Edit Policy-Aligned Text Simplification

Xuanxin Wu[1], Yuki Arase[2], Masaaki Nagata[3]
[1]The University of Osaka, [2]Institute of Science Tokyo, [3]NTT Inc.
xuanxin.wu@ist.osaka-u.ac.jp, arase@c.titech.ac.jp
masaaki.nagata@ntt.com

Abstract

Text simplification aims to modify documents and sentences to make them easier to read and understand while preserving their meanings. Different applications require distinct simplification policies, e.g., lexical-oriented simplification and overall rewriting while trading off detail for simplicity. However, achieving such policy-driven control remains an open challenge. In this work, we introduce a simple yet powerful approach that leverages Large Language Model-as-a-Judge (LLM-Judge) to automatically construct policy-aligned training data, removing the need for costly human annotation or parallel corpora. Our method enables the construction of simplification models that adapt to diverse simplification policies at both the sentence and document levels. Remarkably, small-scale open-source LLMs match or surpass GPT-5, as verified by both automatic metrics and edit analysis. The consistent improvements across model families and sizes demonstrate the robustness of our approach.[1]

1 Introduction

Text simplification benefits users with reading difficulties, such as second-language (L2) learners and people with reading impairments (e.g., dyslexia) (Alva-Manchego et al., 2020b). It involves a series of edits, and document- and sentence-level simplification naturally require different editing strategies. Sentence-level simplification policy can vary from lexical paraphrasing, deletion, and sentence splitting, while document-level simplification further needs content reordering, discourse-level rewriting, and elaboration on uncommon topics (Xu et al., 2015; Wilkens et al., 2020; Vásquez-Rodríguez et al., 2023; Sun et al., 2021; Laban et al., 2023). The preferred edit policy, i.e., permissible or appropriate edits in given texts, varies significantly depending on the target audience and applications. In this study, we focus on simplification for L2 education and learning, which has been one of the primary applications of simplification techniques. In this domain, adaptation to a specific edit policy is crucial to achieve effective learning and satisfy teachers' educational goals. However, such adaptation has been out of scope in previous studies.

Recent studies show that large language models (LLMs) such as OpenAI's ChatGPT (OpenAI, 2023) achieve superior performance than previous fine-tuned sequence-to-sequence models through prompt engineering alone, and often generate a mixture of diverse edit types (Kew et al., 2023; Heineman et al., 2023). However, the use of closed-source LLMs in language education is constrained by limited transparency and controllability. Small-scale open-source LLMs present a more feasible option, yet adapting them with policy-driven simplification remains challenging. Key obstacles include: (1) the intrinsic limitations of LLMs, particularly smaller models, which are strong in overall quality but insensitive in following specific edit policies (Barayan et al., 2025); and (2) the scarcity of policy-specific parallel simplification corpora. Manual construction of such a parallel corpus is very expensive. No existing studies provide an efficient way, in terms of both data and computational demands, for building simplification models adapted to predefined edit policies.

Reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) has proven effective for aligning LLMs with human values. RLHF leverages human preference data rather than parallel corpora. However, collecting human preferences at scale is still costly. Alternatively, LLM-Judge can provide scalable and explainable feedback (Kocmi and Federmann, 2023; Song et al., 2024; Niu et al., 2024). Building on this, reinforcement learning from AI feedback (RLAIF) (Bai et al., 2022) ap-

[1]Our code and data will be available at https://github.com/WuXuanxin/edit_policy_simp.

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 3296–3330
November 6-10, 2026 ©2026 Association for Computational Linguistics

pears promising to replace human preference with preferences generated by off-the-shelf LLMs (Tunstall et al., 2023; Cao et al., 2024; Lee et al., 2024).

In this work, we introduce a framework for policy-aligned simplification that requires neither parallel corpora nor human supervision, while remaining computationally efficient with smaller LLMs. We design four edit policies for sentence- and document-level, spanning lexical level, overall rewriting, and elaboration. Our approach guides LLM-Judge to generate high-quality preference data aligned with the edit policies. These data are then used to fine-tune open-source models, including Phi-3-mini-3.8B (Abdin et al., 2024), Qwen2.5-7B (Qwen Team, 2025), Llama3.1-8B (Grattafiori et al., 2024), and Qwen2.5-14B (Qwen Team, 2025), via light-weight preference optimization algorithms (Xu et al., 2024, 2025). Our approach substantially improves the policy alignment capabilities of small-scale LLMs, enabling them to match or even surpass GPT-5 [2], as verified by both automatic simplification metrics and edit-type analysis.

