprecated language codes, non-overlapping classes and existing data sparsity for many languages in the long tail. All of these factors complicate the evaluation and remain challenging to address.

We sourced the data used to build CommonLID from Common Crawl web crawl data. This means it has the potential to contain personally identifiable information or other harmful content. We mitigated this by using heuristic quality filters on the data prior to presenting it to participants and by making it easy for participants to contact us with any concerns. Moreover, we chose Common Crawl as a source because they have always respected robots.txt[10], meaning that website owners have the option to opt out of crawling very easily.

Finally, given that we targeted annotations for a wide range of languages, we were unable to find more than one volunteer native speaker for many of them. This made it impossible to conduct an inter-annotator agreement study for our data.

We are not aware of any potential harms from this work. In fact, we hope this work helps to address harms from existing LID models and helps improve cross-lingual equity in NLP.

[10] https://commoncrawl.org/ccbot

References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4344–4355, Marseille, France. European Language Resources Association.

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2021. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-9) 2021. Limerick, 12 July 2021 (Online-Event)*, pages 1 – 9, Mannheim. Leibniz-Institut für Deutsche Sprache.

Ife Adebara, AbdelRahim Elmadany, Muhammad Abdul-Mageed, and Alcides Inciarte. 2022. AfroLID: A neural language identification tool for African languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1958–1981, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Željko Agić and Ivan Vulić. 2019. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Verena Blaschke, Hinrich Schuetze, and Barbara Plank. 2023. A survey of corpora for Germanic low-resource languages and dialects. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 392–414, Tórshavn, Faroe Islands. University of Tartu Library.

Su Lin Blodgett, Johnny Wei, and Brendan O'Connor. 2017. A dataset and classifier for recognizing social media English. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 56–61, Copenhagen, Denmark. Association for Computational Linguistics.

Ralf Brown. 2014. Non-linear mapping for improved identification of 1300+ languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 627–632, Doha, Qatar. Association for Computational Linguistics.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 865–879, Toronto, Canada. Association for Computational Linguistics.

Laurie Burchell, Alexandra Birch, Robert Thompson, and Kenneth Heafield. 2024a. Code-switched language identification is harder than you think. In *Proceedings of the 18th Conference of the European*

Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 646–658, St. Julian's, Malta. Association for Computational Linguistics.

Laurie Burchell, Jean Maillard, Antonios Anastasopoulos, Christian Federmann, Philipp Koehn, and Skyler Wang. 2024b. Findings of the WMT 2024 shared task of the open language data initiative. In *Proceedings of the Ninth Conference on Machine Translation*, pages 110–117, Miami, Florida, USA. Association for Computational Linguistics.

Laurie Vear Burchell. 2024. *Improving natural language processing for under-served languages through increased training data diversity*. Ph.D. thesis, University of Edinburgh.

Tommaso Caselli, Roberto Cibin, Costanza Conforti, Enrique Encinas, and Maurizio Teli. 2021. Guiding principles for participatory design-inspired natural language processing. In *Proceedings of the 1st Workshop on NLP for Positive Impact*, pages 27–35, Online. Association for Computational Linguistics.

Isaac Caswell. 2024. Funlangid. https://github.com/google-research/url-nlp/tree/main/fun_langid.

Isaac Caswell, Theresa Breiner, Daan van Esch, and Ankur Bapna. 2020. Language ID in the wild: Unexpected challenges on the path to a thousand-language web text corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6588–6608, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Isaac Caswell, Elizabeth Nielsen, Jiaming Luo, Colin Cherry, Geza Kovacs, Hadar Shemtov, Partha Talukdar, Dinesh Tewari, Baba Mamadi Diane, Djibrila Diane, Solo Farabado Cissé, Koulako Moussa Doumbouya, Edoardo Ferrante, Alessandro Guasoni, Christopher Homan, Mamadou K. Keita, Sudhamoy DebBarma, Ali Kuzhuget, David Anugraha, and 5 others. 2025. SMOL: Professionally translated parallel data for 115 under-represented languages. In *Proceedings of the Tenth Conference on Machine Translation*, pages 1103–1123, Suzhou, China. Association for Computational Linguistics.

