Towards Trustworthy Lexical Simplification: Exploring Safety and Efficiency with Small LLMs

Akio Hayakawa **Stefan Bott** **Horacio Saggion**
Department of Engineering, Universitat Pompeu Fabra
Barcelona, Spain
{akio.hayakawa, stefan.bott, horacio.saggion}@upf.edu

Abstract

Despite their strong performance, large language models (LLMs) face challenges in real-world application of lexical simplification (LS), particularly in privacy-sensitive and resource-constrained environments. Moreover, since vulnerable user groups (e.g., people with disabilities) are one of the key target groups of this technology, it is crucial to ensure the safety and correctness of the output of LS systems. To address these issues, we propose an efficient framework for LS systems that utilizes small LLMs deployable in local environments. Within this framework, we explore knowledge distillation with synthesized data and in-context learning as baselines. Our experiments in five languages evaluate model outputs both automatically and manually. Our manual analysis reveals that while knowledge distillation boosts automatic metric scores, it also introduces a safety trade-off by increasing harmful simplifications. Importantly, we find that the model's output probability is a useful signal for detecting harmful simplifications. Leveraging this, we propose a filtering strategy that suppresses harmful simplifications while largely preserving beneficial ones. This work establishes a benchmark for efficient and safe LS with small LLMs. It highlights the key trade-offs between performance, efficiency, and safety, and demonstrates a promising approach for safe real-world deployment.

1 Introduction

Text Simplification (TS) aims to make texts more accessible by rewriting them in simpler language. TS holds the potential to alleviate reading and understanding difficulties, particularly for individuals with dyslexia (Rello et al., 2013), intellectual disabilities (Säuberli et al., 2024), and Deaf and hard-of-hearing adults (Alonzo et al., 2021). TS is a task strongly oriented towards real-world scenarios, aiming to promote social participation and inclusion among people who face challenges in text comprehension.

Recent advancements in large language models (LLMs) have revolutionized natural language processing and achieved state-of-the-art performance across various tasks (OpenAI, 2024). TS is no exception, as LLMs have outperformed existing TS systems (Feng et al., 2023; Wu and Arase, 2024; Qiang et al., 2025).

However, applying LLMs to TS in real-world scenarios, particularly for vulnerable user groups, faces critical challenges. First, prompts provided to LLMs and texts requiring simplification may contain sensitive personal information, such as data related to cognitive impairments. The use of API-based LLMs involves transmitting that sensitive data over the internet, raising significant privacy concerns. For instance, given that individuals with dyslexia often hesitate to disclose their condition due to concerns about stigma and negative perceptions (Hamilton Clark, 2024), it can be problematic to design prompts for TS such as *"I have dyslexia; Can you simplify this diagnosis result for me?"*. Thus, TS systems capable of running locally are highly desirable.

Open-access LLMs address this privacy concern. However, high-performing open-access LLMs typically require substantial computational resources for inference. Deploying such large models directly on resource-constrained devices, such as smartphones and tablets that are commonly used by the target users (Söderström et al., 2021), is currently impractical. This highlights the need for developing smaller models that can perform effectively within these limited hardware environments.

Building on these challenges, we investigate how to develop efficient TS systems that can operate under constrained computational resources. This approach is essential for supporting information access for all while respecting user privacy.

Utilizing small LLMs is a promising approach,

Proceedings of the 18th International Natural Language Generation Conference, pages 215–231
October 29–November 2, 2025. ©2025 Association for Computational Linguistics

as ~3B models are often explicitly engineered for on-device deployment (MetaAI, 2024), thereby addressing privacy and efficiency issues. However, particular attention must be paid to safety when employing small LLMs, as their limited capacity compared to larger counterparts introduces critical considerations regarding the reliability and harmfulness of the generated simplifications. Poor or inaccurate simplifications can be detrimental, as they may actively provide misinformation or cause confusion, which are more serious issues than leaving the text unchanged (Rello et al., 2013; Säuberli et al., 2024). Therefore, in practice, it is crucial not only to simplify texts effectively, but also to minimize harmful outputs and ensure safety.

As a first step towards addressing these challenges, this paper focuses specifically on lexical simplification (LS), a subtask of TS that replaces complex words in a context sentence with simpler alternatives. LS can be considered a relatively conservative and safe subtask compared to sentence- or document-level simplification, which often involves operations such as information deletion (Al-Thanyyan and Azmi, 2021).

We adopted small LLMs and explored two approaches: in-context learning, which requires no training, and knowledge distillation, which transfers knowledge from a larger teacher model to a smaller student model. Our approach also considers extensibility to diverse languages, as supporting a broad user group requires simplification across multiple languages.

To evaluate the safety of simplification outputs, particularly in suppressing harmful content, we conducted manual evaluations alongside automatic metrics. Manual analysis revealed that, while knowledge distillation generally boosted automatic metric scores, it did not reduce harmful outputs and sometimes even increased them. Furthermore, we observed that, especially in models trained via knowledge distillation, the output probability provided by LLMs may serve as a useful signal for identifying harmful simplifications.[1]

Our contributions are summarized as follows:

- We investigated the potential and challenges of using small LLMs for lexical simplification with respect to safety and efficiency, and we establish a benchmark in this important research area.
- We demonstrated that small LLMs offer significant inference speedups, which highlights their efficiency.
- We found that standard approaches such as in-context learning and knowledge distillation can produce beneficial simplifications, but they inherently risk generating harmful outputs.
- We identified that model's log-probability serves as a useful signal for detecting harmful simplifications, suggesting a promising filtering strategy to ensure safety towards real-world applications.

2 Related Work

Lexical Simplification LSBert (Qiang et al., 2021) established itself as a strong baseline for LS by leveraging BERT's unmasking capabilities and contextual understanding, outperforming earlier systems based on paraphrase databases and word embeddings (Biran et al., 2011; Glavaš and Štajner, 2015). However, such systems based on masked language models (MLMs) were limited in generating multi-token words (Przybyła and Shardlow, 2020) and its effectiveness outside English has been questioned (Stajner et al., 2023). Furthermore, MLM-based systems often require multi-stage pipelines involving candidate ranking, which introduces significant latency that conflicts our goal of on-device efficiency. Their multilingual applicability is also hindered by the inconsistent availability of monolingual models across languages.

More recent auto-regressive approaches, using T5 (Sheang and Saggion, 2021) and GPT-3 (Aumiller and Gertz, 2022), have outperformed MLM-based methods, leading to the widespread adoption of LLMs as the predominant solution for LS (Shardlow et al., 2024b). Notably, a GPT-4-based LS system (Enomoto et al., 2024) achieved remarkable performance across multiple languages.

Smaller LLMs and Efficiency The use of high-performing versatile LLMs poses several challenges in real-world scenarios, including resource limitations, privacy concerns, and high operational costs. To address these issues, various efforts have been made to develop LLMs capable of running on local devices. These include techniques such as quantization (Zhou et al., 2024) and the GPT-Generated Unified Format (GGUF),[2] both of which

[1] Our codes will be available at https://github.com/ahaya3776/safe-efficient-ls.

[2] https://github.com/ggml-org/ggml/blob/master/docs/gguf.md

aim to enable efficient inference without high-end hardware, as well as the development of small LLMs (Qwen Team, 2024; Gemma Team, 2024; Meta AI, 2024).

