Journal of High School Science

Efficient language identification in monolingual and multilingual documents

Sobrier M

Submitted: December 4, 2025, Revised: version 1, February 8, 2026, version 2, March 21, 2026, version 3, March 23, 2026
Accepted: March 24, 2026

Abstract

Language identification (LID) is a critical prerequisite for text processing tasks such as content classification, natural language processing (NLP), machine translation, and large language model (LLM) training. Modern AI systems rely heavily on web data, making accurate detection essential for applying language-specific preprocessing techniques and aligning multilingual datasets. Existing LID systems, however, struggle with multilingual documents, short or noisy text, and resource constraints. This study introduces a lightweight term-frequency-based algorithm for detecting multiple languages in a single document, achieving high accuracy while minimizing memory usage. The algorithm assigns independent relevance scores to each language (not calibrated as proportions), enabling effective identification of both monolingual and multilingual content. The proposed method was benchmarked against state-of-the-art libraries, including BERT, CLD3, fastText, and GlotLID, on WiLI-2018, FLORES+, Tatoeba, OpenSubtitles, and a newly constructed Multilingual Webpages dataset. Results show that the algorithm achieves an F1 score of up to 98%, matching or exceeding existing libraries under comparable evaluation settings while remaining computationally efficient on standard hardware. Its ability to detect dominant languages in multilingual webpages further demonstrates its practical applicability.

Keywords

Language Identification (LID), Multilingual text processing, Natural Language Processing (NLP), Multilingual language detection, Low-resource languages, Code-switching detection, Term frequency methods, Lightweight models, Computational efficiency, Benchmark evaluation

Maxime Sobrier, Saratoga High School, 20300 Herriman Ave, Saratoga, CA, 95070, USA. maxime@sobrier.net

https://doi.org/10.64336/001c.160011

1. Introduction

LID is an essential component of text-processing pipelines used in web classification (1), NLP, machine translation (2, 3), and LLM development. As modern AI systems are trained on billions of web pages (4) collected from large-scale corpora such as Common Crawl (5), accurate language detection plays a crucial role in ensuring that language-specific preprocessors, such as stemming, lemmatization, and stop-word removal, are applied correctly. Errors in this early stage can propagate through subsequent processing steps and degrade the performance of downstream models.

Computational efficiency is also a crucial consideration, as language detection must be performed on all documents during model development, validation, and inference. Maintaining a minimal memory footprint is equally vital, particularly for on-device inference in resource-constrained environments.

Bilingual and multilingual content are prevalent characteristics of modern web pages. The inclusion of cookie banners, consent notices, and legal disclaimers further complicates language detection, as these elements are often displayed in the user's interface language rather than the primary language of the page content. For example, a user located in the United States accessing a French website may encounter disclaimers and interface elements in English, thereby distorting the text distribution and obscuring the page's true linguistic composition. Moreover, variation in web page length poses an additional challenge for language detection systems. Short or fragmentary text segments often provide insufficient linguistic cues, making accurate identification more difficult for conventional algorithms.

Contemporary LID techniques predominantly employ n-gram based models, commonly at the character level using trigrams (i.e., 3-grams). In these approaches, a neural network or statistical model evaluates the frequency distribution of n-letter combinations across languages to estimate a probabilistic score representing the likelihood of a given language, typically ranging from 0 to 1. Traditional machine learning methods, such as Support Vector Machines (SVMs), Naïve Bayes classifiers, and clustering-based algorithms, generally operate at the word level rather than character level. Although these models also produce probabilistic predictions, they are inherently limited in detecting multiple languages in a single text segment, and usually identify only the dominant language. A common heuristic for handling multilingual content is to segment the input text into smaller units and assign each segment its most probable language label. However, this segmentation-based approach substantially reduces overall performance, achieving an F1 score of approximately 61.7% (6), compared to 93.1% or higher reported for n-gram-based models in monolingual settings.

This study introduces a lightweight algorithm designed to detect multiple languages in a single document while matching or exceeding the accuracy, memory efficiency, and processing speed of leading monolingual LID systems. An independent score is assigned to each detected

language. This scoring mechanism provides relative scoring that correlates with language prominence, but is not calibrated as a proportion estimate. A configurable threshold, such as 80% of the cumulative highest scores, can be applied to determine the predominant languages present in the text.

The algorithm, implemented as a JavaScript library, was evaluated against widely used LID libraries such as BERT, CLD3, fastText LID model, and GlotLID. Experiments compared classification F1 score, computational efficiency, including both memory consumption and processing speed, and language coverage across monolingual benchmarks and a multilingual benchmark with a corpus of 48 mixed-language webpages. Results show that the proposed method outperforms existing libraries in accuracy while maintaining comparable or superior efficiency. It demonstrates strong multilingual detection capabilities and broad language generalizability, highlighting its practical applicability to real-world multilingual text processing.

2. Related work

LID has traditionally relied on analyzing term frequencies and comparing the occurrence of standard lexicons across languages with those found in the target text (7). However, this conventional approach often suffers from limited accuracy, particularly when dealing with short or noisy text. To overcome these limitations, researchers have explored more advanced methodologies, including ML algorithms and neural network architectures, to improve both precision and computational efficiency. In recent years, deep learning models have shown considerable promise by leveraging neural networks to capture complex linguistic and contextual patterns. These developments have led to substantial improvements in the accuracy, scalability, and robustness of modern LID frameworks.

Over the past five years, numerous research groups have investigated various methodologies for language detection across diverse document types (8). Babhulgaonkar and Sonavance implemented an SVM approach, achieving an F1 score of 89%, representing an 18% improvement over traditional n-gram-based techniques (9). Their study focused on a dataset comprising three Indian languages: Hindi, Marathi, and Sanskrit. In a separate investigation, Gothe et al. targeted mobile devices, achieving F1 score of 94.5% to 98% for identifying ten languages in multilingual documents using an n-gram model combined with logistic regression (10). While their model was praised for its compactness and computational speed, details regarding its memory footprint were not reported. Meanwhile, Tang et al. employed an n-gram algorithm to achieve a precision of 98.58% for monolingual identification of 61 languages in Internet News articles (11). Half of the target languages in their dataset had fewer than 100 pages. Additionally, Sindane and Marivate conducted a comparative evaluation of twelve n-gram and large pre-trained multilingual models, focusing identifying eleven low-resource languages with an emphasis on model efficiency (12). Their top-performing model achieved an

F1 score of 98.3%. Other studies include BERT (13), CLD3 (14), and GlotLID (15).

