Impact of Sentence Length on Multilingual Language Identification with GlotLID and OpenLID

Jort Koks
ENSAE Paris
jort.koks@ensae.fr

Abstract

Language identification (LID) is a critical preprocessing step for multilingual NLP, yet performance can vary greatly with input sentence length. This paper presents an empirical analysis of how sentence length affects LID accuracy and other metrics, focusing on differences between high-resource and low-resource languages. We compare two state-of-the-art open models: **GlotLID**, trained on the wide-coverage GlotLID-C corpus (1665 languages), and **OpenLID**, a FastText model covering 201 languages. Using a controlled evaluation with sentences truncated to fixed length bins, we show that short inputs (e.g., <20 characters) dramatically degrade identification performance, especially for low-resource languages. OpenLID achieves higher recall on very short texts for high-resource languages, whereas GlotLID maintains lower false positive rates. For longer inputs (>=40 characters), both models approach near-perfect accuracy for most languages. Our analysis highlights the importance of input length in LID evaluation and the need for robust strategies to handle extremely short texts for low-resource languages.

1 Introduction

Multilingual language identification (LID) is the task of automatically determining the language of a given text. Accurate LID is fundamental for multilingual NLP pipelines, filtering corpora coming from the internet, and enabling downstream tasks like machine translation to select the appropriate model. Recent advances have greatly expanded the coverage of LID systems to hundreds or even thousands of languages. However, a persistent challenge in LID is handling short texts. It is well known that very short snippets (e.g., a single word or a few characters) can be ambiguous and hard to classify, especially when distinguishing closely related languages or when dealing with low-resource languages that lack distinctive cues. Critically, the difficulty of LID on short inputs may not be uniform across languages. High-resource languages (with abundant training data) often have well-learned lexical cues, even at the character level, enabling decent identification from a brief phrase. In contrast, low-resource languages might require more context for reliable identification. To date, there has been limited systematic analysis quantifying how LID performance varies with input length, particularly comparing effects on high-resource vs. low-resource languages. This gap is important: if LID models struggle disproportionately on short samples of low-resource languages, it could bias the collection of those languages in web corpora and impede downstream NLP for those communities. In this work, we address this gap with an empirical study of sentence length impact on multilingual LID. We evaluate two recent open LID models representative of state-of-the-art approaches: GlotLID [2], which emphasizes broad coverage of low-resource languages (1665 languages) with a focus on reliability (balanced F1 and false positive rate), and OpenLID [1], which provides high accuracy on 201 high-resource languages using curated training data. We construct a controlled evaluation dataset by sampling and truncating sentences to fixed length ranges, ensuring a fair comparison across

Submitted as part of the evaluation for the course Machine Learning for Natural Language Processing, academic year 2024-2025 at ENSAE Paris. My teacher for the practical work was Ms Houri.

different input lengths. We separately analyze results for high-resource languages (with large training corpora) and low-resource languages. Our contributions are:

- We present a systematic analysis of LID performance as a function of input sentence length.
- We introduce an evaluation methodology using truncated sentences in controlled length bins, which allows robust, fair comparisons of LID models across length conditions.
- We compare two open LID models (GlotLID and OpenLID) and report detailed metrics (Accuracy, macro-F1, and False Positive Rate) across length bins. We find that short-text performance is markedly worse, especially for low-resource languages, and we quantify the extent of this gap.
- We discuss implications for improving LID on short inputs and for evaluating LID models in realistic settings where many inputs may be very brief (e.g., tweets, queries).

2 Related Work

Early language identification research focused on character n-gram models [6] and strong heuristics that can work with minimal input. Traditional systems such as Google's Compact Language Detector (CLD3) were designed for short text detection (like browser content snippets), but their coverage was limited (around 100 languages) and accuracy degraded severely on very short strings. The advent of machine learning approaches expanded both accuracy and language coverage. [5] survey numerous LID methods, noting that classification confidence drops with shorter inputs.

High-Resource LID Models: FastText-based models have become popular for LID. The widely-used FastText-176 model covers 176 languages and achieves high accuracy on relatively long text (e.g., Wikipedia sentences) but is less studied on extremely short text. [4] demonstrated FastText's effectiveness for text classification, including LID, using subword information which can help with short words. More recently, [1] introduced **OpenLID**, training on a carefully curated dataset for 201 languages with the FLORES-200 benchmark for evaluation. OpenLID achieved a macro-averaged $F_1 = 0.93$ and an extremely low macro false positive rate of 0.033%, outperforming earlier open models. However, OpenLID's coverage is constrained to languages present in FLORES-200 (mostly medium/high-resource languages), leaving out many low-resource languages.