Small LLMs can be further trained to improve performance on specific tasks (Xu et al., 2024), including LS (Baez and Saggion, 2023; Xiao et al., 2024). Baez and Saggion (2023) proposed LSLlama, a LLAMA-7B model fine-tuned on an existing LS dataset, which achieved performance comparable to a GPT-3-based approach. Xiao et al. (2024) introduced the PivotKD framework, which trained Chinese-centric small LLMs using pseudo-instances generated by GPT-4, and built a cost-effective Chinese LS system by incorporating web-based synonym and word sense retrieval during inference. These studies demonstrated the potential of task-specific training of small LLMs for LS. However, their applicability to languages beyond English and Chinese remains uncertain, especially given morphological complexity and disparities in pre-training resources.

Safety and Reliability of Text Simplification While TS supports reading and understanding, it also carries the risk of causing confusion or misinterpretation. In practice, outputs from automatic TS systems often suffer from low factuality (Devaraj et al., 2022) and information loss (Agrawal and Carpuat, 2024), which can negatively affect readers' reading time and accuracy on comprehension questions (Rello et al., 2013; Säuberli et al., 2024). In such cases, leaving the original text unchanged may be preferable to applying a harmful simplification. Therefore, adopting a strategy that accepts simplification only when certain criteria are met offers a practical approach in real-world scenarios. In this regard, Trienes et al. (2024) presented one of the few efforts to assess the potential harm of TS by detecting information loss. However, its reliance on LLMs makes it unsuitable for use in constrained environments.

3 Experimental Setup

Figure 1 illustrates the overall flow of our system development and evaluation. We used the HuggingFace Transformers library[3] for the development of our LS models. A single Tesla T4 GPU with 16 GB of memory was used for the development. To enable high-speed inference on CPUs, the models were converted into the GGUF format using llama.cpp.[4]

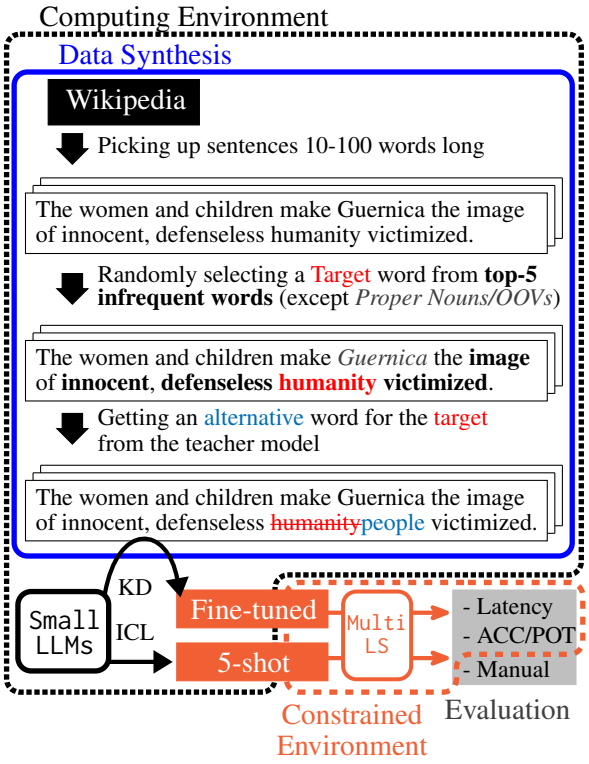

Figure 1: Overall flow of our experiments. We developed and evaluated systems for each language separately.

3.1 Task Formulation

The term *Lexical Simplification* (LS) has been used with varying scopes. In some cases, it refers to a sentence-level simplification pipeline consisting of complex word identification, substitution generation, and ranking (Paetzold and Specia, 2017). However, in this paper, we adopt a narrower definition of LS, focusing solely on the substitution generation. Specifically, we define LS as generating a simpler alternative to a single target word that appears in a given context sentence. An alternative should make the context easier to understand than the original while preserving its meaning. Therefore, an LS system takes a context and target word as input and outputs a single alternative word.

3.2 Dataset

We used MultiLS (Shardlow et al., 2024c), a LS dataset covering 10 languages, to evaluate system performance. We selected five languages, English, Spanish, Catalan, German, and Japanese, to ac-

[3] https://huggingface.co/docs/transformers/

[4] https://github.com/ggml-org/llama.cpp

count for differences in language family, morphological structure, and resource availability.

Table 1 shows an example LS instance, consisting of a context sentence, a target word, and alternative words suggested by multiple human annotators. MultiLS allowed annotators to use a target as an alternative when they could not identify a valid simplification, which often occured when the target was already simple enough (Shardlow et al., 2024a). This annotation scheme enables us to exclude instances where LS is inherently difficult. We removed such instances where the top-ranked alternative was unchanged from the target word. This process resulted in the number of instances per language shown in Table 2. We randomly split the selected instances into two parts, assigning 90 instances for development and the rest for testing.[5]

3.3 LS Systems

We employed two small LLMs: Qwen 2.5 1.5B (Qwen for short) (Qwen Team, 2024) and Llama 3.2 1B (Llama for short) (Meta AI, 2024). Both models were trained on multiple languages from their larger counterparts.[6] To make these models perform LS, we adopted two approaches: in-context learning and knowledge distillation.[7]

3.3.1 In-Context Learning

In-context learning (Brown et al., 2020), which provides several examples as few-shot to guide model behavior, is a common technique to improve output quality. We used five fixed examples in the prompt **(5-shot)** following the template in Appendix A. These examples were sampled from the pilot split of MultiLS, which was separated from the main evaluation data.

3.3.2 Knowledge Distillation

Knowledge distillation, which involves transferring knowledge of larger teacher models to smaller student models, has been widely used to adapt LLMs to specific tasks, including LS (Baez and Saggion, 2023; Xiao et al., 2024). Recent approaches commonly employ simple supervised fine-tuning of student models with hard labels derived from teacher model outputs, due to the advanced capabilities of closed-source LLMs (Xu et al., 2024). Following this framework, we performed knowledge distillation (**fine-tuned**) by synthesizing LS instances.

Context: Electronically controlled motorized zoom lenses are placed on both camera and projector, and synchronized with one another so that both lenses zoom together and at the same focal length at all times.
Target Word: focal
Gold Alternatives: main, main, central, central, basic, primary, ~~focal~~

Table 1: Example from the MultiLS English subset. For this instance, **ACC** is met if the output alternative is "main" or "central", which are the most suggested alternatives. **POT** is met if the output alternative is one of "main", "central", "basic", and "primary". If the output alternative is "focal", which is unchanged from the target word, it does not meet either metric.

Language	# Original Instances	# Selected Instances	Avg. Context Length
English	570	515	25.4
Spanish	593	502	29.3
Catalan	445	261	45.0
German	570	547	37.7
Japanese	570	562	20.3

Table 2: Statistics of MultiLS instances per language.

Synthesizing Context and Targets We randomly extracted context sentences from Wikipedia for each language. Sentences were parsed using MeCab[8] for Japanese and spaCy[9] for the other languages. We retained only those containing between 10 and 100 words as contexts.[10]

To ensure that target words were simplifiable, we excluded proper nouns and out-of-vocabulary words from the set of candidate words within each context sentence. From the remaining candidates, we randomly selected one of the five least frequent words as the target word, based on Zipf frequency.[11]

Synthesizing Alternative Words To obtain alternative words for the context-target pairs described above, we employed the instruction-tuned Gemma 2 9B (Gemma Team, 2024) as a teacher model, an LLM known for its strong performance across diverse languages. The model was prompted to generate a single alternative word using the same 5-shot setting described in § 3.3.1.

[5] As up to three instances share the same context, we assign 90 instances with 30 unique contexts to the development data.

[6] We used base LLMs instead of instruction-tuned versions as base LLMs. See Appendix C for details.

[7] See Appendix B for the hyperparameter settings.

[8] https://taku910.github.io/mecab/

[9] https://spacy.io/

[10] For Japanese, simple tokenization rules were applied. See Appendix D for details.