A subset of prior research has specifically addressed LID in bilingual documents. However, these studies have been limited to small datasets and a narrow range of languages. Prioleau and Aryal proposed novel LID models tailored for bilingual texts, covering six language pairs, including Spanish-English, Hindi-English, Nepali-English, and Modern Standard Arabic-Egyptian Arabic. Their models achieved an average F1 score of 95.48%, with individual performance ranging from 91.19% to 98.33% (16). Nevertheless, their implementation relied on specialized hardware, including graphics processing units (GPUs) and substantial video random-access memory (VRAM), and the evaluation was limited to four datasets. Ahmad and Singla applied Multinomial Naïve Bayes, Decision Tree, and SVM models for word-level LID in English-Hindi and English-Urdu documents (17). The Naïve Bayes approach, which was closely aligned with the term-frequency-based methods used in the present study, achieved a precision of 80.06% for Hindi-English mixed documents. In comparison, the SVM model demonstrated stronger performance, achieving a precision of 83.58%.

Despite these advances in LID, several research gaps remain. Most notably, the majority of multilingual studies have been conducted on relatively small datasets, often comprising only hundreds of documents and typically covering twelve or fewer languages. Additionally, many existing models rely on specialized hardware and do not prioritize memory-efficient architectures. To address these limitations, the present study proposes a term-frequency-based algorithm with a differential weighting system that outperforms existing ML models in F1 score while supporting the identification of over 100 languages. The proposed algorithm achieves a higher F1 score across standard benchmarks while maintaining a low memory footprint.

3. Methods

3.1 Overview of the algorithm

The proposed algorithm independently scores each language, making it well-suited to multilingual documents. It is based on term frequency analysis, incorporating both word- and character-level frequency metrics. Languages were grouped into two categories: those with discrete word boundaries (e.g., English, French, Hebrew, Vietnamese) and those without such boundaries (e.g., Japanese, Chinese, Korean). For each language, the frequency of each word or character within a designated corpus was computed, yielding frequency scores denoted by f_w and f_c, respectively. Experimental evaluations were conducted on datasets of varying sizes to assess linguistic relevance based on these frequency scores. The top 1,000, 2,000, 5,000, 10,000, and 20,000 most frequent words and characters were analyzed. A relevance score was calculated for each language, and languages achieving at least 0.8 of the maximum score were classified as relevant to the document context.

Over 100 algorithmic variants were systematically evaluated across multiple datasets to optimize the associated parameters, ensuring robust performance across diverse linguistic contexts.

3.2 Word-based scoring

For languages that employ discrete word boundaries, a score was computed for each individual word $S_w = f_w * (l_w - 1)^2$ (eq.1), where, S_w: score of the word, f_w: frequency score of the word and l_w: length of the word.

The final score for each language was computed by scaling the summed word-level scores by the squared frequency of identified words relative to the total word count in the text. This formulation captures both the strength of word-level matches and the relative density of language-relevant words in the text. Only the first two occurrences of any word were counted to prevent repeated terms from dominating the score. $S = \frac{F_w^2}{T_w} * \sum S_w$ (eq.2), where, S: final score, F_w: number of words matched in the frequency list, T_w: number of words in the text and S_w: score of each word.

3.3 Character-based scoring

For languages without discrete word boundaries, the score was derived from character-level frequency analysis. This approach maintains consistency with the scoring methodology applied to word-based languages, ensuring comparability across linguistic structures. Consequently, a document composed of equal proportions of English and Korean—e.g., 50% each—yields ~ similar scores for both languages.

The score was computed based on character frequency in the text. Each character's score was determined by its relative frequency of occurrence: $S = \sum f_c * \frac{F_c}{T_c} * 100$ (eq.3), where, S: final score, f_c: frequency of a character, F_c: number of character occurrences matched in the frequency list and T_c: number of character occurrences in the text.

3.4 Handling multiple writing systems

Different languages employ a variety of writing systems, including Simplified and Traditional Chinese, Tamil and Romanized Tamil, as well as Telugu and Romanized Telugu. Each variant was treated as a separate language within the dataset, and the overall score for a document was computed as the cumulative sum of the individual scores assigned to each variant. $S_{zh} = S_{zhs} + S_{zht}$ (eq.4), where S_{zh}: score for the Chinese language, S_{zhs}: score for simplified Chinese language and S_{zht}: score for traditional Chinese language.

3.5 Construction of frequency datasets

The frequency distributions of words and characters were computed separately for each language. Three datasets were evaluated for this purpose: Wikipedia dumps, news article corpora, and CC-100: a Monolingual Datasets from Web Crawl Data (18). Among these, the Common Crawl-based CC-100 dataset yielded the best performance in web language detection tasks. It supports over 100 languages and aligns well with established linguistic coverage

standards, making it particularly suitable for this study.


```
{
  "af": {
    "topWords": {
      "vanaf": 0.19,
      "die": 0.13,
      "sondeval": 0.5,
      "probeer": 0.48,
      "mens": 0.5,
      "homself": 0.5,
      "deur": 2.4,
      "een": 0.43,
      "of": 0.01,
      "ander": 0.95,
      "instelling": 0.03,
      "as": 0.01,
      "verklaar": 0.5,
```


Figure 1. Sample of the frequency dataset for Afrikaans (af)

The datasets used in this study contained between one and five million words, or ~ two million characters per language. Each word or character was assigned a frequency score corresponding to its number of occurrences per 10,000 tokens. All scores were rounded to two decimal places, with a minimum assigned value of 0.01 per 10,000 occurrences. A sample of the resulting frequency distribution is shown in Figure 1.

Before generating the word and character frequency lists, all textual data underwent a comprehensive preprocessing phase to ensure consistency and data quality. All text was converted to lowercase. Numeric values 0–9, including their equivalents in languages such as Chinese and Korean, were systematically removed. Reserved characters, including emojis and non-printable symbols, were also eliminated. Additionally, Uniform Resource Locators (URLs) were removed, and single-character terms were discarded to further enhance the linguistic integrity of the dataset.

3.6 Differential weighting scheme

To improve differentiation among linguistically similar languages, a differential weighting scheme was applied to the frequency scores of words and characters. Unique words and characters that occur exclusively in a single language had their original scores multiplied by 3, subject to a minimum score of 0.5. Conversely, words and characters appearing in multiple languages were down-weighted, with their original scores multiplied by a factor of 0.8 for each additional occurrence.

English terms frequently appear on non-English websites, often because of technical

terminology, navigation elements, or borrowed expressions. To preserve high precision in English detection, a reduction factor of 0.95^{10} was applied exclusively to words that appear ten times or more in other languages. This adjustment maintains English detection precision above 99%, while causing only a marginal overall precision reduction of ~ 0.05%.

4. Experimental setup

A series of experiments were performed to evaluate the effectiveness of the proposed algorithm against four widely used, high-performance LID libraries: BERT, CLD3, fastText, and GlotLID, which have been extensively benchmarked in recent studies (19-21).