2 Related Work

The standard approach in previous studies has been supervised training with complex-simple parallel corpora. While remarkable progress has been achieved in simplification quality, adaptation to various policies has been out of the scope for both sentence and document simplification. Different from these studies, our method does not depend on any parallel corpora and achieves flexible adaptation to various edit-policies.

2.1 Sentence Simplification

Martin et al. (2020) showed that simple surface-level attributes such as sentence length or lexical difficulty can be controlled through prepended control tokens to the input. The control of difficulty levels has also been explored, which aims to simplify sentences to be appropriate for the target audience of the specific proficiency levels (Scarton and Specia, 2018; Horiguchi et al., 2024; Li et al., 2025). With the advent of LLMs, prompt-based techniques have largely surpassed earlier sequence-to-sequence model-based methods in overall simplification quality (Kew et al., 2023; Wu and Arase, 2026). However, prompting offers limited sensitivity to edit policy adaptation, particularly with smaller LLMs (Barayan et al., 2025). Recent studies aim to improve edit policy alignment via fine-tuning, however, the policy-specific training signals still largely depend on human supervision (Gao et al., 2026).

2.2 Document Simplification

Document simplification has been constrained by the scarcity, quality, and diversity of copora. Existing resources are largely based on Simple English Wikipedia (SEW)[3] and Newsela (Alva-Manchego et al., 2020b), but SEW-based data often requires substantial alignment and can be noisy (Xu et al., 2015; Laban et al., 2023). Earlier non-LLM approaches trained on these corpora (Blinova et al., 2023; Cripwell et al., 2023a; Vásquez-Rodríguez et al., 2023) show limited performance and generalization (Bakker and Kamps, 2024). The task is further complicated by the need for coordinated edits across sentences to preserve coherence and meaning (Wilkens et al., 2020; Vásquez-Rodríguez et al., 2023; Laban et al., 2023). Recent work shows that LLM-based prompting outperforms earlier non-LLM systems for document simplification (Fang et al., 2025; Qiang et al., 2025), and LLM outputs can even surpass existing human-written references (Qiang et al., 2025). However, it is also reported that direct prompting often leads to information loss and deletion-heavy behavior (Nandiraju et al., 2025; Fang et al., 2026).

3 Edit-Policy Aligned Simplification

3.1 Problem Definition

We train a simplification model using a decoder-only language model π_θ parameterized by θ. Let $\mathcal{P}$ denote a set of simplification policies and $\mathcal{X}$ be a finite set of source texts, either sentences or documents. For $x \in \mathcal{X}$, let $y^\star(x, p)$ be the optimal simplification under a policy $p \in \mathcal{P}$. Our goal is to learn π_θ such that

$$\max_\theta \ \mathbb{E}_{x\sim\mathcal{X}}\big[\log \pi_\theta\big(y^\star(x,p) \mid x\big)\big]. \qquad (1)$$

Because $y^\star(x, p)$ is latent, prior work approximated it relying on human-written simplifications and optimized π_θ through supervised fine-tuning (Scarton and Specia, 2018; Martin et al., 2020; Cripwell et al., 2023b). In contrast, we build a preference dataset by LLMs, and optimize π_θ using preference optimization.

[2] We used `gpt-5-2025-08-07` through the OpenAI API.

[3] https://simple.wikipedia.org/wiki/Main_Page

Source	Johan was also the original bassist of the Swedish power metal band HammerFall, but quit before the band ever released a studio album.
Sent-Overall	Johan was ~~also~~ the original bassist for HammerFall, a Swedish power metal band. He left before they released their first ~~studio~~ album.
Sent-Lexical	Johan was also the original bassist of the Swedish power metal band HammerFall, but he left before they released any studio albums.