Low-Resource LID and Broad-Coverage Models: The **GlotLID** project [2] significantly improved coverage of these languages by the release of GlotLID-M, a model covering 1665 languages. GlotLID-M was trained on a large collection of texts aggregated from sources like Wikipedia and other web-based datasets. Notably, GlotLID was optimized to balance detection performance with reliability, explicitly evaluating and minimizing false positive rate (FPR) in addition to F_1. In their evaluation, GlotLID-M outperformed CLD3, FastText-176, OpenLID and NLLB on a balanced metric combining F_1 and FPR. These improvements come with trade-offs: GlotLID's precision-focused training may cause it to abstain or predict "unknown" more often on ambiguous short inputs, whereas other models might guess and incur false positives. There has been some analysis of LID errors specific to low-resource languages: [2] discuss challenges such as mislabelled training data and leakage from high-resource languages (e.g., text erroneously labeled as a low-resource language because of proper names or code-switching). Such issues can especially confuse models on short segments with limited context. However, previous work has not explicitly quantified how varying the input length influences LID for different language groups. Our work builds on these efforts by using GlotLID and OpenLID to systematically explore the sentence-length dimension.

3 Methodology

Our goal is to evaluate LID model performance under controlled sentence length conditions. We define several fixed **sentence length bins** and measure each model's accuracy, macro F_1, and false positive rate when restricted to inputs of those lengths. By truncating longer sentences to populate the shorter bins, we ensure that we have comparable content across length conditions, isolating the effect of length.

3.1 Models

We experiment with two multilingual LID models:

- **GlotLID-M** by [2], which supports 1665 languages. We use the latest public GlotLID model (v3) from the authors' repository. GlotLID outputs ISO 639-3 language codes (potentially with script codes) and was trained on the GlotLID-C corpus comprising text from various web sources. Its training objective favored high F_1 while also penalizing false positives, making it a precision-oriented classifier.
- **OpenLID** (FastText) by [1], which supports 201 languages. We obtained the OpenLID model from HuggingFace Hub [3]. OpenLID outputs BCP-47 style language tags and was trained on a curated dataset audited for correctness. It is a recall-oriented model that achieves very high coverage of its supported languages, albeit with a smaller label set than GlotLID.

Both are lightweight FastText models, enabling efficient inference on a large number of sentences. FastText performs classification using subword features, which is advantageous for identifying languages from short character sequences.

3.2 Data and Controlled Binning

For a comprehensive evaluation, we leverage the **GlotLID-C corpus** introduced by [2]. This corpus contains training data for 1941 language-script combinations (we will refer to them as "languages" for brevity). The size of data available per language varies widely. We categorize languages into three resource groups based on the total size of their monolingual text in the corpus, where we assume this is a good proxy for the general amount of resources available for each language. We distinguish between:

High-resource: Languages with $\geq$ 100 MB of text. (83 languages)

Medium-resource: Languages with 10–100 MB. (265 languages)

Low-resource: Languages with $<$ 10 MB. (1593 languages)

Figure 1 shows the distribution of language corpus sizes on a log-scale. The majority of languages fall into the low-resource category, with a long tail of very small corpora (often under 1MB). Only a small number of languages have very large text corpora exceeding 100MB (e.g., English, French, etc.). In our analysis we focus on the extremes: the High-resource vs. Low-resource groups, to highlight disparities.

Next, we construct evaluation sets for various sentence lengths. We define five length bins: 0–20, 20–40, 40–60, 60–80, and 80–100 characters (character count including spaces). These ranges reflect very short segments (0–20 characters, e.g. a short phrase or single word) up to moderately long sentences (80–100 characters, a typical written sentence). To obtain evaluation data:

1. For each language, we sample a large set of sentences from its corpus. We exclude any sentences that were used in training the GlotLID model to avoid evaluation bias,i.e. we throw away all sentences with less than 100 characters. (The GlotLID authors did not release explicit train/test splits, so we assume random sampling and rely on the vastness of the corpus to minimize overlap.)
2. We apply a **safe truncation** procedure: for each sentence, if its length exceeds a given bin's maximum, we cut it at that length and then strip any trailing partial word fragment (to avoid dangling characters from a cut-off word). This yields a truncated sentence of at most the bin length, without breaking word boundaries mid-word.
3. We populate each length bin with a large number of truncated sentences. Importantly, we aim to use an equal total number of sentences in each bin *for each resource group*. For High-resource languages, we aggregate all sentences from languages in the high group that fall in a given length bin. Similarly, for Low-resource languages. We then down-sample if necessary to ensure the same total count N in each bin for fair comparison. In practice, we chose $N \approx 14,000$ sentences per bin per group, limited by the bin with the fewest available samples (for low-resource, the 0–20 bin was slightly limiting with $\sim$14K sentences).

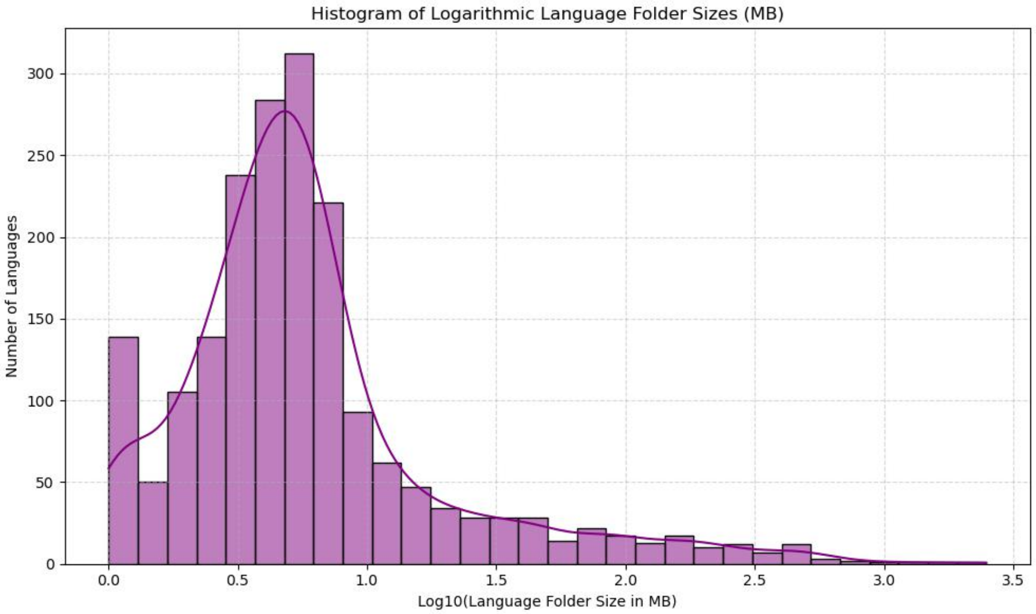

Figure 1: Histogram of Log_{10} of language corpus sizes (in MB) in the GlotLID-C dataset. Most languages have modest data (peak around $10^{0.5} \approx 3$ MB), while only 83 languages exceed 100 MB (High-resource).

This process yields two evaluation sets: one for High-resource languages (combined) and one for Low-resource languages, each stratified into five length bins of equal size. Note that by truncating longer sentences, we introduce some artificial short examples that may not naturally occur. However, this is intentional to control the content across lengths: e.g., a sentence "*This ancient script has been rediscovered.*" might appear truncated as "*This ancient script*" in the 0–20 bin. While such truncation might remove some context, it simulates the scenario where only a fragment is seen by the LID model. We believe this yields a more robust understanding of model behavior on partial inputs. Finally, we generate language predictions on these evaluation sentences using each model. For OpenLID, if it encounters a language outside its 201 supported labels, it will still predict one of its known labels (essentially it will misclassify that sentence as some other language). In our Low-resource evaluation (1593 languages), many languages are not in OpenLID's label set. Rather than confound the analysis with a large number of "unknown language" errors for OpenLID, we restrict the direct comparison between GlotLID and OpenLID to the High-resource set (which largely overlaps with OpenLID's 201 languages). For the Low-resource set, we report GlotLID's performance alone, since OpenLID cannot predict most of those languages.