The evaluation was performed using four publicly available benchmark datasets: WiLI-2018, FLORES+, Tatoeba, and OpenSubtitles, which vary in both linguistic diversity and content length, thereby providing a comprehensive baseline for assessing LID precision. In addition, a newly constructed Multilingual Webpages Dataset, comprising 1,845 web pages with multilingual content, was introduced to evaluate the algorithm's performance in real-world web conditions.

4.1 Datasets

4.1.1 *WiLI-2018 dataset*

The WiLI-2018 dataset (22) is a monolingual corpus of concise Wikipedia text excerpts, specifically designed for evaluating LID libraries. It contains 1,000 paragraphs in 235 languages, totaling 235,000 paragraphs of varying lengths. However, the original dataset contains several inconsistencies, such as mislabeled entries, duplicate samples, and paragraphs containing multiple predominant languages. To address these issues, a revised dataset was developed, incorporating 2,440 corrected labels and removing 295 erroneous entries. This corrected version, which is publicly available at https://huggingface.co/datasets/MAximeSobrier/WiLI-2018-corrected , was used in this study.

4.1.2 *FLORES+ dataset*

The FLORES+ dataset (23) is similar to WiLI-2018 in its multilingual structure but differs in its construction methodology. It was developed by translating English sentences sourced from Wikinews, Wikijunior, and Wikivoyage into 200 languages, thereby ensuring broad linguistic diversity. The dataset is balanced, containing 2,011 sentences per language, making it well-suited for evaluating cross-lingual generalization and translation-aligned LID performance.

4.1.3 *Tatoeba sentences dataset*

The Tatoeba Project (24) is a large-scale collaborative initiative led by volunteers to build a comprehensive database of example sentences translated across a wide range of languages. For this study, a subset comprising 10,673,337 sentences spanning 114 languages was used. The sentences vary substantially in length, from concise three-word examples to longer and more complex constructions, with an average length of approximately 35 characters. This variability provides a valuable benchmark

for evaluating LID models across diverse text lengths and structures.

4.1.4 *OpenSubtitles 2024 dataset*

OpenSubtitles (25) is an extensive corpus of translated movie subtitles. For benchmarking purposes, a subset was created containing up to 10,000 samples per language across 81 languages, with each sample consisting of 10 subtitle lines to provide medium-length conversational text that captures informal and idiomatic language patterns.

4.1.5 *Multilingual webpages dataset*

Given the absence of an accurate benchmark for multilingual webpages (26), a new dataset was developed to evaluate real-world multilingual performance. The Multilingual Webpages Dataset (see Data availability) comprises 1,845 webpages containing content in two or more languages. The dataset spans 48 languages, with English (1,705 occurrences), French (1,043 occurrences), and Mandarin (336 occurrences) being the most prevalent. The two dominant languages on each webpage account for most of the content, typically 90% or more, and are of ~ comparable size, providing a robust benchmark for evaluating multilingual detection performance.

4.2 Models and libraries

4.2.1 *BERT*

BERT is a widely recognized LID model. In this study, the xlm-v-base-language-id model, https://huggingface.co/juliensimon/xlm-v-base-language-id, was evaluated. This model was derived from Facebook's XLM-V Base model (13) and was trained on Google's FLEURS dataset (27), which is derived from the FLORES+ benchmark and supports 101 languages.

4.2.2 *Compact Language Detector Version3 (CLD3)*

CLD3, https://github.com/google/cld3, is a LID library developed by Google, and widely used in products such as the Chrome browser. The model employs a neural network that leverages n-gram frequency features and supports 101 languages.

Similar to BERT, fastText, and GlotLID, CLD3 produces a confidence score for each potential language, which is used to rank candidate predictions. In this study, the predicted language was defined as the one with the highest score. These models are designed and trained for single-label classification tasks, and their output scores are not calibrated to represent language proportions within a multilingual document.

4.2.3 *fastText and GlotLID*

fastText (28) is a ML tool developed by Meta. Its LID model was trained on the Wikipedia, Tatoeba, and SETimes datasets and it can recognize 176 languages. Its memory footprint is 126 MB, making it relatively lightweight compared with larger transformer-based models.

GlotLID (15) version 3 is an extension of the fastText approach and supports the identification of 2,102 languages. It was developed using a combination of datasets, including WiLI-2018 and Tatoeba, and was

included in the benchmark due to its stronger performance compared to the original fastText model.

4.2.4 *Language-detector-web (ldw)*

The LID library developed in this study is an open-source project released under the MIT license on GitHub (see Data availability).

To enable meaningful comparisons with existing models, new training datasets were constructed to match the language coverage of each benchmark library while maintaining comparable model sizes, i.e., ldw 20k BERT dataset, ldw 10k CLD3 dataset, and ldw 20k fastText dataset. The Fineweb2 dataset (29) was used to complement the CC-100 dataset. However, certain languages were either missing from both datasets or had insufficient data for effective modeling. Additionally, Fineweb2 did not yield results as robust as those obtained with CC-100.

The library assigns a score to each identified language. For monolingual datasets, the language with the highest score is selected as the predicted language. For multilingual datasets, all languages achieving at least 80% of the maximum score are considered relevant, enabling accurate detection of multiple dominant languages within a single document. The 80% threshold was selected empirically based on experiments conducted on the Multilingual Webpages dataset (Section 5.4), where different threshold values were evaluated to balance precision and recall in multi-language detection. As shown in Table 7, lower thresholds increase recall at the expense of over-identification, while higher thresholds improve precision but may miss relevant languages. A threshold of 0.8 was found to provide a stable trade-off across documents, capturing the dominant languages in most cases while limiting false positives.

Importantly, this threshold is not theoretically derived but represents a practical heuristic calibrated on the evaluation datasets. It can be adjusted depending on the application context—for example, lower thresholds may be preferred in recall-sensitive settings, while higher thresholds may be used when precision is critical.

5. Results

Ldw was evaluated against BERT, CLD3, fastText, and GlotLID on four widely used benchmarks. The evaluation was restricted to single-language inputs to ensure comparability with existing LID models. For each comparison, datasets were aligned with the language coverage supported by the evaluated model.

The macro- and micro-averaged precision, recall, F1 score, and accuracy were calculated along with 95% confidence intervals. Statistical significance between paired systems was assessed using McNemar's test. In addition, paired bootstrap resampling with 2,000 iterations was used to estimate confidence intervals for differences in macro accuracy. In each bootstrap iteration, 100 languages were sampled with replacement, and the mean difference in macro accuracy was computed; intervals excluding zero were interpreted as being statistically significant. McNemar's test

evaluates paired instance-level classification differences, while bootstrap resampling assesses variability across languages. Both metrics were reported to capture complementary sources of uncertainty.

5.1 Efficiency and scalability

Table 1 compares the evaluated libraries based on model size, inference time, and language coverage.

Table 1. Computational efficiency and resource usage of the evaluated LID systems. Inference time was measured per input on an AMD Ryzen 9 3950X (4.1 GHz, 128 GB RAM). Memory footprint refers to the total model and data size required at inference time.