[11] Calculated using wordfreq Python library: https://github.com/rspeer/wordfreq/

The performance of fine-tuned student models can often be improved by removing low-quality outputs from the teacher (Jung et al., 2023; Huang et al., 2023). Therefore, we filtered out low-confidence alternatives, approximating confidence using output probabilities (described later in § 3.4.4). For each language, we generated alternatives for 60,000 synthesized context-target pairs and selected the top 30,000 high-confidence instances for training.

Fine-tuning Models We fine-tuned each student model for each language separately, using the corresponding 30,000 instances for up to five epochs. To reduce memory consumption, we adopted the QLoRA framework (Dettmers et al., 2023). In this setup, the weights of base models were quantized to 4-bit precision using the bitsandbytes[12] library. Fine-tuning was then performed via 16-bit LoRA adapters. Following Dettmers et al. (2023), we only fine-tuned Query and Key projections layers within the attention modules. Each type of student model was fine-tuned with three different random seeds. We saved a checkpoint every 0.2 epochs and selected the one that achieved the highest Potential@1 (described later in § 3.4.1) on the development set. The prompt template in Appendix A was used for training and inference.

3.3.3 Baselines

As a baseline, we employed the instruction-tuned Gemma 2 9B (Gemma for short) in the same 5-shot setting used for the teacher model.

3.4 Evaluation

3.4.1 Automatic LS Metrics

To automatically evaluate the performance of LS systems, we used **Accuracy@1@top1 (ACC)** and **Potential@1 (POT)**, as defined in Saggion et al. (2022). As shown in Table 1, ACC is the percentage of predictions matching the most frequently suggested alternative. POT is the percentage of predictions matching any suggested alternative. Given that all instances were assumed simplifiable after the selection process in § 3.3.1, any predictions unchanged from the target word were not considered a match for either ACC or POT, even if the target word was included in the gold alternatives.

3.4.2 Latency Evaluation

To estimate model response time in resource-constrained environments, we constructed a virtual small-scale infrastructure using computing instances from Amazon Web Services (AWS). We selected m6g.large and m6g.xlarge computing instances from AWS Elastic Computing Cloud, which provide 2 and 4 virtual CPUs and 8 GB and 16 GB of memory, respectively. These configurations reflect the hardware commonly found in smartphones and tablets. Both computing instances are based on Graviton processors, which are widely applied in mobile devices.[13]

Total latency mainly consists of prompt processing time and inference time. As both depend on the number of tokens in the prompt and the generated output, we measured the average pre-token prompt processing and inference times for each model using llama.cpp. Notably, llama.cpp caches the initial fixed portion of the prompt (i.e., few-shot examples), so its processing latency is not incurred on subsequent inferences. While this caching is key to the efficiency, it makes dynamic prompting strategies impractical, as they would require frequent cache invalidation.

3.4.3 Manual LS Evaluation

To gain a more nuanced understanding of LS quality and safety from a user perspective, we conducted a manual evaluation. We randomly sampled 100 instances per language and assigned harmfulness tags to the alternatives generated by each system. Our manual evaluation focused on instances that were not covered by our automatic metrics. For this purpose, we only assigned tags to alternatives that were neither unchanged from the target nor included in the gold alternatives.

Taking into account the standard human evaluation criteria of fluency, adequacy, and simplicity in TS, we defined the following four harmful tags:

- *Grammar Error*: The alternative is grammatically incorrect, including inflection, and conjugation errors.
- *Change of Meaning*: Replacing the target with the alternative drastically changes the meaning of context.
- *More Difficult*: The alternative is clearly more difficult than the target, even though it preserves the meaning to some extent.

[12] https://github.com/bitsandbytes-foundation/bitsandbytes

[13] https://aws.amazon.com/ec2/instance-types/m6g/

Model	Settings	LS Performance										Latency (msec / token)			
		English		Spanish		Catalan		German		Japanese		m6g.large		m6g.xlarge	
		ACC	POT	ACC	POT	ACC	POT	ACC	POT	ACC	POT	read	pred	read	pred
Gemma(9B)	5-shot	.529	.751	.427	.774	.333	.690	.405	.643	.252	.494	652	581	326	292
Qwen(1.5B)	5-shot	.358	.534	.274	.473	.076	.205	**.186**	**.298**	.064	.150	91	275	45	139
	fine-tuned	**.382**	**.574**	**.318**	**.537**	**.129**	**.265**	.119	.206	**.076**	**.154**	86	274	43	138
Llama(1B)	5-shot	.202	.278	.053	.092	.047	.105	.090	.142	.023	.042	70	219	35	110
	fine-tuned	**.370**	**.544**	**.293**	**.529**	**.160**	**.292**	**.138**	**.217**	**.058**	**.145**	66	221	33	107

Table 3: Performance of models on the MultiLS dataset. Gemma was quantized to 4-bit due to memory constraints.

- *Gibberish*: The alternative does not make sense at all.

For each language, annotation was performed by a single in-house annotator, all of whom were native speakers except for Catalan. The Catalan annotation was conducted by a CEFR C1 level speaker with over ten years of experience. The task was designed as a simple binary decision to minimize subjectivity, ensuring the evaluation framework is easily extensible to other languages and domains.

Based on the automatically and manually assigned tags, alternatives were categorized into following three groups. Tags determined by automatic metrics are marked with A, while those requiring manual annotation are marked with M.

- `Beneficial`
  - `ACC` (A) : equivalent to Accuracy@1@top1.
  - `POT` (A) : Potential@1 but not `ACC`
  - `Good` (M) : no harmful tags were assigned.
- `Unchanged` (A) : alternative was identical to target.
- `Harmful`
  - `Degraded` (M) : one or more non-*Gibberish* harmful tags were assigned.
  - `Gibberish` (M) : *Gibberish* was assigned.

See Appendix E for detailed examples of the harmful tags and groups.

3.4.4 Filtering Strategy

To address the risk of introducing harmful simplifications discussed above, we propose and evaluate a filtering strategy. This strategy leverages the output probability score as a reliability signal in a threshold-based decision mechanism to determine whether to perform LS.

Probability Score We computed the probability score as the sum of the log-probabilities of the tokens forming the alternative word, including the token indicating the end of the word (e.g., a newline or EOS token). We considered the probability scores of individual alternatives as candidate thresholds. For each threshold value, alternatives with scores above the threshold were accepted, while others were rejected, and no simplification was applied.

Evaluation To quantitatively evaluate the effectiveness of the proposed strategy, we defined the following metrics:

- **AUC** (`Beneficial` vs `Harmful`): To assess how well the probability score functions as a safety signal, we computed the Area Under the ROC Curve (Bradley, 1997) for classifying alternatives as `Beneficial` vs. `Harmful`, excluding `Unchanged` alternatives.
- $\boldsymbol{B_{H_{0.1}}}$ (`Beneficial` Rate at 10% `Harmful`): To quantify practical benefit under a safety constraint, we reported the rate of `Beneficial` achieved when the rate of `Harmful` introduced was limited to 10% of total instances. We chose the 10% threshold to balance safety and utility by offering a practical reference point for comparison that remains adaptable to different needs.

4 Results

4.1 Automatic Evaluation

Table 3 shows the automatic metric scores for our LS systems. The results confirm our hypothesis that fine-tuning, as part of the knowledge distillation, improved the performance of small LLMs. For example, fine-tuned Llama achieved 0.370 ACC on English, significantly higher than the 5-shot score (0.202). Similar gains were observed for both Llama and Qwen across most languages.