Marta R Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, and 1 others. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

David Dale, Laurie Burchell, Jean Maillard, Idris Abdulmumin, Antonios Anastasopoulos, Isaac Caswell, and Philipp Koehn. 2025. Findings of the WMT 2025 shared task of the open language data initiative. In *Proceedings of the Tenth Conference on Machine Translation*, pages 495–502, Suzhou, China. Association for Computational Linguistics.

Jonathan Dunn. 2020. Mapping languages: the corpus of global language use. *Lang. Resour. Eval.*, 54(4):999–1018.

Negar Foroutan, Jakhongir Saydaliev, Ye Eun Kim, and Antoine Bosselut. 2025. Conlid: Supervised contrastive learning for low-resource language identification. *arXiv preprint arXiv:2506.15304*.

Karën Fort, Gilles Adda, and K. Bretonnel Cohen. 2011. Last words: Amazon Mechanical Turk: Gold mine or coal mine? *Computational Linguistics*, 37(2):413–420.

Karën Fort, Gilles Adda, Benoît Sagot, Joseph Mariani, and Alain Couillault. 2014. Crowdsourcing for language resource development: Criticisms about amazon mechanical turk overpowering use. In *Human Language Technology Challenges for Computer Science and Linguistics*, pages 303–314, Cham. Springer International Publishing.

Rob Van Der Goot. 2025. Identifying open challenges in language identification. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18207–18227, Vienna, Austria. Association for Computational Linguistics.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. FastText.zip: Compressing text classification models. *arXiv e-prints*, arXiv:1612.03651.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. 2023. GlotLID: Language

identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore. Association for Computational Linguistics.

Amr Keleg, Sharon Goldwater, and Walid Magdy. 2023. ALDi: Quantifying the Arabic level of dialectness of text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10597–10611, Singapore. Association for Computational Linguistics.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T Joshi, Hanna Moazam, and 1 others. 2023. Dspy: Compiling declarative language model calls into self-improving pipelines. *CoRR*.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, and 33 others. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: A multilingual and document-level large audited dataset. In *Advances in Neural Information Processing Systems*, volume 36, pages 67284–67296. Curran Associates, Inc.

Shayne Longpre, Robert Mahari, Ariel Lee, Campbell Lund, Hamidah Oderinwale, William Brannon, Nayan Saxena, Naana Obeng-Marnu, Tobin South, Cole Hunter, Kevin Klyman, Christopher Klamm, Hailey Schoelkopf, Nikhil Singh, Manuel Cherep, Ahmad Mustafa Anis, An Dinh, Caroline Chitongo, Da Yin, and 30 others. 2024. Consent in crisis: The rapid decline of the ai data commons. In *Advances in Neural Information Processing Systems*, volume 37, pages 108042–108087. Curran Associates, Inc.

Thomas Mayer and Michael Cysouw. 2014. Creating a massively parallel Bible corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3158–3163, Reykjavik, Iceland. European Language Resources Association (ELRA).

Paul McNamee. 2005. Language identification: a solved problem suitable for undergraduate instruction. *J. Comput. Sci. Coll.*, 20(3):94–101.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

Chijioke I Okorie. 2025. The fair dealing/fair use landscape for artificial intelligence innovation and computational research in africa. *International Review of Law, Computers & Technology*, pages 1–28.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019*, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

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*.