Library	Languages Supported	Data Size	Computation Time
ldw 1k (CC-100 dataset)	106	1.6 MB	0.22 ms
ldw 2k (CC-100 dataset)	106	3.5 MB	0.24 ms
ldw 5k (CC-100 dataset)	106	8.3 MB	0.27 ms
ldw 10k (CC-100 dataset)	106	22 MB	0.31 ms
ldw 20k (CC-100 dataset)	106	34 MB	0.37 ms
BERT	101	3.0 GB	8.81 ms
ldw 20k (BERT dataset)	97	32 MB	0.39 ms
CLD3	101	10 MB	0.22 ms
ldw 10k (CLD3 dataset)	101	20 MB	0.31 ms
fastText	176	126 MB	0.37 ms
GlotLID	2,102	1.6 GB	0.40 ms
ldw 20k (fastText dataset)	163	53 MB	0.58 ms

BERT exhibited the highest computational cost, with inference times ~ 25 times slower than other systems, despite truncation to 512 tokens, and a memory footprint of approximately 3 GB. GlotLID similarly required over 1.6 GB of memory to support its large language inventory.

In contrast, ldw maintained consistently low inference latency across different frequency-list sizes, ranging from 0.22 ms for ldw 1k to 0.37 ms for ldw 20k, while requiring at most 53 MB even when aligned with fastText's language coverage. These results indicate that ldw offers a favorable balance among efficiency, memory usage, and multilingual coverage, making it well-suited for both large-scale deployments and resource-constrained environments.

5.2 Overall performance across benchmarks

Table 2 reports precision, recall, and F1 score for ldw across increasing frequency-list sizes on the four benchmarks. Macro-averaged metrics were emphasized because they better reflect performance on low-resource and typologically similar languages. Performance generally improved with richer frequency information, but diminishing returns set in beyond 10k entries.

WiLI and FLORES+ emphasize clean text and broad language coverage. Ldw 20k BERT-

aligned achieved strong macro-averaged performance, reaching a macro F1 98.90% on WiLI and 99.14% on FLORES+. These results indicate that ldw's support for multilingual detection does not degrade its performance in a single-language setting.

Performance on Tatoeba, which contains shorter sentences, remained lower in macro-averaged metrics across all models, reflecting known challenges of sentence-level LID. Nevertheless, ldw demonstrated consistent improvements as the frequency list increased, with macro F1 increasing from 62.30% to 71.93%. On OpenSubtitles, performance was comparatively stable across configurations, with macro F1 exceeding 91% for ldw 20k.

Table 2. Performance of the ldw library across benchmarks as a function of frequency-list size. Macro- and micro-averaged precision, recall, and F1 score are reported for ldw trained on Common Crawl frequency lists of increasing size (1k–20k). Evaluation was restricted to single-language inputs. All metrics are reported as percentages.

Dataset	Metric	Averaging	ldw 1k	ldw 2k	ldw 5k	ldw 10k	ldw 20k
WiLI	Precision	Macro	98.04	98.20	98.54	98.73	98.94
		Micro	97.88	98.34	98.73	98.86	98.96
	Recall	Macro	97.93	98.33	98.66	98.79	98.92
		Micro	97.84	98.32	98.72	98.86	98.95
	F1 score	Macro	97.64	98.05	98.49	98.70	98.90
		Micro	97.86	98.33	98.72	98.86	98.95
FLORES+	Precision	Macro	98.64	98.90	99.35	99.50	99.61
		Micro	97.23	97.74	98.31	98.47	98.59
	Recall	Macro	96.97	97.66	98.42	98.69	98.85
		Micro	96.60	97.29	98.03	98.28	98.44
	F1 score	Macro	97.56	98.11	98.77	98.99	99.14
		Micro	96.91	97.51	98.17	98.38	98.52
Tatoeba	Precision	Macro	62.21	62.45	66.11	68.37	69.52
		Micro	89.53	90.44	92.54	93.94	94.84
	Recall	Macro	80.14	82.89	86.73	88.92	90.26
		Micro	87.11	88.70	91.43	93.17	94.29
	F1 score	Macro	62.30	63.40	67.59	70.28	71.93
		Micro	88.30	89.56	91.98	93.55	94.56
OpenSubtitles	Precision	Macro	94.40	94.37	94.59	94.51	94.22
		Micro	94.42	94.10	94.27	94.30	94.34
	Recall	Macro	91.59	91.99	92.55	92.83	92.94
		Micro	92.28	92.69	93.27	93.62	94.84
	F1 score	Macro	90.49	90.55	90.95	90.97	91.03
		Micro	93.34	93.39	93.77	93.96	94.09

5.3 Comparison by baseline family

To enable fair comparison, ldw was evaluated against each baseline under matched language coverage. Tables 3–5 report precision, recall, and F1 score for these aligned evaluations.

Table 3. Precision comparison across LID systems under matched language coverage. All values are reported as percentages. Macro- and micro-averaged precision are reported for each benchmark after restricting evaluation to the common set of languages. For each baseline–ldw pair, 95% confidence intervals were estimated using paired bootstrap resampling. McNemar's test reports the number of instances misclassified by one system but not the other (n01, n10) and the associated χ^2 statistic; statistically significant differences are indicated by $p < 0.05$.

Benchmark	Type		BERT	ldw20k BERT	CLD3	ldw10k CLD3	fastText	GlotLID	ldw20k fastText
WiLI	Macro		97.95	98.73	96.52	97.80	90.48	96.76	97.01
	Micro		98.06	98.79	96.27	97.79	79.35	98.93	96.75
	95% CI		97.01-99.14	97.81 - 99.37	94.84 - 97.91	96.24 - 99.01	87.04 - 93.72	92.35 - 99.14	95.02 - 98.61
	McNemar	n01	620		1,798		14,875	1,508	
		n10	319		720		1,643	897	
		χ^2	95.85		460.65		10,598	154.72	
FLORES+	Macro		98.80	98.55	95.74	98.31	87.14	99.23	99.05
	Micro		98.60	98.42	91.98	97.31	82.41	99.13	97.66
	95% CI		97.83 - 99.79	97.96 - 99.06	94.79 - 96.81	96.79 - 99.39	82.78 - 91.33	98.67 - 99.67	98.46 - 99.54
	McNemar	n01	451		6,584		31,128	3,973	
		n10	1,324		2,965		2,345	4,862	
		χ^2	428.39		1,370.82		24,748	89.25	
Tatoeba	Macro		69.88	68.15	76.12	64.18	54.48	57.18	55.78
	Micro		96.47	94.34	87.58	93.41	93.77	97.03	85.04
	95% CI		63.54 - 78.34	60.53 - 75.61	67.23 - 84.31	56.64 - 71.69	49.19 - 61.41	50.15 - 63.68	48.96 - 61.72
	McNemar	n01	180,673		1,049,571		500,554	466,871	
		n10	449,866		441,178		1,522,466	1,531,229	
		χ^2	114,924		248,292		516,209.47	566,967	
Open Subtitles	Macro		94.79	94.00	92.80	94.14	90.47	92.18	95.25
	Micro		92.70	95.13	93.24	93.43	86.87	96.69	92.69
	95% CI		91.23 - 98.75	89.55 - 97.66	89.16 - 96.06	90.38 - 97.17	87.29 - 94.57	86.88 - 96.67	91.09 - 98.45
	McNemar	n01	12,301		12,252		48,366	32,128	
		n10	5,058		14,933		20,927	17,370	
		χ^2	3,021		264.20		10,865	4,399.55	