The fine-tuned Llama performed comparably to Qwen despite its smaller size, suggesting that the 1B model can approach 1.5B model in performance

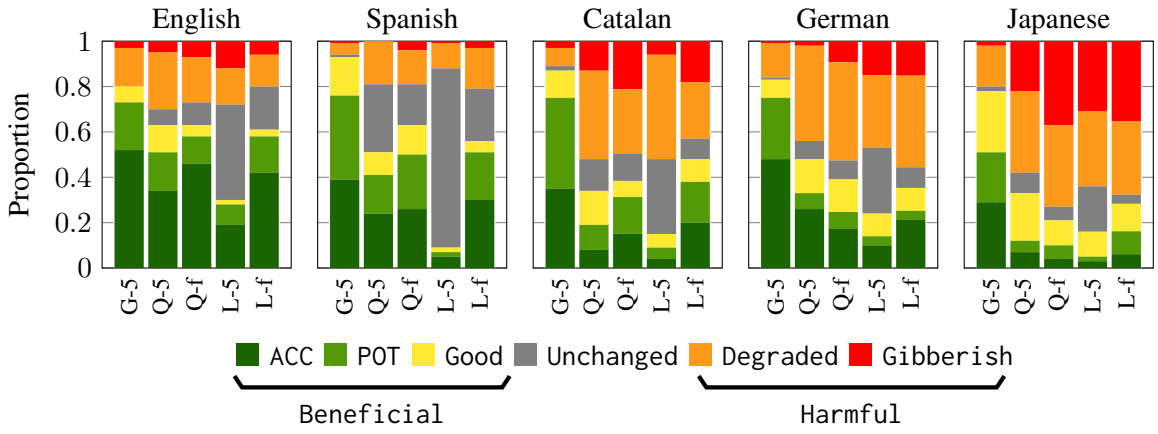

Figure 2: Distribution of output alternative categories. G: Gemma, Q: Qwen, L: Llama. -5: 5-shot, -f: fine-tuned.

after training. However, neither student models reached the teacher's level.

Table 3 also reports the latency (ms/token) for prompt reading (read) and output generation (pred). Both student models showed substantially lower latency than the teacher model. On m6g.large, Llama's read latency (66 msec/token) was nearly 10 times faster than Gemma's (652 msec/token), with similar trends across environments.

4.2 Manual Evaluation

Figure 2 shows the distribution of alternative categories, as judged by human evaluators, across models, settings, and languages. Each stacked bar represents the proportion of output alternatives falling into the categories.

Under 5-shot settings, small LLMs, especially Llama for English and Spanish, produced a high proportion of `Unchanged` outputs, indicating safer but less proactive simplification behavior. Fine-tuning reduced `Unchanged` and corresponding rise in `Beneficial` simplifications, reflecting a general improvement in LS capability. However, fine-tuning also introduced a safety trade-off, as it increased the proportions of `Harmful` alternatives.

In contrast, such trade-off was not observed for German and Japanese. For these languages, performance remained low across both 5-shot and fine-tuned settings, with `Harmful` alternatives consistently dominating the results. This suggests a more fundamental challenge stemming from the inherent difficulty for current small LLMs to perform LS effectively in these languages.

Lang	Model	Settings	r_B	r_H	AUC	$B_{H_{0.1}}$
En	Qwen	5-shot	.63	.30	.679	.41
		fine-tuned	.63	.27	**.707**	**.46**
	Llama	5-shot	.30	.28	.510	.12
		fine-tuned	.61	.20	**.797**	**.54**
Es	Qwen	5-shot	.51	.19	.737	.46
		fine-tuned	.63	.19	**.850**	**.61**
	Llama	5-shot	.09	.12	**.907**	.09
		fine-tued	.56	.21	.804	**.50**
Ca	Qwen	5-shot	.34	.52	.735	.18
		fine-tuned	.38	.49	**.904**	**.34**
	Llama	5-shot	.15	.52	.614	.03
		fine-tuned	.46	.42	**.813**	**.36**
De	Qwen	5-shot	.41	.38	**.841**	**.34**
		fine-tuned	.38	.51	.721	.16
	Llama	5-shot	.19	.40	.730	.11
		fine-tuned	.35	.55	**.737**	**.16**
Ja	Qwen	5-shot	.33	.58	**.807**	**.16**
		fine-tuned	.21	.73	.799	.13
	Llama	5-shot	.16	.64	.745	.04
		fine-tuned	.28	.67	**.845**	**.19**

Table 4: Evaluation of Filtering Strategy. r_B and r_H refer to the original rate of `Beneficial` and `Harmful` outputs.

4.3 Filtering Strategy

Table 4 presents the results of filtering strategy. First, the AUC scores are notably high, especially under fine-tuned settings, suggesting that log-probability serves as an effective signal for detecting **Harmful** alternatives. Moreover, the fine-tuned models generally show higher AUC across model types and languages, which indicates that knowledge distillation enhances the quality of probability as a safety indicator.

The $B_{H_{0.1}}$ metric shows the practical value of this strategy. For example, in Spanish, fine-tuned

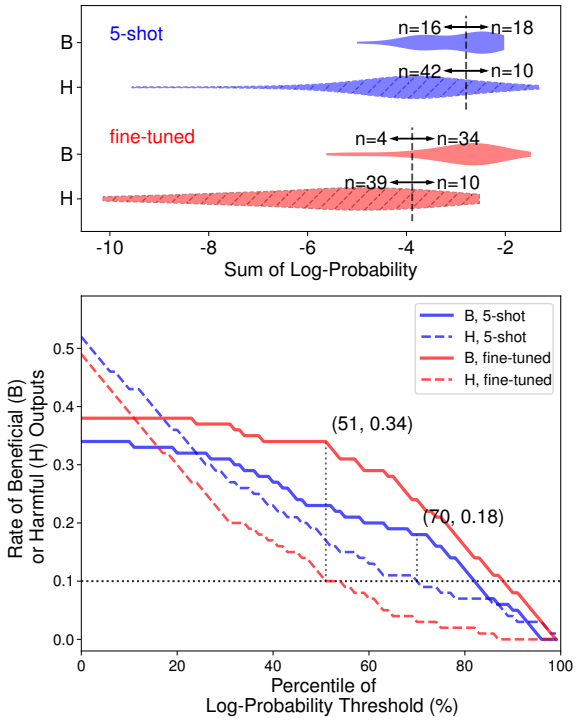

Figure 3: `Beneficial` and `Harmful` alternatives and their probability of Qwen in Catalan. (Top) Distribution of raw probability scores. (Bottom) Rate of `Beneficial` and `Harmful` alternatives after filtering at each percentile threshold. Dotted lines are plotted on thresholds where `Harmful` becomes 10%.

Qwen reduced `Harmful` rate from 19% to 10% with only a slight drop in `Beneficial` from 63% to 61%. $B_{H_{0.1}}$ also highlights the superiority of fine-tuning to 5-shot settings.

To further explore these findings, we focus on the behavior of Qwen models in Catalan. Here, while the original `Beneficial` and `Harmful` rates are close between 5-shot and fine-tuned settings, the impact of filtering strategy differs significantly. In Figure 3, the violin plot (top) visualizes the distribution of log-probability scores, where fine-tuning leads to a clear separation between `Beneficial` and `Harmful` alternatives.

The line plot (bottom) tracks `Beneficial` and `Harmful` rates across thresholds percentiles. For the fine-tuned model, increasing the threshold reduces `Harmful` rapidly, while `Beneficial` declines more gradually. As a result, the `Harmful` rate is reduced from nearly 50% to 10%, with most `Beneficial` simplification preserved.