3.3 Evaluation Metrics

We evaluate three metrics:

- **Accuracy**: the fraction of sentences whose language is correctly identified.
- **Macro-averaged F_1**: we compute F_1 score for each language in the set and average across languages (treating each language equally). This metric is sensitive to performance on less-represented languages and is appropriate given our balanced per-language sampling in each group.
- **False Positive Rate (FPR)**: following [2], we define FPR as the proportion of non-target sentences that are incorrectly labeled as a given language, averaged across languages. In other words, for each language label ℓ, we consider all evaluation sentences that are *not* actually language ℓ, and determine what fraction of those were wrongly predicted as ℓ. This measures how often the model erroneously "hallucinates" each language. We average this

rate over all languages (macro-FPR). A low macro-FPR indicates the model is conservative in assigning languages (few false alarms).

GlotLID's design emphasis was to minimize FPR while maintaining high recall, whereas OpenLID's training aimed to maximize F_1 (and accuracy) on its set of languages without an explicit penalty on false positives. We are particularly interested in how these metrics trade off for short vs. long inputs. All metrics are computed on a per-bin basis for each model and resource group.

4 Experiments and Results

We ran predictions for each model on the prepared evaluation sets. For High-resource languages, both GlotLID and OpenLID were applied; for Low-resource, only GlotLID's predictions are meaningful. We then aggregated the results by length bin and computed the metrics above. Table **??** in the Appendix provides the detailed numerical results. Here we highlight key trends with visualizations.

4.1 Sentence Length vs. Performance

Figure 2 plots the macro F_1 score as a function of sentence length bin for High-resource (GlotLID and OpenLID) and Low-resource (GlotLID) languages. Several observations can be made:

- There is a steep increase in F_1 from the shortest bin to the next for all cases. With 0–20 character inputs, performance is very low: GlotLID achieves $F_1 \approx 0.22$ on high-resource languages, and even lower (≈ 0.25) on low-resource languages, indicating that on average the model can barely do better than chance for many languages with such short input. OpenLID does better on high-resource at 0–20 characters ($F_1 \approx 0.34$), suggesting its precision/recall trade-off is tuned to be more effective on short text for those languages. However, all models see a large jump once we have 20–40 characters: e.g., GlotLID's F_1 more than doubles to 0.43 (high-res) and 0.76 (low-res) in the 20–40 bin.
- Low-resource languages start off worse than high-resource at 0–20 (0.25 vs 0.34 F_1 for GlotLID low vs OpenLID high), but by 40–60 characters GlotLID's performance on low-resource languages actually overtakes its performance on high-resource languages. For 40–60, GlotLID Low $F_1 \approx 0.914$ whereas GlotLID High $F_1 \approx 0.598$. This somewhat surprising result suggests that when sufficient context is available, GlotLID can very reliably identify even low-resource languages. The lower F_1 on high-resource languages at mid-lengths might reflect confusions among certain high-resource languages or dialects that are similar (e.g., among European languages) or that GlotLID's precision bias causes it to be cautious on some high-resource cases, lowering recall.
- For longer sentences (60+ characters), all curves plateau. By the 80–100 bin, OpenLID and GlotLID on high-resource languages both exceed $F_1 = 0.75$ and are nearly converged (OpenLID 0.771 vs GlotLID 0.748). Low-resource GlotLID reaches $F_1 = 0.967$ in this bin, indicating extremely high precision and recall when given a full sentence of 100 characters.
- OpenLID consistently outperforms GlotLID on the high-resource set in terms of F_1 at every length bin, with the gap most pronounced at the shortest length (0.341 vs 0.222). This is likely because OpenLID does not penalize false positives as much, so it can recall more true positives for short ambiguous inputs (at the cost of some more false positives, as we will see). GlotLID's more conservative approach yields lower F_1 for short texts but it closes the gap as length increases.

A similar trend is observed for **Accuracy**, shown in Figure 3. Accuracy is generally higher than F_1 since it is dominated by the most frequent languages in each set, but it reinforces the same pattern: short inputs yield much lower accuracy for low-resource languages (only 21% accuracy at 0–20 characters for GlotLID-Low) compared to high-resource (GlotLID-High 61%, OpenLID-High 66% at 0–20). With more characters, accuracy rapidly approaches ceiling. By 80–100 characters, GlotLID achieves 98.87% accuracy on low-resource languages and 98.36% on high-resource, essentially closing the gap, while OpenLID reaches 98.45% on high-resource. We note that OpenLID slightly surpassed GlotLID in high-resource accuracy at every bin, but the differences beyond the shortest bin are very small (all models $> 95\%$ accurate from 40 characters onward).