Within the same language set, ldw 20k consistently matched or exceeded BERT across all benchmarks in macro-averaged metrics. Gains were most pronounced on WiLI, where macro F1 increased from 98.07% to 98.73% (Table 5), and on OpenSubtitles, where ldw achieved higher recall without loss of precision (Tables 3 and 4). McNemar's tests indicated that these differences corresponded to statistically significant reductions in error counts. BERT's strong performance on FLORES+ was expected, as this benchmark overlaps with its pretraining data; therefore, results on FLORES+ should be interpreted with this overlap in mind.

Table 4. Recall comparison across LID systems under matched language coverage. Macro- and micro-averaged recall are reported under the same evaluation protocol as Table 3. Confidence intervals were computed from paired predictions for each benchmark.

Benchmark	Type	BERT	ldw20k BERT	CLD3	ldw10k CLD3	fastText	GlotLID	ldw20k fastText
WiLI	Macro	98.35	98.83	96.15	97.88	81.01	95.96	96.94
	Micro	98.06	98.79	96.18	97.79	79.35	96.77	96.75
	95% CI	97.91 - 99.12	98.52 - 99.11	94.29 - 97.67	96.67 - 98.81	77.30 - 87.82	91.56 - 98.41	94.91 - 98.24
FLORES+	Macro	99.03	98.69	91.53	97.44	82.31	98.11	97.84
	Micro	98.60	98.27	91.52	97.12	82.41	97.94	97.52
	95% CI	97.70 - 99.95	97.23 - 99.61	89.09 - 93.71	95.47 - 98.99	77.91 - 89.29	97.01 - 99.03	96.47 - 98.85
Tatoeba	Macro	87.87	89.03	86.06	88.08	55.63	62.92	82.95
	Micro	96.47	93.77	86.60	92.66	93.77	94.26	84.53
	95% CI	85.46 - 91.49	86.65 - 91.34	81.71 - 89.90	85.15 - 90.61	50.75 - 62.96	56.66 - 69.17	80.43 - 85.20
Open Subtitles	Macro	91.72	93.50	92.30	91.97	85.87	88.32	91.07
	Micro	92.70	94.60	93.13	92.71	86.87	90.02	92.24
	95% CI	87.11 - 96.10	88.99 - 97.35	87.31 - 96.15	86.69 - 96.20	81.47 - 91.71	82.29 - 93.53	86.06 - 95.36

Relative to CLD3, ldw 10k yielded substantial improvements on WiLI, FLORES+, and OpenSubtitles. On WiLI, macro F1 increased from 96.14% to 97.77%, and on FLORES+ from 93.41% to 97.72% (Table 5). These improvements were consistently supported by McNemar's tests (Table 3).

fastText exhibited lower macro-averaged performance on WiLI and FLORES+, particularly in recall for closely related or low-resource languages. fastText-aligned ldw 20k improved both precision and recall (Tables 3–5). GlotLID achieved strong performance on FLORES+, but required substantially more memory, whereas ldw achieved comparable

macro F1 with significantly lower resource requirements.

Table 5. F1 score comparison across LID systems under matched language coverage. Macro- and micro-averaged F1 scores are reported for each benchmark and were derived from the precision and recall reported in Tables 3 and 4.

Benchmark	Type	BERT	ldw20k BERT	CLD3	ldw10k CLD3	fastText	GlotLID	ldw20k fastText
WiLI	Macro	98.07	98.73	96.14	97.77	79.68	96.33	96.20
	Micro	98.06	98.79	96.23	97.79	79.35	97.84	96.75
	95% CI	97.46 - 99.00	98.16 - 99.13	94.58 - 97.48	96.48 - 98.80	76.25 - 85.76	92.00 - 98.69	93.90 - 98.01
FLORES+	Macro	98.76	98.49	93.41	97.72	80.38	98.57	98.31
	Micro	98.60	98.35	91.75	97.21	82.41	98.53	97.59
	95% CI	97.81 - 99.74	97.55 - 99.17	91.71 - 94.97	96.03 - 99.04	76.40 - 86.86	97.90 - 99.14	97.35 - 99.04
Tatoeba	Macro	72.71	70.27	77.46	66.23	51.82	57.20	57.85
	Micro	96.47	94.05	87.09	93.03	93.77	95.62	84.78
	95% CI	66.94 - 80.34	63.41 - 76.86	69.47 - 84.54	59.15 - 73.04	46.60 - 58.93	50.56 - 63.65	51.54 - 63.35
Open Subtitles	Macro	91.62	91.47	91.08	91.01	85.78	89.24	90.93
	Micro	92.70	94.86	93.18	93.07	86.87	93.23	92.46
	95% CI	87.25 - 96.11	86.47 - 95.86	86.23 - 94.95	85.95 - 95.07	81.99 - 91.12	83.58 - 94.16	86.13 - 95.09

5.4 Multilingual webpages

The Multilingual Webpages dataset benchmark demonstrates that the ldw library effectively identifies multiple languages in real-world web content, as shown in Table 6. Performance improved substantially as the frequency-list size increased: macro F1 increased from 69.14% for ldw 1k to 83.37% for ldw 20k, and accuracy exceeded 98% for ldw 10k and ldw 20k. Accuracy was computed for each detected language per document using exact-match evaluation. These results demonstrate that richer frequency information is critical for detecting multiple languages within a single document.

As shown in Table 7, a threshold of 0.8 provided a practical balance between precision and recall, while lower thresholds increased recall at the cost of over-identification. This trade-off allows ldw to be tuned for different application scenarios.

Table 6. Performance of ldw on the Multilingual Webpages dataset. Macro- and micro- averaged precision, recall, accuracy, and F1 score are reported. Results evaluate ldw's ability to detect multiple languages within a single webpage. All metrics are reported as percentages.