Context: There are also different editing styles in the sense of how bold people are willing to be.
Target Word: editing
Gold Alternatives: changing, modifying, altering ...
Gemma 5-shot (4%): writing (*Change of Meaning*)
Qwen 5-shot (92%): writing (*Change of Meaning*)
Qwen fine-tuned (3%): proofreading (*More Difficult*)

Table 5: Example outputs from the LS systems. Percentages next to system names indicate log-probability percentiles within each system.

5 Discussion

5.1 Case Study

To better understand the characteristics of model outputs, particularly harmful simplifications overlooked by automatic metrics and the potential of the log-probability signal, we present an example in Table 5. In this example, model output alternatives "writing" and "proofreading" were categorized as `Harmful`, with the tags "*Change of Meaning*" and "*More Difficult*", respectively. Crucially, these alternatives were associated with lower log-probability percentiles for Gemma (5-shot) and fine-tuned Qwen, while they were much higher for Qwen under the 5-shot setting. This case confirms our findings that fine-tuned models effectively leverage log-probability to identify harmful alternatives. It also shows that log-probability is a useful signal for the teacher model, even without fine-tuning. This validates the filtering processed used during data synthesis. Examples in other languages are described in Appendix F.

5.2 Safety

As exemplified by the case study above, harmful LS alternatives pose a serious risk in real-world scenarios. Our manual evaluation revealed key limitations of standard automatic evaluation metrics based on human-provided alternatives. They fail to identify acceptable simplifications not in the gold alternatives, and they do not expose harmful alternatives. Although manual evaluation is costly and not scalable, our harmfulness annotations provide a valuable basis for building automatic detection methods, such as LLM-as-a-judge, to support more practical safety assessment.

Harmful alternatives were particularly pronounced in German and Japanese. In these languages, complex morphology may hinder the consistent generation of correct and simple single-word alternatives by small LLMs. Our error anal-

ysis highlights a critical challenge related to this: alternatives with the *Grammar Error* tag in German and Japanese often received high probability scores from small LLMs (both few-shot and fine-tuned), making them difficult to distinguish from beneficial alternatives or other harmful types. For instance, the average log-probability score for *Grammar Error* from the fine-tuned Llama model in Japanese was -2.992, which was notably higher than that for *Change of Meaning* (-3.762) and *Gibberish* (-4.457). This suggests that our filtering strategy had limited effectiveness in mitigating grammar errors.

Interestingly, this issue was less prevalent in the teacher model (see Appendix G for details across all tags and models). This disparity implies that non-small LLMs can better leverage output probability as a signal for grammatical correctness even in morphologically complex languages. In contrast, small LLMs may struggle to capture these fine-grained grammatical nuances with simple approaches such as in-context learning and knowledge distillation. Incorporating instances with grammatical errors as negative examples in contrastive learning may help student models learn to avoid them, enhancing the reliability of threshold-based filtering.

While log-probability is effective for filtering harmful alternatives, selecting an appropriate threshold for real-world use requires careful tuning based on human evaluation, taking into account domain- and language-specific considerations and practical application needs, to ensure both safety and utility.

5.3 Latency

While the smaller models offer substantial speed improvements, their practical inference speed for real-time and on-device LS needs further consideration. Assuming that a standard input consists of 30 tokens and the output alternative word is composed of two tokens, the overall inference time for fine-tuned Llama on the faster m6g.xlarge environment would be about 1.2 seconds: (30 tokens * 33 ms/token [read]) + (2 tokens * 107 ms/token [pred]) = 1204 ms.

Although a response time of around one second may be tolerable in some cases, further reduction would likely improve the user experience on mobile devices. One possible approach is to reduce the prompt size by including only a limited window of words surrounding the target, rather than the full sentence. Naturally, this strategy would require careful safety assessment.

6 Conclusion

This study addressed the challenge of building efficient and safe LS systems using small LLMs, motivated by real-world needs. We proposed benchmark systems in five languages based on in-context learning and knowledge distillation, and introduced a filtering strategy using log-probability as a safety signal. Experiments showed that small LLMs offer significant efficiency gains, but that knowledge distillation, while improving automatic metrics score, increases harmful outputs.

We demonstrated that output log-probability serves as an effective signal for detecting harmful simplifications. This signal enables filtering strategy that reduce harmful outputs while retaining beneficial ones. These findings lay the foundation for lightweight LS systems that remain safe and effective across languages.

Future work should improve training methods to reduce harmfulness and explore real-time LS for mobile environments. Ultimately, this research advances deployable, trustworthy LS tools that support inclusive information access.

Limitations

Our study, while demonstrating the potential of small LLMs for efficient and safer lexical simplification, has several limitations that highlight directions for further investigation.

First, the manual evaluation of harmfulness was conducted by a single annotator per language. While the annotation task was designed as a simple binary decision to minimize subjectivity, we were unable to assess inter-annotator agreement, which may affect the generalizability of the harmfulness evaluations. Establishing a more robust evaluation protocol with multiple annotators would be a valuable next steop to create a gold-standard dataset for harmfulness detection in LS.

Next, we employed relatively simple prompt engineering, using fixed 5-shot examples and prompt templates to ensure reproducibility and establish baseline performance. We did not explore advanced prompt engineering techniques, which could potentially enhance the models' performance. Future work could investigate how more sophisticated prompting strategies impact the trade-off between performance and safety explored in this study.

This study adopted a narrow task definition, focusing on generating a single simpler alternative for each target word. The systems were not designed to produce multiple candidate simplifications or to handle multi-word expressions, which are often important for user understanding and for simplifying nuanced concepts. Extending our framework to sentence- or paragpraph-level simplification would be a crucial step towards more practical TS tools.

Furthermore, our investigation focused only on generating simpler alternatives. Other important aspects of lexical simplification, such as identifying complex words and selecting the most appropriate simplification, were not addressed in this work. Integrating our safety-aware models into a full LS pipeline is an essential direction for future research.

Finally, the sensitivity of model performance to quantization is a critical limitation. Our methodology involves distinct quantization steps, 4-bit precision during fine-tuning and GGUF for deployment, which can introduce performance discrepancies. Although we observed only negligible performance changes in this study, smaller models are generally more vulnerable to degradation from such processes. Therefore, there is a possibility that our framework might not operate as expected under different quantization schemes, potentially affecting its reliability.

Ethical Considerations

We used publicly available data sources and followed their respective licenses. The training data is from Wikipedia, which is available under the Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0) license. For evaluation, we used the MultiLS dataset, also distributed under a CC BY-SA license. The code used for training and evaluation will be made publicly available upon publication.

Acknowledgments

This work is partially financed by the Ministerio de Ciencia, Innovación y Universidades, Agencia Estatal de Investigaciones: project CPP2023-010780 funded by MICIU/AEI/10.13039/501100011033 and by FEDER, UE ("Habilitando Modelos de Lenguaje Responsables e Inclusivos"). We also acknowledge funding from the European Union's Horizon Europe research and innovation program under Grant Agreement No. 101132431 (iDEM Project). Horacio Saggion also receives support from the Spanish State Research Agency under the Maria de Maeztu Units of Excellence Programme (CEX2021-001195-M) and from the Departament de Recerca i Universitats de la Generalitat de Catalunya (ajuts SGR-Cat 2021).

References

Sweta Agrawal and Marine Carpuat. 2024. Do text simplification systems preserve meaning? a human evaluation via reading comprehension. *Transactions of the Association for Computational Linguistics*, 12:432–448.

Suha S Al-Thanyyan and Aqil M Azmi. 2021. Automated text simplification: a survey. *ACM Computing Surveys (CSUR)*, 54(2):1–36.