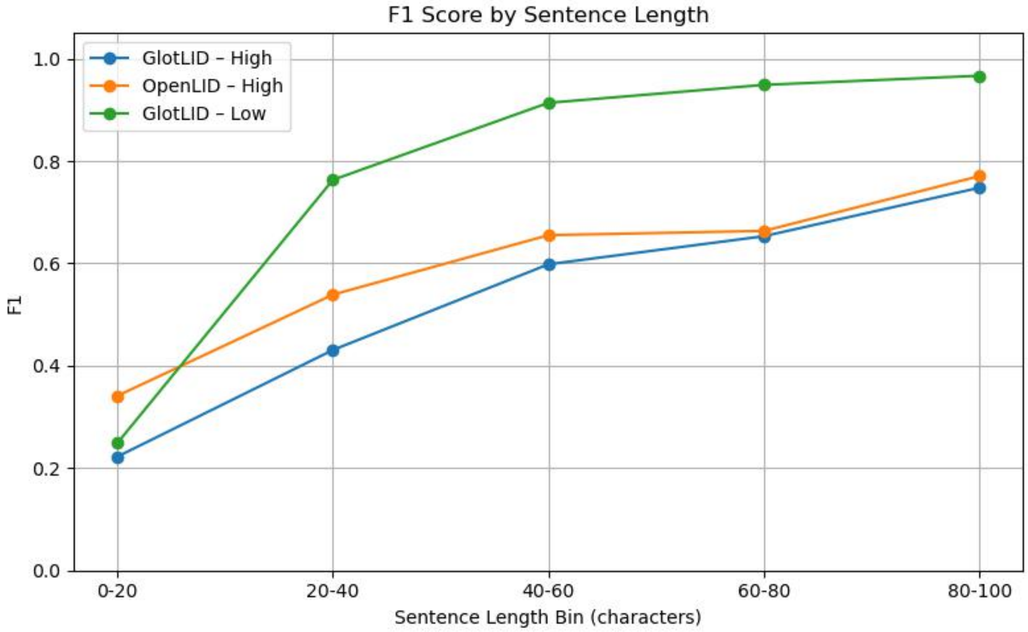

Figure 2: Macro F_1 score by sentence length bin (in characters). Blue/orange curves are GlotLID/OpenLID performance on High-resource languages; green curve is GlotLID on Low-resource languages. Short texts (<20 characters) dramatically reduce F_1, especially for low-resource languages. Performance improves with length for all, and GlotLID Low-resource surpasses High-resource performance once length ≥ 40 characters.