Alex Salcianu, Andy Golding, Anton Bakalov, Chris Alberti, Daniel Andor, David Weiss, Emily Pitler, Greg Coppola, Jason Riesa, Kuzman Ganchev, Michael Ringgaard, Nan Hua, Ryan McDonald, Slav Petrov,

Stefan Istrate, and Terry Koo. 2020. cld3: Compact language detector v3. https://github.com/google/cld3. Archived; Apache-2.0 License.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

Dick Sites, Jason Riesa, and Ivan Giuliani. 2013. cld2: Compact language detector 2. https://github.com/CLD2Owners/cld2. Apache-2.0 License; detects 83 languages :contentReference[oaicite:0]index=0.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Titus Wormer. 2023. pyfranc.

Arkaitz Zubiaga, Iñaki San Vicente, Pablo Gamallo, José Ramom Pichel, Inaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Víctor Fresno. 2016. Tweetlid: a benchmark for tweet language identification. *Language Resources and Evaluation*, 50(4):729–766.

A CommonLID statistics by language

Lang. code	Num. lines	Mean length	Lang. code	Num. lines	Mean length
ace	1	28.0	jpn	3,344	171.9
acf	603	241.0	kab	116	118.2
aeb	15	220.1	kan	2,554	286.1
afr	88	193.7	kik	62	237.2
amh	1,617	207.0	kor	8	70.1
apd	27	224.5	lat	50	355.7
ara	16,306	184.3	lav	512	214.1
arb	26,152	211.4	lij	147	320.1
arg	2,342	333.7	lin	55	246.1
ars	229	224.9	ltg	38	234.5
ary	226	180.8	lug	1,361	170.6
arz	1,102	243.5	lvs	353	287.4
asm	213	247.9	mal	2,061	266.6
aze	29	227.6	mar	1,061	278.8
azj	847	277.4	mlg	2,153	197.0
bak	46	136.0	msa	28,224	188.8
bcl	270	331.9	nld	3,299	263.9
ben	1,886	204.5	nso	99	137.0
bik	1,499	268.6	nyn	5	284.4
bre	2,348	183.3	oci	1,314	225.6
bul	109	190.9	orm	1,060	233.7
cat	93	262.6	ory	608	197.0
ces	933	275.0	pan	1,020	234.9
cmn	865	169.4	pcm	2	20.5
crh	405	235.2	pol	1	273.0
deu	7,553	218.7	por	2,443	171.7
ell	7	23.9	rcf	401	195.7
eng	27,461	212.9	rus	4,003	220.9
est	659	174.3	san	895	282.3
ext	7	91.7	sna	1,355	161.1
fas	19,318	237.0	sot	943	455.6
fil	58	229.1	spa	4,236	212.7
fin	1,030	267.3	swa	4,031	158.7
fra	3,233	206.7	swh	12,383	222.5
fro	39	740.9	tam	81	218.5
fry	965	265.7	tat	1,029	203.3
fuv	37	258.7	tel	11,747	223.0
gaz	11	226.2	tgl	2,223	217.3
gcf	24	157.9	tha	3,118	237.0
gcr	111	156.1	tuk	22	285.4
gla	929	202.6	tur	4,486	233.7
gle	1	23.0	ukr	2	37.0
gom	338	219.8	urd	204	232.4
grc	1	7.0	uzb	43,189	206.9
gug	548	219.2	uzs	9	409.8
guj	948	222.4	vec	1,558	193.8
guw	4	71.0	vie	21,803	213.1
hau	16,455	166.4	wuu	631	137.9
hbo	808	535.3	xho	575	257.7
heb	5,055	167.4	yor	2,290	93.5
hin	3,666	249.2	yue	425	91.2
ibo	168	309.0	zho	11,738	152.7
ind	33,828	278.8	zsm	343	172.8
ita	4,387	229.7	zul	4	20.2
jav	1,656	231.5			

Table 4: Number of lines and mean line length in characters for each language variety in the CommonLID dataset.