Table 1: Sentence simplifications by our model under two edit policies (Phi-3-mini-3.8B (Abdin et al., 2024)). We highlight the main simplification edits in each part of the sentence using different colors. Green: Paraphrasing Red: Delete Blue: Structural Change

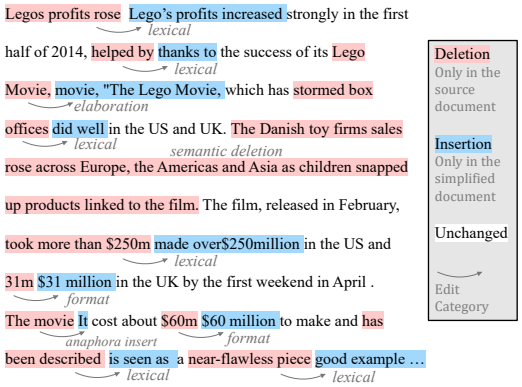


Figure 1: Document simplifications by our model under the Doc-Lexical policy (Llama3.1-8B (Grattafiori et al., 2024)). Following Laban et al. (2023), we visualize the edits applied to the source document.

3.2 Edit Policies

In this study, we focus on L2 education and design edit policies informed by research on second language acquisition and language education. All policies are intentionally differentiated by their edit priorities rather than by mutually exclusive edit boundaries, as different edit operations are often interdependent and strict separation may compromise overall text quality.

3.2.1 Sentence Level Policy

In L2 education, sentence-level simplification is desired because teachers can easily identify where and how a text has been modified. Previous work in both NLP and language education research has shown that the desired type and degree of simplification edits change depending on learner proficiency and readability levels (Agrawal et al., 2021; Zhong et al., 2020). Specifically, low- to intermediate-level learners benefit from a combination of lexical paraphrasing, structural modifications, and selective deletions to reduce cognitive load. In contrast, advanced learners benefit from lexical paraphrasing, which supports vocabulary acquisition (Chen, 2019), but they gain comparatively less from added cohesion or deletion (Hosoda, 2016; Zhong et al., 2020).

Motivated by these findings, we introduce two distinct sentence-level edit policies. As illustrated in Table 1, **Sent-Overall** simplification often combines lexical paraphrasing and structural modifications, and deletions to improve readability for intermediate-level language learners. In contrast, **Sent-Lexical** (Paetzold and Specia, 2016; Li et al., 2025) adheres to the original sentence closely while supporting more efficient vocabulary acquisition for advanced learners.

3.2.2 Document Level Policy

Prior work shows that lexical support in context improves comprehension and vocabulary learning (Chen, 2019; Yanagisawa et al., 2020), while document comprehension also depends on discourse-level factors (Horiba, 1996). This suggests that document-level simplification for L2 learners should combine discourse-level rewriting with lexical support, with greater emphasis on lexical accessibility. Furthermore, as an orthogonal approach, elaboration is effective to complement the necessary knowledge to understand a content. Elaborative edits, i.e., adding clarification and background information, can help reduce readers' cognitive load and improve reading comprehension (Oh, 2001; Srikanth and Li, 2021).

Motivated by these findings, we introduce two document-level edit policies. **Doc-Lexical** prioritizes lexical support (see Figure 1), whereas **Doc-Elaboration** places greater focus on elaboration.

3.3 Proposed Method

Figure 2 illustrates the overview of our method, which consists of three steps as described below.

3.3.1 Step 1: Create Candidate Pool

We begin with a collection of N source texts $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$, and use LLMs to generate candidate simplifications. Diversity is crucial to effectively distinguish preferred outputs that align with the target policy from those that deviate from it. Previous studies have shown that models of different families and sizes exhibit distinct performance and editing behavior (Heineman et al., 2023; Kew et al., 2023; Wu and Arase, 2026). Motivated