Dataset	Averaging	Precision	Recall	Accuracy	F1 score
dw 1k	Macro	88.96	63.80	97.46	69.14
	Micro	96.66	57.43	99.15	72.05
ldw 2k	Macro	90.22	68.99	97.95	73.37
	Micro	97.34	60.72	99.22	74.79
ldw 5k	Macro	91.78	70.97	97.96	76.50
	Micro	97.71	65.02	99.30	78.08
ldw 10k	Macro	88.51	76.12	98.20	78.23
	Micro	97.45	63.30	99.27	76.75
ldw 20k	Macro	87.65	84.36	98.33	83.37
	Micro	97.02	62.95	99.25	76.36

Table 7. Effect of the decision threshold on multilingual language detection with ldw 20k. Performance on the Multilingual Webpages dataset is reported for different confidence thresholds. Lower thresholds favor recall, while higher thresholds improve precision. All metrics are reported as percentages.

Threshold	Averaging	Precision	Recall	Accuracy	F1 score
0.6	Macro	81.54	88.13	99.02	82.43
	Micro	96.16	79.58	99.55	87.09
0.7	Macro	82.22	85.44	98.66	81.42
	Micro	96.27	70.68	99.38	81.52
0.8	Macro	83.50	80.80	98.31	79.20
	Micro	96.50	61.95	99.23	75.46
0.9	Macro	79.88	73.32	98.01	72.70
	Micro	96.55	54.37	99.09	69.56

5.5 Ablation study

Table 8 presents an ablation study examining the effects of differential weighting and English bias on precision, recall, and F1 score. The Multilingual Websites dataset was used to identify the top language in these tests. Across WiLI and FLORES+, differences between configurations remained below 1% in aggregate metrics, indicating overall robustness. On the Multilingual Webpages dataset, differential weighting produced the largest improvement, while adding English bias provided a further gain in precision for documents that included English.

Table 8. Metrics under ablation settings for ldw 20k. Three configurations are compared: no differential weighting, differential weighting only, and differential weighting with English bias. Results are reported for WiLI, FLORES+, and the Multilingual Webpages dataset. All metrics are reported as percentages.

Dataset	Metric	Averaging	WiLI	FLORES+	Multilingual
No Differential Weighting	Precision	Macro	98.19	98.41	48.39
		Micro	97.80	97.39	95.81
	Recall	Macro	97.79	97.55	36.73
		Micro	97.80	97.25	64.72
	F1 score	Macro	97.93	97.77	37.57
		Micro	97.80	97.32	77.25
Differential Weighting	Precision	Macro	97.85	98.45	82.49
		Micro	97.78	97.47	95.26
	Recall	Macro	98.10	97.63	87.01
		Micro	97.78	97.33	50.03
	F1 score	Macro	97.93	97.87	81.85
		Micro	97.78	97.40	65.80
Differential Weighting with English Bias	Precision	Macro	98.18	98.44	83.50
		Micro	98.05	97.47	96.50
	Recall	Macro	98.12	97.63	80.80
		Micro	98.05	97.32	61.95
	F1 score	Macro	98.11	97.87	79.20
		Micro	98.05	97.40	75.46

However, per-language analysis revealed meaningful effects for closely related languages. As shown in Table 9, differential weighting substantially improved precision for Indonesian (+11%) while slightly reducing precision for Malay (−0.73%), together with a 16% increase in Malay recall. The English bias further stabilized English performance, maintaining an F1 score near 98% and increasing recall to 99.16%. These ablations primarily affected a small subset of closely related languages and did not materially change aggregate performance.

Table 10 reports the effect of the exponent used in the word-scoring function by comparing word length to its square, cube, and fourth power. Among the four settings, squaring the word length yields the best overall performance on the Multilingual Webpages benchmark. In particular, the multilingual macro F1 score increases from 59.31% with exponent 1 to 79.20% with exponent 2, while higher exponents reduce performance. By contrast, the impact on the monolingual benchmarks (WiLI, FLORES+, Tatoeba, and OpenSubtitles) is comparatively small. These results suggest that squaring the word length provided a better balance between emphasizing more informative longer words and avoiding excessive weighting of long tokens.

Table 9. Per-language metrics for closely related languages in the corrected WiLI 2018 dataset. Results highlight the effect of differential weighting and English bias on language pairs known to be difficult to distinguish. All metrics are reported as percentages.

Dataset	Metric	Bosnian	Croatian	Indonesian	Malay	English
No Differential Weighting	Precision	63.22	59.48	77.66	98.43	95.73
	Recall	54.91	70.07	98.96	74.20	98.94
	F1 score	58.77	64.34	87.02	84.62	97.31
Differential Weighting	Precision	65.17	60.27	88.96	97.57	99.28
	Recall	55.15	73.16	97.92	90.41	89.32
	F1 score	59.74	66.09	93.22	93.86	94.04
Differential Weighting with English Bias	Precision	65.36	60.26	88.94	97.70	96.88
	Recall	55.15	72.92	97.69	90.41	99.16
	F1 score	59.82	65.99	93.11	93.92	98.01

Table 10. Ablation results for ldw 20k under different exponents in the word-scoring function: word length to the powers of 1, 2, 3, and 4. Results are reported for WiLI, FLORES+, Tatoeba, OpenSubtitles, and Multilingual Webpages dataset. All metrics are reported as percentages.

Power	Metric	Type	WiLI	FLORES+	Tatoeba	OpenSubtitles	Multilingual
1	Precision	Macro	97.10	98.04	69.33	93.21	77.29
		Micro	90.71	92.87	87.31	86.77	96.75
	Recall	Macro	91.81	92.65	84.91	86.58	55.51
		Micro	90.71	92.73	86.91	86.29	57.02
	F1 score	Macro	91.33	92.97	67.89	84.34	59.31
		Micro	90.71	92.80	87.06	86.53	71.75
2	Precision	Macro	98.94	99.61	69.52	94.22	83.50
		Micro	98.96	98.59	94.84	94.34	96.50
	Recall	Macro	98.92	98.85	90.26	92.94	80.80
		Micro	98.95	98.44	94.29	94.84	61.95
	F1 score	Macro	98.90	99.14	71.93	91.03	79.20
		Micro	98.95	98.52	94.56	94.09	75.46
3	Precision	Macro	99.05	99.60	68.61	94.19	70.15
		Micro	98.91	98.57	95.75	94.33	95.63
	Recall	Macro	98.77	98.77	91.94	92.94	70.27
		Micro	98.90	98.90	95.19	93.83	53.87
	F1 score	Macro	98.89	98.89	71.81	91.05	67.37
		Micro	98.90	98.90	95.47	94.08	68.92
4	Precision	Macro	98.93	99.52	67.76	93.96	65.42
		Micro	98.72	98.48	95.81	94.30	95.46
	Recall	Macro	98.39	98.74	91.35	92.92	66.61
		Micro	98.72	98.32	95.26	93.80	52.42
	F1 score	Macro	98.62	99.03	71.24	91.03	63.29
		Micro	98.72	98.40	95.54	94.05	67.68

Table 11 shows the effect of limiting the number of times a repeated word can contribute to the document score. The impact on the monolingual benchmarks is small, but this parameter had a much larger effect on the Multilingual Webpages dataset. In particular, applying an occurrence cap substantially improved multilingual performance compared with counting all occurrences of the same word. The highest micro F1 score is obtained when each word is counted up to three times (75.99%), while counting each word only once or twice also performs better than using no limit. For the macro F1 score, the best result is achieved with a cap of one occurrence (80.21%), compared with 59.71% when no limit is applied. These results indicate that restricting repeated occurrences helps reduce the influence of duplicated or boilerplate text in multilingual documents.