B Additional Model Details

Name	# langs.	Architecture	Data sources	Open data?
AfroLID (Adebara et al., 2022)	517	Transformer	≈ 100M curated sents.	✗
CLD2 (Sites et al., 2013)	158	Naïve Bayes	Web pages (curated and scraped)	✗
fasttext (NLLB Team et al., 2024)	218	FastText	"publicly available datasets"	✗
FUN-LangID (Caswell, 2024)	1634	Common sub-strings	Web+Wikipedia+Bibles	✗
pyFranc (Wormer, 2023)	414	Trigram distribution	UDHR	✓
gCLD3 (Salcianu et al., 2020)	99	Neural network	?	✗
GlotLID v3 (Kargaran et al., 2023)	1868	FastText	Curated open sources	✓
OpenLID-v2 (Burchell et al., 2023)	193	FastText	Curated, audited open sources	✓

Table 5: Summary of LID models used.

	GlotLID	FUN-LangID	AfroLID	pyFranc	fasttext	OpenLID-v2	CLD2	gCLD3
GlotLID	**1868**							
FUN-LangID	1351	**1549**						
AfroLID	401	313	**515**					
pyFranc	362	312	93	**410**				
fasttext	202	180	49	166	**210**			
OpenLID-v2	190	157	47	156	180	**193**		
CLD2	128	153	30	121	114	104	**158**	
gCLD3	83	99	13	80	79	78	98	**99**

Table 6: Counts of mutual language coverage between LID models (models in descending order by coverage).

Model	Samples/s	Duration @$1e+6$ samples
CLD2	43735	22s
OpenLID-v2	13123	1m 16s
fasttext	10867	1m 32s
FUN-LangID	7237	2m 18s
GlotLID	3127	5m 19s
gCLD3	‡2026	8m 13s
Franc	520	32m 03s
AfroLID	†66	4h 12m 49s

Table 7: Inference speed for LID models on FLORES+. Models are listed in descending order of samples per second. For all models other than gCLD3, inference is done on a 14-core Apple M4 Pro chip with 64GB of RAM, using PyTorch MPS optimisation where possible. †AfroLID uses transformers - performance is likely significantly better on CUDA hardware. ‡Measured on an AMD EPYC 7351P (16 cores @ 2.4GHz) Linux machine with 256GB RAM as gCLD3 is not usable on modern macOS.

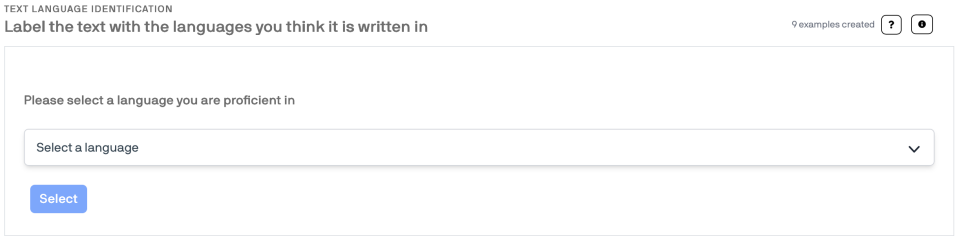

Figure 3: A screenshot of the annotation platform interface. The participant selects the language they want to annotate for.

C Annotation platform details

C.1 Annotation Instructions

In this task, annotators will be first give a prompt in which they select a language that they are proficient on. The bar is a search field so that the annotator can easily find the language they are looking for, as can be seen in Figure 3.

Then they will be presented with a text passage from a processed Common Crawl record that potentially contains content in the selected language. The annotator will then select the spans of text that they are able to identify as being written in the language they are proficient in. If the whole text passage is written in a single language, the annotator can just press the "select all text area button" to select all the text, as can be seen in Figure 4.

There are some Optional annotations that the user might wish to add, they are displayed at the bottom of the image in a dropdown. These tags are optional and are intended to start preparing future tasks.

If there are multiple languages in a single example and you can identify them, you can select an another language from the list and the select the span of text. Please try to annotate no more than single language per line. If you see any instance of code-switching or even script-switching in a single line, please try to annotate it with the language that you think is "dominant", as the example in Figure 5.