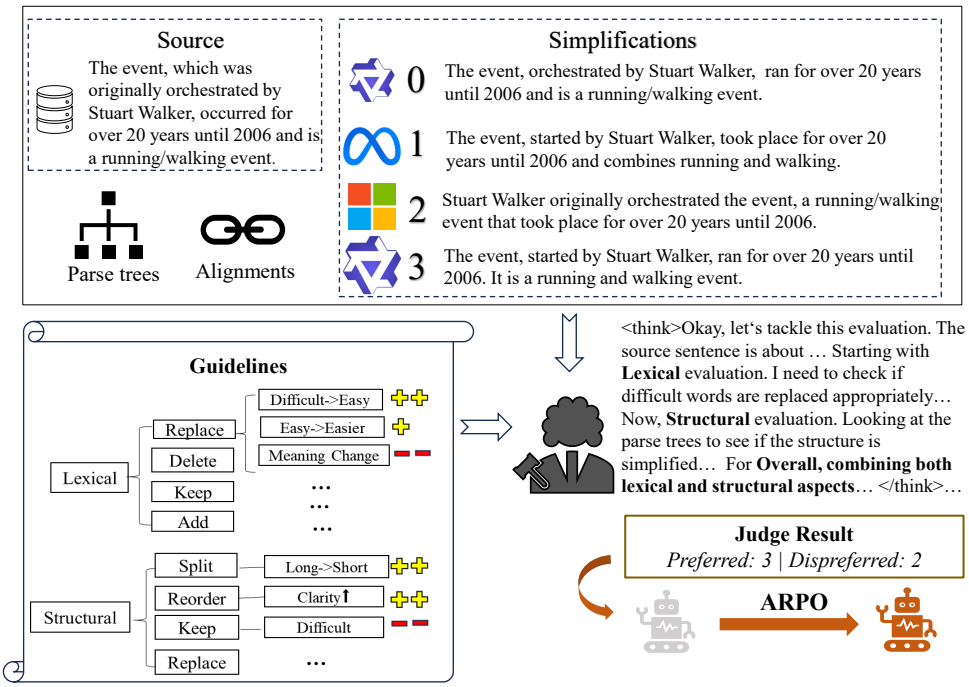


Figure 2: Overview of our sentence-level simplification; the same framework can be applied to the document level.

by this, we construct the candidate pool using a set of K LLMs, $\mathcal{M} = M_1, \ldots, M_K$, varying from different families and sizes. For each source text $x_i \in \mathcal{X}$ and policy p, every model $M_k \in \mathcal{M}$ generates one candidate simplification $y_{i,k}$, yielding a total of K candidates per source text. The simplification pool for each x_i is defined as:

$$\mathcal{C}(x_i) = \{y_{i,k} \mid y_{i,k} \sim M_k(x_i, s_p)\}, \quad (2)$$

where s_p is the natural-language instruction that describes policy p.

3.3.2 Step 2: LLM-Judge

We employ a reasoning LLM as the judge, selected for its strong performance on complex judge tasks (Huang et al., 2026). The model follows carefully designed guidelines for policies. The corresponding prompts are provided in Appendix A.4.1.

Sentence Level Judgement The Sent-Lexical policy encourages (a) minimal edits to preserve sentence structure and (b) replacing complex words with simpler alternatives. The Sent-Overall policy encourages broader edits at both lexical and structural levels to enhance simplicity. These nuances can be included in prompts, allowing LLMs to follow them with ease. The model is guided by carefully designed principles that specify edit types, their effects, and associated rewards or penalties.

- **Lexical Principles:** We define four edit operations: *replace*, *delete*, *keep*, and *add*. E.g., simplifying a complex word receives a high reward, while replacing an already simple word yields only a moderate reward.
- **Structural Principles:** Similarly, we define four edit operations: *split*, *reorder*, *keep*, and *replace*. Structural transformations are rewarded if they improve readability or conciseness without changing meaning. Unnecessary or unhelpful modifications are penalized.

To support these judgments, we provide the lexical judge with word alignment between the source and simplifications derived using OTAlign (Arase et al., 2023) and the structural judge with syntactic parse trees for each sentence extracted using Qwen3-32B (Yang et al., 2025), respectively. This evidence-guided design improves judge transparency and generally enhances downstream performance. See Appendix A.5.1 for details.