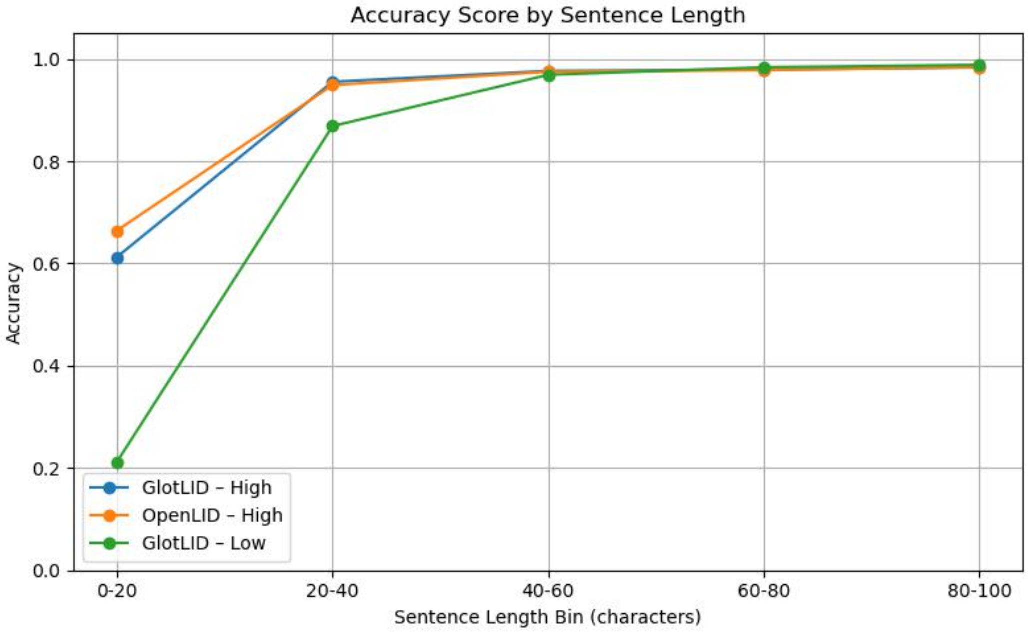

Figure 3: Accuracy by sentence length bin. High-resource languages (blue = GlotLID, orange = OpenLID) and Low-resource (green = GlotLID). With <20 characters, low-resource LID is very poor (21% accuracy) while high-resource is modest (60–66%). Accuracy converges to $> 95\%$ once ≥ 40 characters.

4.2 False Positive Rate Analysis

False positive rate (FPR) is a particularly important metric for real-world LID, because a model that over-predicts a language can introduce spurious data (e.g., mislabeling random strings as a particular language). Figure 4 plots the macro FPR for GlotLID and OpenLID on the high-resource set, and GlotLID on low-resource. Note the y-axis is in the 10^{-4} range (a very low rate). Even for short inputs, the FPR values are on the order of 10^{-3} or less, which indicates that both models are reasonably careful. However, the differences align with each model's design philosophy:

- At 0–20 characters, OpenLID's FPR for high-resource languages is 0.001462 (0.146%), about double GlotLID's FPR of 0.000760. This means OpenLID is more likely to falsely predict a language when the text is actually in another language, presumably because it tries harder to make a guess. GlotLID's conservative approach yields a lower FPR (fewer false identifications), at the expense of lower recall (as reflected in its lower F_1).
- GlotLID's FPR on low-resource languages is even lower: 7.3×10^{-5} at 0–20 characters, an order of magnitude smaller than its FPR on high-resource. This could be because GlotLID is extremely cautious about predicting an obscure language unless it has strong evidence, resulting in very few false alarms for low-resource languages (indeed, GlotLID might instead default to labeling such a short snippet as some high-resource language or "unknown" rather than a low-resource language, thus low FPR for the low-resource class). The flip side is the very low accuracy we saw for low-resource at short lengths—the model often refrains from committing to the correct low-resource language without enough context.
- As length increases, all FPRs plummet. By 40–60 characters, GlotLID's FPR is $\sim 10^{-5}$ and OpenLID's $\sim 10^{-4}$. In the longest bin, FPR is essentially zero for GlotLID-Low (it made virtually no false predictions of low-resource languages given long inputs) and on the order of 10^{-5} for GlotLID-High and OpenLID-High. These values confirm that when plenty of context is available, both models can be very precise in their predictions.