If at some point you wish to annotate examples in another language, simply select the desired language on the dropdown, and click on the "skip and load new text" button.

C.2 Potential Issues

Samples have been automatically pre-annotated by ML models so the performance for some languages might be lower than ideal. If you have seen a large amount of samples that are incorrectly labelled, please report it by clicking on the button labelled with an "i" on the to right corner. This will show you a pop-up with instructions on how to report issues with the data.

C.3 Rewards for Participation

All contributors who complete 100 annotations or more will be invited to be co-authors of a scientific paper.

D Evaluation Dataset Details

FLORES+ This is a multilingual machine translation benchmark sourced from Wikitext articles (NLLB Team et al., 2024; Guzmán et al., 2019; Goyal et al., 2022; Burchell et al., 2024b; Dale et al., 2025). We evaluated on version 4.1 of the dev split, which contains 997 parallel sentences translated into 222 language varieties. FLORES+ is a common LID evaluation set due to its high quality and wide coverage, though it only contains relatively formal text.

UHDR-LID This is a collection of cleaned translations of the Universal Declaration of Human Rights into 374 languages (Kargaran et al., 2023). Whilst its high language coverage and open availability make it attractive as a LID evaluation dataset, these factors also mean it is often used as part of multilingual training data. This means that scores on this data are likely to be inflated.

SmolSent This dataset consists of 863 English sentences covering 5519 of the most common English tokens, professionally translated into 88 under-served languages (Caswell et al., 2020). These sentences come from Common Crawl and are chosen as the smallest set of sentences covering the largest number of common English tokens. Their vocabulary coverage and web domain makes them a useful evaluation set for our task.

Bible We sourced translations of parts of the Bible in 1144 language varieties, based on the work of Mayer and Cysouw (2014). Different language varieties have varying amounts of data available, from 361,74 lines for English to just 5 for Ndjébbanatha. Religious text, particularly the Bible, is often used as the sole source of both training and test data for many under-served language varieties, the result of which is that downstream models for this language only perform well on religious text and do not generalise to other domains.

Social media To test on more informal text, we curate a multilingual dataset of 169,019 lines of social media data containing examples of posts in 97 language varieties. The largest language variety class contains 28,545 lines, whilst the seven smallest contain just one example each. This dataset contains phenomena typical of web text such as emoji, URLs, non-standard orthography and non-linguistic content like hashtags, making it a challenging but useful domain for testing robustness.