Document Level Judgement Both of the document-level policies, Doc-Lexical and Doc-Elaboration, involve a mixture of edit types, but differ in the edits they prioritize. Assuming that it is difficult for users to verbalize the appropriate compositions of edit types, we introduce an *anchor* example for each policy. The use of behavioral anchors has a long history in human evaluation

and rating-scale design. Smith and Kendall (Smith and Kendall, 1963) introduced a procedure for constructing rating scales anchored by concrete examples of expected behavior, which became foundational to Behaviorally Anchored Rating Scales (BARS). The underlying idea is to make abstract evaluation criteria more concrete through representative examples. Inspired by this idea, we use an anchor example to concretely instantiate the editing strategy associated with each policy for our LLM-Judge. Specifically, the anchor illustrates both (a) which edit categories are typically applied and (b) how frequently and strongly they are used under the target policy. To support the decision, we provide edit information for the anchor using a document-level edit classifier (Laban et al., 2023). Judges are then required to compare candidate simplifications with the anchor and assess how well their editing strategies align with the target policy, while also considering document-level qualities such as coherence and coreference.

Preference data For each source text $x_i \in \mathcal{X}$, the LLM-judge selects a preferred candidate $y_w^{(i)}$ and a dispreferred candidate $y_l^{(i)}$ from the candidate pool $\mathcal{C}(x_i)$ according to our guidelines $\mathcal{G}$:

$$(y_w^{(i)}, y_l^{(i)}) = J(x_i, \mathcal{C}(x_i, p), \mathcal{G}). \quad (3)$$

This procedure yields a preference dataset $\mathcal{D} = (x^{(i)}, y_w^{(i)}, y_l^{(i)})_{i=1}^{N}$ for each policy p.

3.3.3 Step 3: Preference Optimization

Preference optimization has emerged as a powerful post-training paradigm for aligning LLMs with human preferences, typically formatted as `{input, preferred output, dispreferred output}`. We adopt Adaptive Rejection Preference Optimization (ARPO) (Xu et al., 2025) to optimize π_θ with our preference dataset $\mathcal{D}$. ARPO is a variant of memory-efficient Contrastive Preference Optimization (CPO) (Xu et al., 2024) designed to mitigate its tendency to overly penalize dispreferred responses that are only marginally worse than preferred ones. We integrate the Simple Preference Optimization (SimPO) (Meng et al., 2024) loss into ARPO to improve the performance.[4] See Appendices A.1 and A.2 for detailed preliminaries and experimental settings.

[4] https://github.com/fe1ixxu/CPO_SIMPO

4 Evaluation

4.1 Preference Data Construction

Candidate Generation We construct the preference optimization dataset using *only* the source side texts from existing simplification corpora. For sentence-level simplification, we use CoEdit (Raheja et al., 2023), which aggregates multiple existing simplification parallel corpora. For document-level simplification, we aggregate source documents of Newsela (Xu et al., 2015)[5] and SWiPE (Laban et al., 2023). For each source text, we collected outputs from four instruction-tuned LLMs: Qwen2.5-7B, Llama3.1-8B, Phi4-14B, and Qwen3-32B. Prompts used to generate simplifications are described in Appendix A.4.2. To ensure meaningful simplification, we apply heuristic filtering, such as removing very short source texts that leave little room for editing.

LLM-Judge We selected Qwen3-32B as the LLM judge based on its strong judging performance in our preliminary experiments. Although Qwen3-32B also serves as a candidate generation model, we did not observe a consistent tendency for it to prefer its own generations. In the sentence-level evaluation, the Qwen3-32B judge in fact prefers outputs from Phi4-14B more frequently. Appendix A.7.1 provides detailed results and discussion. To mitigate potential positional bias in LLM-based evaluation (Wang et al., 2024a), we randomize the presentation order of generated candidates before passing them to the judge model. Consequently, for each policy we obtain a preference dataset consisting of triplets of the form `{source, preferred simplification, dispreferred simplification}`.

Constructed Data The sentence-level dataset contains 8K triplets, split into 7K for training and 1K for development. For each document-level policy, we obtained 4K triplets, split into 3.5K training instances and 500 development instances.