Oliver Alonzo, Jessica Trussell, Becca Dingman, and Matt Huenerfauth. 2021. Comparison of methods for evaluating complexity of simplified texts among deaf and hard-of-hearing adults at different literacy levels. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–12.

Dennis Aumiller and Michael Gertz. 2022. UniHD at TSAR-2022 shared task: Is compute all we need for lexical simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 251–258, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Anthony Baez and Horacio Saggion. 2023. LSLlama: Fine-tuned LLaMA for lexical simplification. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 102–108, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Or Biran, Samuel Brody, and Noémie Elhadad. 2011. Putting it simply: a context-aware approach to lexical simplification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 496–501, Portland, Oregon, USA. Association for Computational Linguistics.

Andrew P Bradley. 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Ashwin Devaraj, William Sheffield, Byron Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7331–7345, Dublin, Ireland. Association for Computational Linguistics.

Taisei Enomoto, Hwichan Kim, Tosho Hirasawa, Yoshinari Nagai, Ayako Sato, Kyotaro Nakajima, and Mamoru Komachi. 2024. TMU-HIT at MLSP 2024: How well can GPT-4 tackle multilingual lexical simplification? In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 590–598, Mexico City, Mexico. Association for Computational Linguistics.

Yutao Feng, Jipeng Qiang, Yun Li, Yunhao Yuan, and Yi Zhu. 2023. Sentence simplification via large language models. *arXiv preprint arXiv:2302.11957*.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Goran Glavaš and Sanja Štajner. 2015. Simplifying lexical simplification: Do we need simplified corpora? In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 63–68, Beijing, China. Association for Computational Linguistics.

Charlotte H Hamilton Clark. 2024. Dyslexia concealment in higher education: Exploring students' disclosure decisions in the face of uk universities' approach to dyslexia. *Journal of Research in Special Educational Needs*.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1051–1068, Singapore. Association for Computational Linguistics.

Jaehun Jung, Peter West, Liwei Jiang, Faeze Brahman, Ximing Lu, Jillian Fisher, Taylor Sorensen, and Yejin Choi. 2023. Impossible distillation: from low-quality model to high-quality dataset & model for summarization and paraphrasing. *arXiv preprint arXiv:2305.16635*.

Meta AI. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`. Accessed: 2025-05-05.

MetaAI. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models.

Microsoft. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Gustavo H Paetzold and Lucia Specia. 2017. A survey on lexical simplification. *Journal of Artificial Intelligence Research*, 60:549–593.

Piotr Przybyła and Matthew Shardlow. 2020. Multiword lexical simplification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1435–1446, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jipeng Qiang, Minjiang Huang, Yi Zhu, Yunhao Yuan, Chaowei Zhang, and Kui Yu. 2025. Redefining simplicity: Benchmarking large language models from lexical to document simplification. *Preprint*, arXiv:2502.08281.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, Yang Shi, and Xindong Wu. 2021. Lsbert: Lexical simplification based on bert. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3064–3076.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Luz Rello, Ricardo Baeza-Yates, Stefan Bott, and Horacio Saggion. 2013. Simplify or help? text simplification strategies for people with dyslexia. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*, pages 1–10.

Horacio Saggion, Sanja Štajner, Daniel Ferrés, Kim Cheng Sheang, Matthew Shardlow, Kai North, and Marcos Zampieri. 2022. Findings of the TSAR-2022 shared task on multilingual lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 271–283, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Andreas Säuberli, Franz Holzknecht, Patrick Haller, Silvana Deilen, Laura Schiffl, Silvia Hansen-Schirra, and Sarah Ebling. 2024. Digital comprehensibility assessment of simplified texts among persons with intellectual disabilities. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–11.

Matthew Shardlow, Fernando Alva-Manchego, Riza Batista-Navarro, Stefan Bott, Saul Calderon Ramirez, Rémi Cardon, Thomas François, Akio Hayakawa, Andrea Horbach, Anna Hülsing, Yusuke Ide, Joseph Marvin Imperial, Adam Nohejl, Kai North, Laura Occhipinti, Nelson Peréz Rojas, Nishat Raihan, Tharindu Ranasinghe, Martin Solis Salazar, Marcos Zampieri, and Horacio Saggion. 2024a. An extensible massively multilingual lexical simplification pipeline dataset using the MultiLS framework. In *Proceedings of the 3rd Workshop on Tools*

and Resources for People with REAding DIfficulties (READI) @ LREC-COLING 2024, pages 38–46, Torino, Italia. ELRA and ICCL.

Matthew Shardlow, Fernando Alva-Manchego, Riza Batista-Navarro, Stefan Bott, Saul Calderon Ramirez, Rémi Cardon, Thomas François, Akio Hayakawa, Andrea Horbach, Anna Hülsing, Yusuke Ide, Joseph Marvin Imperial, Adam Nohejl, Kai North, Laura Occhipinti, Nelson Peréz Rojas, Nishat Raihan, Tharindu Ranasinghe, Martin Solis Salazar, Sanja Štajner, Marcos Zampieri, and Horacio Saggion. 2024b. The BEA 2024 shared task on the multilingual lexical simplification pipeline. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 571–589, Mexico City, Mexico. Association for Computational Linguistics.

Matthew Shardlow, Kai North, and Marcos Zampieri. 2024c. Multilingual resources for lexical complexity prediction: A review. In *Proceedings of the Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC-COLING 2024*, pages 51–59, Torino, Italia. ELRA and ICCL.

Kim Cheng Sheang and Horacio Saggion. 2021. Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 341–352, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Sylvia Söderström, May Østby, Hege Bakken, and Karl Elling Ellingsen. 2021. How using assistive technology for cognitive impairments improves the participation and self-determination of young adults with intellectual developmental disabilities. *Journal of intellectual disabilities*, 25(2):168–182.

Sanja Stajner, Daniel Ibanez, and Horacio Saggion. 2023. LeSS: A computationally-light lexical simplifier for Spanish. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1132–1142, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Jan Trienes, Sebastian Joseph, Jörg Schlötterer, Christin Seifert, Kyle Lo, Wei Xu, Byron Wallace, and Junyi Jessy Li. 2024. InfoLossQA: Characterizing and recovering information loss in text simplification. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4263–4294, Bangkok, Thailand. Association for Computational Linguistics.

Xuanxin Wu and Yuki Arase. 2024. An in-depth evaluation of gpt-4 in sentence simplification with error-based human assessment. *arXiv preprint arXiv:2403.04963*.

ZiHao Xiao, Jiefu Gong, Shijin Wang, and Wei Song. 2024. Optimizing Chinese lexical simplification across word types: A hybrid approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15227–15239, Miami, Florida, USA. Association for Computational Linguistics.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. 2024. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*.

Zixuan Zhou, Xuefei Ning, Ke Hong, Tianyu Fu, Jiaming Xu, Shiyao Li, Yuming Lou, Luning Wang, Zhihang Yuan, Xiuhong Li, Shengen Yan, Guohao Dai, Xiao-Ping Zhang, Yuhan Dong, and Yu Wang. 2024. A survey on efficient inference for large language models. *Preprint*, arXiv:2404.14294.

```
Given the context and the specified target in
{language}, provide a simpler alternative word.

{5-shot examples}

Context: {context}
Target Word: {target}
Alternative Word:
```

Table 6: Prompt template for 5-shot settings.

```
Context: {context}
Target Word: {target}
Simplified: {alternative}
```

Table 7: Prompt template for training models. *{alternative}* was removed during the inference of fine-tuned models.

A Prompts Provided to LLMs

We provided the prompt in Table 6 for few-shot learning and in Table 7 for fine-tued models. The prompt for fine-tuning was shortened to minimize inference time.