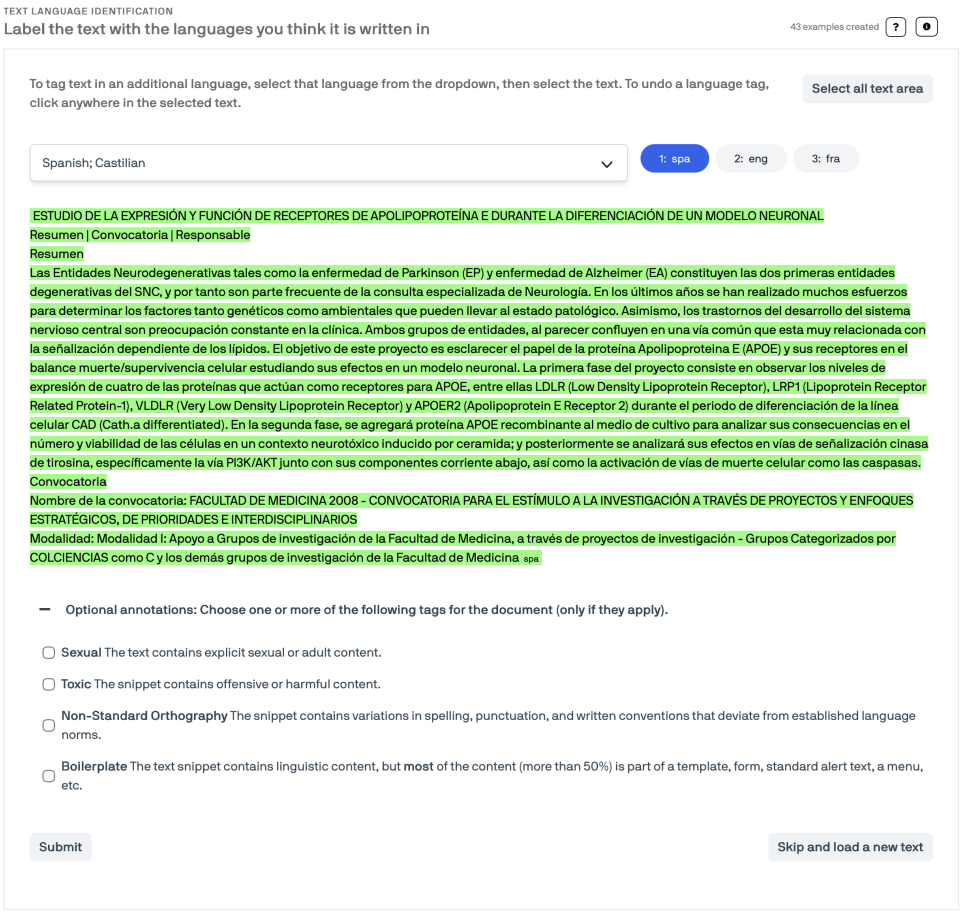

Figure 4: A screenshot of the annotation platform interface. The participant has highlighted all text as Spanish.

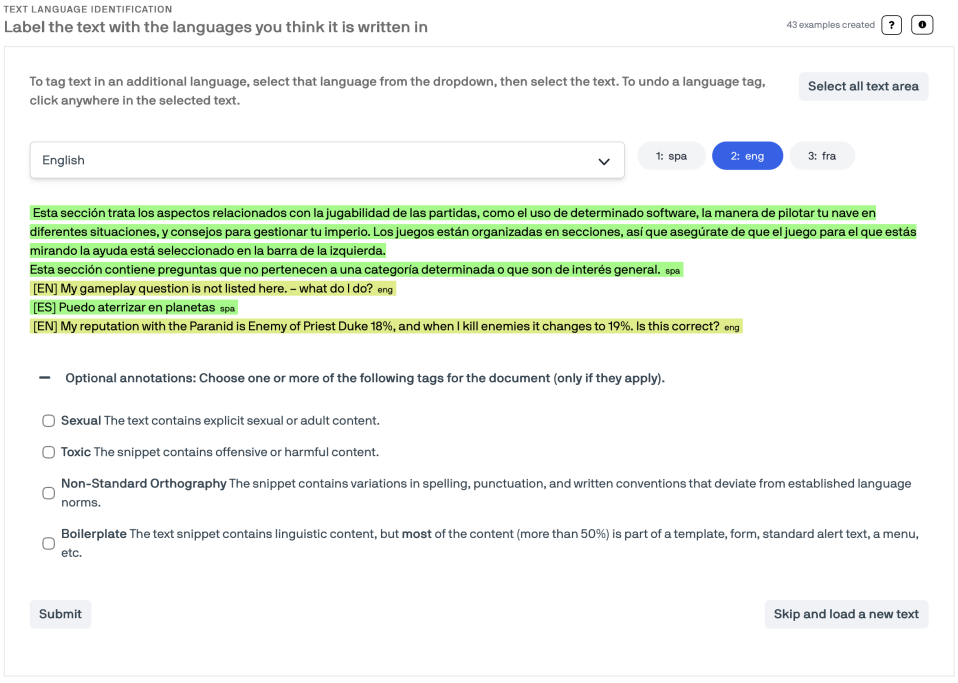

Figure 5: A screenshot of the annotation platform interface. The participant has highlighted the English and Spanish text in the extract in different colours.