4.2 Evaluation Datasets and Metrics

Sentence-level Datasets and Metrics For **Sent-Lexical** policy, we used the **Turk** test set (Xu et al., 2016), containing 359 source sentences paired with 8 crowdworker-written simplification references,

[5] We only used documents from the highest reading level. Newsela can be requested for academic use at https://newsela.com/legal/data.

constructed specifically for lexical-based simplification. We accordingly use **SARI** (Xu et al., 2015) as the evaluation metric because it explicitly assesses lexical edit (keep, add, delete) quality by comparing system outputs with both references and sources. For **Sent-Overall** policy, we used the **ASSET** test set (Alva-Manchego et al., 2020a). While ASSET shares the same source sentences as Turk, it differs in its edit policy: more diverse edits, i.e., paraphrasing, deletion, and sentence splitting. Each source sentence pairs with 10 crowdworker-written references. To capture this broader range of edits, we evaluated with **LENS** (Maddela et al., 2023). LENS is trained directly on human ratings of overall simplicity quality and has shown strong correlation with overall simplicity quality (Huang and Kochmar, 2024; Wu and Arase, 2026).

Document-Level Datasets For **Doc-Lexical**, we use **OneStopEnglish (OSE)** (Vajjala and Lučić, 2018), an expert-written document simplification dataset created for L2 learners. OSE contains 189 source documents, each paired with intermediate- and elementary-level simplifications. For **Doc-Elaboration**, we sample 200 complex–simple document pairs (not used in the preference data) from expert-written **Newsela** (Xu et al., 2015) (size: $1.9k$). Newsela includes simplifications at multiple reading levels. We use the elementary-level OSE and the middle-level Newsela as references, as we found that they consistently require lexical and elaboration edits while better balance between meaning preservation and simplicity.

Document-Level Metrics We use **Agg-LENS** (Maddela and Alva-Manchego, 2025), a document simplification metric that shows higher correlation with human judgments than earlier metrics, e.g., D-SARI (Sun et al., 2021). Agg-LENS adapts the sentence-level LENS metric to the document level through graph-based alignment. We also report **Agg-BERT** from the same study, which shows high consistency for assessing coherence. Furthermore, because Agg-LENS is not explicitly designed to capture the edit policies studied in this work, we additionally report **edit distribution**. To do so, we use an edit-category classifier[6] (Laban et al., 2023), which was trained on a large collection of human-annotated document-level edits. We measure the percentage of *lexical* edits for Doc-Lexical and *elaboration* edits for Doc-Elaboration. We also report *deletion*, as prior work has shown that LLMs tend to over-delete content (Nandiraju et al., 2025; Fang et al., 2026). These distributions are then compared with those of the reference simplifications. Implementation details are in Appendix A.3.1.

4.3 Comparison

We apply preference optimization to four open-source instruction-tuned LLMs for sentence-level policies, covering different model families and scales: Phi-3-mini-3.8B (`Phi3.8B`), Qwen2.5-7B (`Qwen7B`), Llama3.1-8B (`Llama8B`), and Qwen2.5-14B (`Qwen14B`). For document-level, we use Qwen7B and Llama8B.[7]

Prior work has shown that LLMs outperform earlier simplification systems on both sentence- and document-level simplification (Kew et al., 2023; Wu and Arase, 2026; Qiang et al., 2025). We therefore compare our method against LLM-based baselines. LLMs are known to be sensitive to prompting. We evaluate using diverse prompting settings: **zero-shot (zs), few-shot (fs), and iterative feedback (iter)**. The **zs** and **fs** prompts adhere to policies and the editing behaviors observed in human references (see Appendix A.4.2). **Iter** combines automatic evaluation with self-correction: using the same LLM-Judge and evaluation criteria as in our framework, we first generate feedback on a model's output and then prompt the model to revise its simplification accordingly for two rounds (*Iter 1* and *Iter 2*). Prompts for the **iter** setting are provided in Figures 15 to 20 (Appendix). We use **zs** prompts for all fine-tuned models, while applying multiple prompting strategies to the vanilla LLMs.

Common Baseline For both sentence- and document-level simplification, we compare our method (**PO**) against the following baseline:

- **Base models (Vanilla)**: Instruction-tuned LLMs used directly without additional training. We use **zs**, **fs**, and **iter** prompts.
- **GPT-5**: A state-of-the-art proprietary LLM that performs strong on simplification (Wu and Arase, 2026). We use both **zs** and **fs**.
- **SFT on human-written corpora (Parallel)**: Models fine-tuned on policy-aligned parallel corpora written by human, representing the

[6] https://github.com/salesforce/simplification

[7] Phi3.8B supports only a 4K context window, which limits its applicability to document-level simplification. Training Qwen14B on long documents is challenging given our available computational resources.

scenario where such data is available. We use dev sets of Turk and ASSET (size: $2k$) for Sent-Lexical and Sent-Overall, respectively.[8] For Doc-Elaboration, we use the remaining 1.7k parallel data from Newsela. We do not include the same setting for Doc-Lexical because no training data is available.