B Hyperparameters

Table 8 shows the hyperparameter settings we used for inference and training. For other hyperparameters, we used default values of `GenerationConfig`, `TrainingArguments`, and `LoraConfig` classes of Huggingface Transformers.

C Model Selection

For selecting the teacher model, we considered LLMs that are mid-sized, open-source, and capable of high multilingual performance, taking into account the need to synthesize large amounts of data. Table 9 presents the results for the candidate models: Qwen 2.5 14B, Phi-3 medium (Microsoft, 2024), and Gemma 2 9B. Although these models did not reach the performance of state-of-the-art LS system by Enomoto et al. (2024), which used GPT-4 along with ensembling and reranking, Gemma 2 9B was selected due to its relatively small size and balanced high performance across languages.

For selecting suitable lightweight models, we initially considered Gemma 2 2B, Llama 3.2 1B, Qwen 2.5 1.5B, and Qwen 2.5 0.5B due to their multilingual support and small model size. Gemma 2 2B was excluded due to its latency on the m6g.xlarge instance, where the base model with 5-shot setting required 139 ms/token for reading and 476 ms/token for prediction, which was not sufficient for practical use. For the remaining LLMs, we evaluated performance on development set across both the base and instruction-tuned models under three settings: 0-shot, 5-shot, and knowledge distillation. In the 0-shot setting, the prompt was created by removing *{5-shot examples}* from the prompt in Table 6.

Inference	
Parameter	Value
Decoding	Greedy
Sampling	Disabled
Temperature	1.0
Max generation length	10
Training	
Parameter	Value
Optimizer	AdamW
Weight decay	0.01
Learning Rate	3e-5
Scheduler	Linear
Batch Size	16
Max Epoch	5
Lora r	8
Lora alpha	4
Lora dropout	0.1

Table 8: Hyperparameters for training and inference.

In the results in Table 9, the following trends were observed. Firstly, Qwen 2.5 0.5B consistently showed poor performance across all settings. Next, for other models, the 5-shot setting generally outperformed 0-shot. Lastly, while instruction-tuned models slightly outperformed base models in the 5-shot setting, the base models achieved better performance in the knowledge distillation setting. Based on these results, we selected Qwen 2.5 1.5B and Llama 3.2 1B as representative models. To ensure a fair comparison of the proposed methods, we used the base models for both 5-shot and knowledge distillation settings.

D Japanese Tokenization

Since Japanese does not use spaces to separate words, tokenization is required to extract individual words. We primarily used MeCab for tokenization. However, considering the characteristics of the target words in MultiLS, we applied the following rules to select candidate words during data synthesis: (1) Consecutive nouns were grouped together

Model	IT	Settings	English		Spanish		Catalan		German		Japanese	
			ACC	POT	ACC	POT	ACC	POT	ACC	POT	ACC	POT
GPT-4	-	-	.522	.833	.578	.844	.489	.767	.544	.800	.478	.722
Qwen 2.5 14B	✓	5-shot	.511	.767	.444	.778	.289	.600	.400	.611	.244	.489
Phi 3 medium (14B)	✓	5-shot	.467	.733	.478	.733	.200	.467	.367	.567	.278	.433
Gemma 2 9B	✓	5-shot	.489	.700	.422	.711	.333	.611	.478	.689	.200	.444
Qwen 2.5 1.5B		0-shot	**.311**	.467	.089	.167	.044	.111	.000	.056	.067	.122
	✓	0-shot	.289	**.489**	**.333**	**.544**	**.067**	**.200**	**.111**	**.200**	**.067**	**.178**
		5-shot	.300	.522	.300	**.511**	.067	**.244**	**.178**	**.289**	**.089**	.144
	✓	5-shot	**.333**	**.533**	**.322**	.500	**.078**	**.244**	.144	.256	**.089**	**.178**
		fine-tuned	**.344**	.533	**.378**	**.611**	**.144**	**.256**	**.144**	**.267**	**.089**	**.211**
	✓	fine-tuned	**.344**	**.567**	.278	.433	.022	.133	.089	.144	.033	.111
Llama 3.2 1B		0-shot	.022	.022	.000	.044	.000	.011	.000	.000	.011	.011
	✓	0-shot	**.211**	**.378**	**.078**	**.189**	.000	**.022**	**.056**	**.089**	**.044**	**.078**
		5-shot	.211	.300	.022	.078	**.033**	**.144**	**.089**	**.122**	**.056**	.078
	✓	5-shot	**.289**	**.533**	**.233**	**.356**	**.033**	.122	**.089**	**.122**	**.056**	**.111**
		fine-tuned	**.444**	**.622**	**.367**	**.544**	**.167**	.289	**.189**	.244	**.122**	**.200**
	✓	fine-tuned	.422	**.622**	.267	.478	.122	**.333**	.156	**.256**	.022	.156
Qwen 2.5 0.5B		0-shot	.144	.178	.056	.111	**.011**	**.044**	**.011**	**.033**	**.022**	**.056**
	✓	0-shot	**.156**	**.233**	**.111**	**.233**	**.011**	**.044**	**.011**	**.033**	.000	.011
		5-shot	.033	.067	.022	.056	**.011**	**.011**	**.011**	**.044**	.033	**.067**
	✓	5-shot	**.144**	**.244**	**.089**	**.133**	.000	**.011**	**.011**	.022	**.044**	**.067**
		fine-tuned	.200	.344	**.189**	**.311**	**.033**	**.067**	**.044**	**.056**	**.067**	**.111**
	✓	fine-tuned	**.267**	**.389**	.156	.256	.000	.022	.022	.022	.056	**.111**

Table 9: Performance on MultiLS across various models and settings. For the performance of GPT-4, we used outputs of Enomoto et al. (2024). Checkmarks on the IT column refer to the performance from instruction-tuned version. **Bold** numbers are the better scores between the base and instruction-tuned models under the same setting. Underlined numbers are the best performance among 0-shot and 5-shot settings. Red numbers are the best performance across all settings.

Context: An ingenious alphabet allowed the Maya to record information on their monuments and temples, giving anthropologists an excellent way to learn about Maya life and culture.
Target Word: ingenious

Alternative	*GE*	*CM*	*MD*	*GB*	Group
innovative					Good
innovatively	✓				Degraded
sophisticated					Good
adroit			✓		Degraded
anonymous		✓			Degraded
anonymously				✓	Gibberish
simple		✓			Degraded
simply	✓	✓			Degraded

Table 10: Example tags provided to annotators.

as a single unit; (2) For inflected parts-of-speech such as verbs and adjectives, auxiliary verbs were included along with the word stem. It should be noted that the above rules may not always yield exact matches, as the dataset includes multi-word expressions as target words.

E Manual Evaluation Examples

Table 10 shows examples of harmfulness tags assigned to alternatives. These are provided to annotators as reference.

In this example, "ingenious" is the target word to be simplified. While "sophisticated" and "innovative" are appropriate simplifications, other alternatives are harmful. Although replacing "ingenious" with "sophisticated" makes the sentence ungrammatical due to the article-adjective agreement (an sophisticated), such an inconsistency is not considered a harmful simplification in our evaluation.

F Simplification Examples

Table 11 presents examples for languages other than English.

In the Spanish example, the target word "desequilibrado" *(not in equilibrium)* was simplified to "equilibrado" *(in equilibrium)* by Llama under both 5-shot and fine-tuned settings, which reversed the meaning of the context. These harmful outputs had high log-probability scores, which made them difficult to eliminate through the filtering strategy. On the other hand, the teacher model produced a

beneficial output, but its low log-probability would likely lead to its removal.