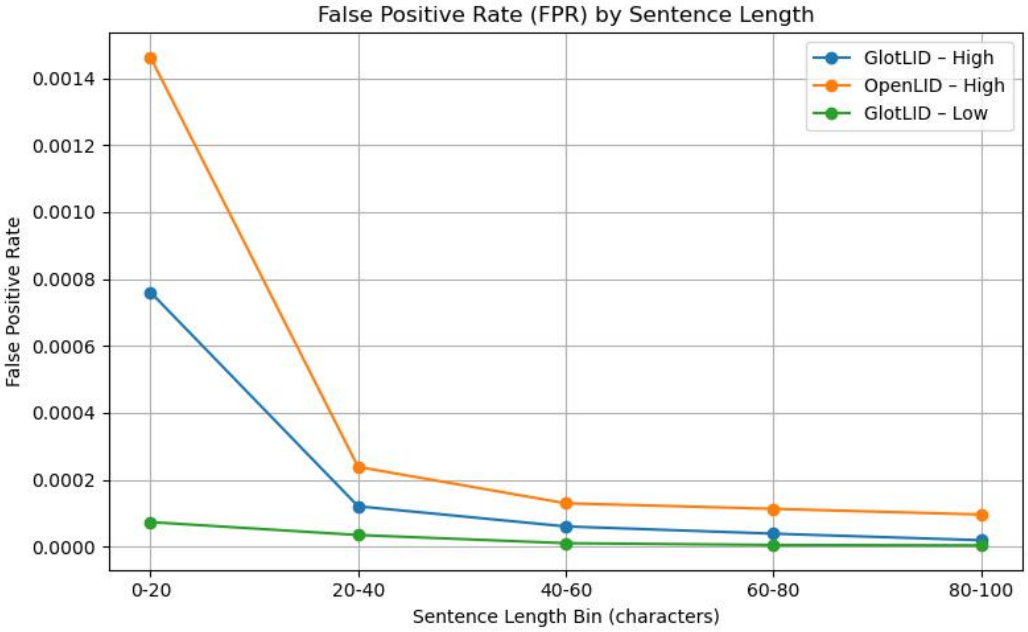

Figure 4: False positive rate (FPR) by sentence length. Lower is better. GlotLID (blue=High, green=Low) consistently has lower FPR than OpenLID (orange) for short inputs, indicating fewer incorrect language hallucinations. All models' FPR drops to near-zero as length increases. Note the log-scale vertical axis.

In summary, our results show a clear pattern: sentence length is a critical factor in LID performance, and its impact is far more severe for low-resource languages. With only a handful of characters, current models struggle: the low-resource recall is especially poor, indicating that many languages cannot be identified from extremely short text. However, given sufficient length (40+ characters, roughly 5-10 words), modern LID models can achieve very high accuracy even on low-resource languages. OpenLID and GlotLID both excel with longer inputs, but OpenLID shows an advantage on very short inputs for high-resource languages, likely due to its more aggressive predictions.

5 Discussion

These findings have several implications:

Challenges with Short Texts. Our analysis quantifies how dramatically LID performance degrades on short texts, especially for low-resource languages. For a low-resource language, a 10-character string (for example, a single common word or name) is often indistinguishable from other languages to the model. There are a few possible reasons: (1) The model's training data for that language may not include such short instances or may be dominated by longer sentences, so it never learned distinctive short patterns. (2) Many low-resource languages share scripts and even loanwords with higher-resource languages; with limited context, the model may default to a more "likely" language. For instance, a short Swahili word might be misidentified as a similar-looking word in Arabic or English if given alone. Our results show that GlotLID tends to avoid making a confident guess in this scenario (hence low F1 but also very low FPR), whereas OpenLID might choose a best guess from its limited label set (yielding higher F1 on average, but at the cost of more false positives).

High-resource vs Low-resource Behavior. It is noteworthy that once input length is sufficient, GlotLID actually performs *better* on the low-resource set than on the high-resource set. One interpretation is that low-resource languages, while numerous, are often quite linguistically distinct from each other (especially if we exclude cases like dialects). If a model can capture the key character sequences or words (which likely requires seeing a full sentence), it can identify the language with high precision. High-resource languages include many closely related pairs (e.g., Spanish vs. Portuguese, Serbian vs. Croatian) that even with a full sentence might occasionally be confused by a model, since these languages can have very similar vocabulary and both are well represented in training (so the model might over-generalize). Additionally, high-resource languages in our set likely include languages with non-Latin scripts that are easier (distinct script = easy detection even with few characters) as well as many with Latin script. The macro-average over that mix might lower the overall F1 a bit compared to the low-resource set which is skewed toward languages with unique scripts or orthographies (for example, many low-resource languages use unique combinations of characters that, once seen, give them away). This could explain the inversion where low-resource F1 > high-resource F1 at longer lengths for GlotLID.

Model Design Trade-offs. GlotLID and OpenLID demonstrate two different philosophies: GlotLID is more conservative (prioritizing precision/FPR), OpenLID more aggressive (prioritizing recall/coverage). In practice, which is preferable can depend on the application. For cleaning a multilingual corpus, one might prefer fewer false positives (to avoid contaminating language-specific corpora with wrong-language text) – here GlotLID's lower FPR is advantageous. But for a user-facing language detector (e.g., in a keyboard or browser), one might prefer the model to always make a best guess for even very short inputs – OpenLID's higher recall would ensure the language is detected more often. Our controlled evaluation reveals that these design choices manifest especially on short inputs. Interestingly, the differences largely vanish with longer inputs; this suggests that it is possible to tune a model to be aggressive on long inputs (where it won't make many mistakes anyway) and cautious on short inputs. Future LID systems might implement length-aware confidence thresholds – for example, requiring higher model confidence to make a prediction if the input is below a certain length.