Table 11. Ablation results for ldw 20k under different limits on repeated word counting in a document: once, twice, three times, and no limit. Results are reported for WiLI, FLORES+, Tatoeba, OpenSubtitles, and Multilingual Webpages dataset. All metrics are reported as percentages.

Occurrences	Metric	Type	WiLI	FLORES+	Tatoeba	OpenSubtitles	Multilingual
1	Precision	Macro	98.91	99.61	69.53	94.11	84.49
		Micro	98.96	98.59	94.83	94.36	96.45
	Recall	Macro	98.90	98.85	90.19	92.94	80.40
		Micro	98.96	98.44	94.28	93.85	58.83
	F1 score	Macro	98.87	99.14	71.93	91.02	80.21
		Micro	98.96	98.52	91.56	94.10	73.08
2	Precision	Macro	98.94	99.61	69.52	94.22	83.50
		Micro	98.96	98.59	94.84	94.34	96.50
	Recall	Macro	98.92	98.85	90.26	92.94	80.80
		Micro	98.95	98.44	94.29	94.84	61.95
	F1 score	Macro	98.90	99.14	71.93	91.03	79.20
		Micro	98.95	98.52	94.56	94.09	75.46
3	Precision	Macro	98.93	99.61	69.51	94.24	76.62
		Micro	98.94	98.59	94.84	94.34	96.26
	Recall	Macro	98.90	98.85	90.26	92.94	72.45
		Micro	98.94	98.44	94.29	93.83	62.77
	F1 score	Macro	98.89	99.14	71.92	91.04	71.13
		Micro	98.94	98.51	94.56	94.08	75.99
No limit	Precision	Macro	98.93	99.61	69.51	94.18	72.15
		Micro	98.93	98.59	94.84	94.32	96.89
	Recall	Macro	98.89	98.84	90.26	92.93	58.50
		Micro	98.93	98.43	94.29	93.82	62.22
	F1 score	Macro	98.88	99.13	71.92	91.01	59.71
		Micro	98.93	98.51	94.56	94.07	75.78

5.6 Summary

Across standard monolingual benchmarks, *ldw* achieves strong performance comparable to existing methods while maintaining high efficiency and low memory usage. Notably, its ability to detect multiple languages within a single input does not appear to compromise monolingual accuracy under the evaluated settings and, in some cases, yields statistically significant improvements over selected baselines. These results suggest that *ldw* is a practical and extensible approach for both traditional language identification tasks and real-world multilingual web content, within the scope of the current evaluation framework.

6. Limitations

Each evaluated model supports a distinct set of languages and was trained on different datasets, some of which overlap with the benchmarks used for evaluation. Retraining these models on a consistent set of languages using a uniform training dataset would enable a more rigorous and controlled assessment of their comparative effectiveness.

Comparisons with baseline models are constrained by differences in training data, architecture, and task formulation, particularly between single-label and multi-label LID approaches. Many of the benchmarks used in this study are unbalanced and do not cover all the languages supported by the models. Therefore, improvements against these benchmarks cannot be construed as a fair comparison. Developing balanced benchmarks that include all supported languages would allow for more accurate and equitable comparisons of LID capabilities.

This study evaluated the proposed algorithm using 110 languages. Expanding beyond CC-100 to include additional high- and low-resource languages is necessary to fully evaluate the algorithm's generalizability and robustness across the full spectrum of linguistic diversity. The new model was not evaluated on low-resource languages, subtle intra-document language mixing, or code switching. The model relies on frequency distributions derived from web-based corpora. Its performance under domain shift (e.g., technical, medical, or legal text) was not systematically evaluated. Additionally, the robustness of the model to noisy input, extremely short text, or OCR artifacts was not systematically evaluated.

The scoring mechanism is not calibrated to represent true language proportions. While relative scores correlate with language prominence, they should not be interpreted as precise estimates of content distribution without further calibration.

Extensive parameter tuning was conducted across multiple datasets, which may introduce a risk of overfitting. Future work should include evaluation on strictly held-out datasets and cross-domain validation. The model relies on empirically derived heuristics, including weighting schemes and scoring functions, which may not generalize optimally across all languages or datasets.

Character-based scoring may bias detection toward writing systems rather than linguistic identity, potentially limiting performance for languages sharing scripts.

These limitations highlight opportunities for future work, including the development of unified training pipelines, expanded and more balanced dataset construction, and the use of controlled evaluation frameworks with properly aligned and comparable baseline models. Further improvements may also be achieved through the integration of supplementary linguistic features to enhance differentiation among closely related languages.

7. Conclusion

A new library, *ldw*, has been developed to identify over 100 languages in multilingual web pages, demonstrating strong performance relative to existing machine learning and frequency-based approaches under the evaluated conditions. The method achieves an overall precision of up to 98.96% across the tested benchmarks, with low variability across languages. Its ability to assign independent relevance scores to each language enables effective identification of dominant languages in multilingual webpages, although these scores are not calibrated to represent exact proportions. Across standard benchmarks, *ldw* matches or exceeds the performance of existing methods in comparable single-label evaluation settings, while maintaining a smaller memory footprint and faster inference times. These results suggest that *ldw* provides a practical and efficient approach for large-scale and resource-constrained language identification tasks, particularly in scenarios involving multilingual content.

Acknowledgments

The author would like to thank Dr. Chen Huang at the Institute of Data Science (IDS), National University of Singapore for his fruitful discussion and reviews of the paper.