In the Catalan example, fine-tuned Llama created an adverb-looking word combining the word "mal" *(bad)* with a replication of adverbial suffixes "-ment". This output is clearly *Gibberish*, and similar cases were observed multiple times in the fine-tuned model. Such outputs need to be removed, and the filtering strategy is likely to be effective in achieving this.

In the German example, the output from Gemma 5-shot and Llama 5-shot were assigned *Grammar Error*, while the output from Llama fine-tuned was assigned *More Difficult*. For Llama 5-shot, a noun was proposed while the output should be an adjective as with the target word. This suggests that the system failed to fully understand the task of providing a contextually appropriate word. In German, capitalized words indicate nouns. However, due to the auto-regressive nature of the output, previously generated tokens cannot be revised. General methods such as beam search can mitigate this issue, but they are not applicable to real-time generation, and thus solutions will rely on strategies during training. For the teacher model, grammatical agreement requires "grundlegender" rather than the output "grundlegende". The output is nearly correct, and a finer-grained language-specific tags may be needed for further analysis. The output from Llama fine-tuned fits the and preserves the intended meaning, but the word appears to be an invented term. *More Difficult* was assigned to this output, and its low log-probability suggests that this kind of words could be filtered out.

Lastly, in the Japanese example, both outputs from Qwen were assigned *Grammar Error*. Both systems attempted to produce the appropriate verb "使う", but the Qwen 5-shot output contains an incorrect inflection, while the Qwen fine-tuned output lacks an inflectional suffix. These outputs have relatively high log-probabilities, and therefore it is difficult to filter them out.

G Probability Scores across Categories

Table 12 shows the distribution of harmful tags and their average log-probabilities for each model and language. *More Difficult* is generally rare, but the distribution of other tags varies across models and languages. As mentioned in § 5.2, the average log-probability score of *Grammar Error* from small LLMs in German and Japanese are higher, often comparable or sometimes superior to that of entire outputs. This trend is not pronounced in other languages or from the teacher model.

Spanish

Context: Pero si eso ocurre habitualmente, tienes un flujo de fondos negativo y tu presupuesto está desequilibrado.
(But if that happens habitually, you have a negative cash flow and your budget is not in equilibrium.)
Target Word: desequilibrado *(not in equilibrium)*
Gold Alternatives: inestable *(unstable)*, desnivelado *(uneven)*, desbalanceado *(unbalanced)* ...
Gemma 5-shot (5%): desbalanceado *(unbalanced)* (`Beneficial (POT)`)
Llama 5-shot (55%): equilibrado *(in equilibrium)* (*Change of Meaning*)
Llama fine-tuned (77%): equilibrado *(in equilibrium)* (*Change of Meaning*)

Catalan

Context: En el manifest s'ha qualificat "d'escandalosa" la sentència contra els membres de "la Manada" ja que "se'n riu i menysprea una dona jove" que va ser agredia "brutalment per un grup de salvatges".
(In the statement, the sentence against the members of "la Manada" was described as "scandalous" since "laughs at and despises a young woman" who was assaulted "bruttally by a group of savages".)
Target Word: brutalment *(bruttally)*
Gold Alternatives: violentament *(violently)*, fortament *(strongly)*, durament *(severely)* ...
Gemma 5-shot (51%): violentament *(violently)* (`Beneficial(POT)`)
Llama 5-shot (41%): brutalment *(bruttally)* (`Unchanged`)
Llama fine-tuned (8%): malamentamentamentamentamentamentamentament (*Gibberish*)

German

Context: Salzborn nennt als in die moderne Begriffsgenese von Demokratie eingeschriebene Werte: (...) und die Gewähr elementarer Rechte der Menschen gegen den Staat.
(Salzborn names as values inscribed into the modern conceptual genesis of democracy: (...) and the guarantee of elementary rights of human beings against the state.)
Target Word: elementarer *(elementary)*
Gold Alternatives: grundlegender *(fundamental)*, wichtiger *(important)*, essentieller *(essential)* ...
Gemma 5-shot (57%): grundlegende *(fundamental)* (*Grammar Error*)
Llama 5-shot (35%): Grundrecht *(fundamental right)* (*Grammar Error*)
Llama fine-tuned (9%): grundstehender *(ground-standing)* (*More Difficult*)

Japanese

Context: 迅速に適切な解決を図るために、相談窓口を活用することをお奨めします。
(To ensure a prompt and appropriate resolution, we recommend utilizing the consulation service.)
Target Word: 活用する *(utilizing)*
Gold Alternatives: 使う *(use)*, 利用する *(make use of)*, ...
Gemma 5-shot (97%): 利用する *(make use of)* (`Beneficial (ACC)`)
Qwen 5-shot (63%): 使おう (*Grammar Error*)
Qwen fine-tuned (76%): 使 (*Grammar Error*)

Table 11: Example outputs from the LS systems. Percentages next to system names indicate log-probability percentiles within each system.

Tags	English #	English Logprob	Spanish #	Spanish Logprob	Catalan #	Catalan Logprob	German #	German Logprob	Japanese #	Japanese Logprob
Gemma-5shot										
(All)	100	-1.615	100	-1.567	100	-1.679	100	-1.588	100	-2.268
More Difficult	4	-1.905	2	-1.989	0	-	1	-1.300	4	-2.031
Change of Meaning	14	-1.874	2	-1.266	6	-1.907	6	-1.620	4	-2.447
Grammar Error	1	-2.158	2	-1.409	3	-1.836	7	-1.835	10	-2.528
Gibberish	3	-2.056	1	-2.944	3	-2.227	1	-1.975	2	-3.617
Qwen-5shot										
(All)	100	-1.884	100	-2.129	100	-3.592	100	-2.754	100	-4.132
More Difficult	2	-2.203	1	-3.013	0	-	3	-3.217	3	-3.882
Change of Meaning	20	-2.088	8	-2.957	24	-3.976	20	-3.834	23	-4.766
Grammar Error	3	-2.339	12	-2.469	24	-3.927	20	-3.254	15	-4.220
Gibberish	5	-1.991	0	-	13	-4.440	2	-4.253	2	-5.209
Qwen-fine-tuned										
(All)	100	-1.297	100	-2.063	100	-4.431	100	-3.667	100	-3.337
More Difficult	2	-2.033	0	-	0	-	1	-4.934	1	-3.421
Change of Meaning	14	-1.617	12	-3.001	18	-5.692	34	-4.250	21	-3.697
Grammar Error	5	-1.514	9	-3.390	17	-5.161	10	-3.296	17	-3.018
Gibberish	7	-1.408	4	-5.029	21	-6.047	9	-5.206	37	-4.021
Llama-5shot										
(All)	100	-1.807	100	-1.244	100	-2.873	100	-3.135	100	-4.204
More Difficult	3	-1.603	1	-1.802	0	-	2	-4.501	0	-
Change of Meaning	14	-2.045	8	-1.573	32	-3.246	13	-3.537	16	-4.520
Grammar Error	0	-	3	-1.851	28	-3.374	14	-3.275	19	-3.011
Gibberish	12	-1.604	1	-2.686	6	-3.517	13	-3.375	31	-6.016
Llama-fine-tuned										
(All)	100	-1.161	100	-1.862	100	-2.880	100	-3.645	100	-3.360
More Difficult	0	-	0	-	0	-	4	-4.867	3	-4.091
Change of Meaning	11	-1.465	16	-2.720	17	-3.012	25	-4.147	20	-3.762
Grammar Error	3	-1.764	6	-2.834	13	-3.260	14	-3.304	12	-2.992
Gibberish	6	-1.697	3	-2.219	18	-4.415	15	-4.918	35	-4.457

Table 12: Average log-probability scores for each language and harmful tag.