Improving LID for Short, Low-Resource Texts. What can be done to handle cases where only a very short snippet in a low-resource language is available? One idea is to incorporate external signals or context: for instance, if the text is a tweet or query, meta-data like user location or preceding conversation context might help narrow down plausible languages. Another approach is to augment training data with plenty of short examples (perhaps by fragmenting sentences during training) so the model learns to recognize languages from minimal cues. Training a character-level language model that can assign probabilities to different languages given a character sequence might also complement classification: e.g., a model might not be confident to pick one language for "ana" but it could produce a distribution that indicates it's likely either in Swahili or in Malay. Ensemble or multi-step classification (first detect script, then language family, then specific language) could also improve robustness on short inputs by ruling out large subsets of languages early. These directions are beyond the scope of this paper, but our analysis clearly identifies short-text LID for low-resource languages as a problem area deserving attention.

Evaluation Methodology. By constructing a balanced, length-controlled benchmark, we aimed to fairly compare models and reveal trends that might be obscured in a normal test. Often, LID models are evaluated on test sets with a typical or fixed length (e.g., one sentence from Wikipedia per language). Such evaluations might not show the dramatic failure modes that occur with shorter inputs. We advocate that future evaluations of LID systems include an analysis by input length. In particular, reporting performance on very short (e.g., <20 character) inputs can help users understand if a model is suitable for tasks like identifying the language of search queries, social media posts, or subtitles, which can be very brief. Likewise, our truncation approach allowed us to reuse the same content across bins (e.g., a sentence truncated to 20 characters vs the full sentence at 100 characters), thereby controlling topical content while varying length. This paired evaluation could be developed into a standard methodology for analyzing any text classifier's sensitivity to input truncation.

6 Conclusion

We presented a detailed empirical study on the impact of sentence length in multilingual language identification, comparing the GlotLID (1665-language) and OpenLID (201-language) models. Using a novel evaluation design with controlled sentence-length bins populated by truncated examples, we found that input length plays a pivotal role in LID performance. Very short texts (under 20 characters) severely challenge current LID models, leading to sharp drops in accuracy and F_1—especially for low-resource languages, where identifying the correct language with so little context is often unsuccessful. High-resource languages also see performance degrade on short input, but to a lesser extent, and models like OpenLID can still often guess correctly based on common words or letter sequences. GlotLID and OpenLID showed complementary strengths: OpenLID attained higher recall and F_1 on short high-resource text, whereas GlotLID maintained lower false positive rates and substantially broader language coverage. For inputs of moderate length (40-100 characters), both models achieved excellent results, with GlotLID's performance on low-resource languages matching or exceeding its high-resource performance when ample context is available. This suggests that the fundamental limitation for low-resource LID is not the model's capability but the information content in short utterances. Our findings underscore the need for LID systems to be evaluated and possibly adapted for short-text scenarios. In real-world deployments, a significant fraction of texts may be only a few words (think of news headlines, search queries, or chat messages). A model that performs well on full sentences but poorly on snippets could lead to disproportionate exclusion of low-resource language content. Future work should explore length-aware training objectives or hybrid models to better handle short inputs, as well as incorporating user or contextual cues to assist language identification. Moreover, the community would benefit from shared benchmarks that include a spectrum of input lengths for a wide range of languages, to drive progress on this aspect of multilingual NLP. In conclusion, as multilingual language identification models continue to expand their coverage (thousands of languages in the case of GlotLID) and improve their accuracy, it is equally important to understand their limitations. Our study highlights one such limitation—sentence length—and provides insights that can guide both practitioners (in choosing or calibrating LID models for their use case) and researchers (in designing the next generation of LID systems). By acknowledging and addressing challenges with short-text LID, we move closer to truly robust language identification across the vast diversity of languages.

References

[1] Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. *An Open Dataset and Model for Language Identification.* In *Proc. of ACL 2023 (Short Papers)*, pages 865–879.

[2] Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schütze. 2023. *GlotLID: Language Identification for Low-Resource Languages.* In *Findings of EMNLP 2023*, pages 6155–6218.

[3] OpenLID Model Card. 2023. `https://huggingface.co/laurievb/OpenLID`.

[4] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. *Bag of Tricks for Efficient Text Classification*. In *Proc. of EACL 2017*.

[5] Tommi Jauhiainen, et al. 2019. *Automatic Language Identification in Texts: A Survey*. Journal of Artificial Intelligence Research.

[6] Ted Dunning. 1994. *Statistical Identification of Language*. Technical report, CRL.