Data availability

All datasets, benchmarks, and evaluation scripts are publicly available at https://github.com/MaximeSobrier/language-detector-research , enabling full reproducibility and per-language analysis. The Multilingual Webpages Dataset is available at https://huggingface.co/datasets/MAximeSobrier/Web-multilingual

8. References

1. Apandi, S. H., Sallim, J., & Mohamed, R. (2021). Data pre-processing of website browsing record: An initial step for web page classification. In Proceedings of the 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM) (pp. 679–684). https://doi.org/10.1109/ICSECS52883.2021.00129

2. Lyu, C., Du, Z., Xu, J., Duan, Y., Wu, M., Lynn, T., et al. (2024). A paradigm shift: The future of machine translation lies with large language models. arXiv. https://arxiv.org/pdf/2305.01181

3. Chen, L., Wang, W., & Hu, D. (2024). Optimizing language model training for translation via enhancing efficiency and effectiveness. In Proceedings of the 23rd China National Conference on Computational Linguistics (pp. 1023–1034). https://aclanthology.org/2024.ccl-1.79.pdf

4. Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Cappelli, A., Alobeidli, H., et al. (2023). The refined web dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. arXiv. https://arxiv.org/pdf/2306.01116

5. Su, D., Kong, K., Lin, Y., Jennings, J., Norick, B., Kliegl, M., et al. (2025). Nemotron-CC: Transforming Common Crawl into a refined long-horizon pretraining dataset. arXiv. https://arxiv.org/pdf/2412.02595

6. Frohmann, M., Sterner, I., Vulić, I., Minixhofer, B., & Schedl, M. (2024). Segment any text: A universal approach for robust, efficient and adaptable sentence segmentation. arXiv. https://arxiv.org/pdf/2406.16678

7. Dodiya, T., & Reha, A. Y. (2021). Using term frequency–inverse document frequency to find the relevance of words in Gujarati language. International Journal for Research in Applied Science and Engineering Technology, 9, 378–381. https://doi.org/10.22214/ijraset.2021.33625

8. Hidayatullah, A. F., Qazi, A., Lai, D. T. C., & Apong, R. A. (2022). A systematic review on language identification of code-mixed text: Techniques, data availability, challenges, and framework development. IEEE Access, 122812–122831. https://ieeexplore.ieee.org/document/9956817

9. Babhulgaonkar, A., & Sonavane, S. (2020). Language identification for multilingual machine translation. In Proceedings of the 2020 International Conference on Communication and Signal Processing (ICCSP) (pp. 401–405). https://ieeexplore.ieee.org/document/9182184

10. Gothe, S. V., Ghosh, S., Mani, S., Bhanodai, G., Agarwal, A., & Sanchi, C. (2020). Language detection engine for multilingual texting on mobile devices. In Proceedings of the 2020 IEEE 14th International Conference on Semantic Computing (ICSC) (pp. 279–286). https://doi.org/10.1109/ICSC.2020.00057

11. Tang, J., Chen, X., & Liu, W. (2021). Efficient language identification for all-language internet news. In Proceedings of the 2021 International Conference on Asian Language Processing (IALP) (pp. 165–169). https://doi.org/10.1109/IALP54817.2021.9675270

12. Sindane, T., & Marivate, V. (2024). From n-grams to pre-trained multilingual models for language identification. arXiv. https://arxiv.org/pdf/2410.08728

13. Liang, D., Gonen, H., Mao, Y., Hou, R., Goyal, N., Ghazvininejad, M., et al. (2023). XLM-V: Overcoming the vocabulary bottleneck in multilingual masked language models. arXiv. https://arxiv.org/pdf/2301.10472

14. Salcianu, A., Golding, A., Bakalov, A., Alberti, C., Andor, D., Weiss, D., et al. (2018). Compact language detector v3. https://chromium.googlesource.com/external/github.com/google/cld_3/+/f01672272dacc4cb3409f458ed61f7d4eb0f47de/README.md

15. Kargaran, A. H., Imani, A., Yvon, F., & Schütze, H. (2023). GlotLID: Language identification for low-resource languages. In Findings of the Association for Computational Linguistics: EMNLP 2023 (pp. 6155–6218). Association for Computational Linguistics. https://aclanthology.org/2023.findings-emnlp.410.pdf

16. Prioleau, H., & Aryal, S. K. (2023). Benchmarking current state-of-the-art transformer models on token-level language identification and language pair identification. In Proceedings of the 2023 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 193–199). https://doi.org/10.1109/CSCI62032.2023.00036

17. Ahmad, G. I., & Singla, J. (2022). Machine learning approach towards language identification of code-mixed Hindi-English and Urdu-English social media text. In Proceedings of the 2022 International Mobile and Embedded Technology Conference (MECON) (pp. 215–220). https://doi.org/10.1109/MECON53876.2022.9751958

18. Wenzek, G., Lachaux, M.-A., Conneau, A., Chaudhary, V., Guzmán, F., Joulin, A., & Grave, E. (2020). CCNet: Extracting high quality monolingual datasets from web crawl data. In Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020) (pp. 4003–4012). https://aclanthology.org/2020.lrec-1.494.pdf

19. van der Goot, R. (2025). Identifying open challenges in language identification. In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (pp. 18207–18227). Association for Computational Linguistics. https://aclanthology.org/2025.acl-long.891.pdf

20. Agarwal, M., Alam, M. M. I., & Anastasopoulos, A. (2023). LIMIT: Language identification, misidentification, and translation using hierarchical models in 350+ languages. In Proceedings of

the 2023 Conference on Empirical Methods in Natural Language Processing (pp. 14496–14519). Association for Computational Linguistics. https://aclanthology.org/2023.emnlp-main.895.pdf

21. Burchell, L., Birch, A., Bogoychev, N., & Heafield, K. (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) (pp. 865–879). Association for Computational Linguistics. https://aclanthology.org/2023.acl-short.75.pdf

22. Thoma, M. (2018). The WiLI benchmark dataset for written language identification. arXiv. https://arxiv.org/pdf/1801.07779

23. NLLB Team. (2024). Scaling neural machine translation to 200 languages. Nature, 630, 841–846. https://doi.org/10.1038/s41586-024-07335-x

24. Tiedemann, J. (2020). The Tatoeba translation challenge: Realistic data sets for low-resource and multilingual MT. In Proceedings of the 5th Conference on Machine Translation (WMT) (pp. 1174–1182). Association for Computational Linguistics. https://aclanthology.org/2020.wmt-1.139.pdf

25. Lison, P., Tiedemann, J., & Kouylekov, M. (2018). OpenSubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA). https://aclanthology.org/L18-1275.pdf

26. Kreutzer, J., Caswell, I., Wang, L., Wahab, A., van Esch, D., Ulzii-Orshikh, N., et al. (2022). Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics, 10*, 50–72. https://aclanthology.org/2022.tacl-1.4.pdf

27. Conneau, A., Ma, M., Khanuja, S., Zhang, Y., Axelrod, V., Dalmia, S., et al. (2022). *FLEURS: Few-shot learning evaluation of universal representations of speech*. arXiv. https://arxiv.org/pdf/2205.12446

28. Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). *Bag of tricks for efficient text classification*. arXiv. https://arxiv.org/pdf/1607.01759

29. Penedo, G., Kydlíček, H., Sabolcec, V., Messmer, B., Foroutan, N., Kargaran, A. H., et al. (2025). *FineWeb2: One pipeline to scale them all – Adapting pre-training data processing to every language*. arXiv. https://arxiv.org/pdf/2506.20920