Sentence-level Baseline To investigate the effect of our edit-policy-oriented preference optimization corpus created by LLM-Judge, we compare to LENS-SALSA Preference Optimization **(LENS_SALSA)**. LENS-SALSA (Heineman et al., 2023) is a simplification metric trained on large-scale fine-grained human annotations to approximate LENS scores in a reference-free manner. We use the LENS-SALSA score to construct the preference dataset; candidates with the highest scores are treated as preferred, whereas those with the lowest scores are treated as dispreferred.

Document-level Baseline For document-level, we further compare to **AgentSimp** (Fang et al., 2025), an LLM-based multi-agent framework for document simplification built on ChatGPT and reported to outperform the ChatGPT baseline.[9]

4.4 Evaluation Results

Results are reported in Tables 2 and 3. Our method yields consistent gains across edit policies, model architectures, model sizes, on both sentence- and document-level simplification. Simplification examples are in Tables 12 and 13 (Appendix).

Our method enables smaller LLMs to match or surpass GPT-5. Across all policies, our method not only surpasses the vanilla models but also achieves results comparable to or even exceeding GPT-5. At the sentence level, the improvements of Turk-SARI range from $+4.5$ to $+7.8$ points under the same zero-shot setting. Even under the Sent-Overall policy, where LLMs are already strong due to their capacity for diverse edits, our method remains robust with improvements ranging from $+2.3$ to $+5.8$ points. Remind that LENS_SALSA is optimized for LENS score, thus expected to achieve high score on ASSET. In contrast, it struggled with the Turk-SARI that requires lexical-oriented simplification. It is remarkable that our method still achieves comparable performance against LENS_SALSA on ASSET while outperforms it on Turk. A key advantage of our method is its ability to adapt to different policies simply by changing the LLM-Judge prompts. The same trend holds at the document level. On Newsela, both PO models outperform GPT-5, with Qwen7B improving over vanilla zero-shot prompting by $+17.0$ and $+31.4$ on Agg-LENS and Agg-BERT, and Llama8B by $+17.9$ and $+30.5$. On OSE, GPT-5 remains the strongest model, but our method substantially narrows the gap: Qwen7B gains $+12.1$ and $+22.8$, while Llama8B improves by $+6.4$ and $+16.5$, bringing both close to GPT-5.

Model	Method	Turk-SARI	ASSET-LENS
GPT-5	zs	41.2	68.1
	fs	42.2	70.7
Phi3.8B	Vanilla-zs	35.5	68.9
	Vanilla-fs	<u>42.2</u>	70.0
	Vanilla-iter1	38.9	66.6
	Vanilla-iter2	39.8	66.0
	Parallel	36.7	69.6
	LENS_SALSA	41.8	**73.5**
	PO (Ours)	**43.3**	<u>71.4</u>
Qwen7B	Vanilla-zs	38.0	65.0
	Vanilla-fs	41.0	68.4
	Vanilla-iter1	39.7	63.7
	Vanilla-iter2	40.6	63.6
	Parallel	<u>42.4</u>	64.9
	LENS_SALSA	41.5	**72.5**
	PO (Ours)	**43.7**	<u>70.8</u>
Llama8B	Vanilla-zs	39.2	66.9
	Vanilla-fs	40.7	70.0
	Vanilla-iter1	42.4	64.3
	Vanilla-iter2	<u>42.8</u>	64.0
	Parallel	35.4	68.4
	LENS_SALSA	41.7	**73.0**
	PO (Ours)	**43.7**	<u>70.2</u>
Qwen14B	Vanilla-zs	38.7	68.0
	Vanilla-fs	<u>42.6</u>	66.5
	Vanilla-iter1	41.8	64.8
	Vanilla-iter2	42.3	65.7
	Parallel	40.4	68.4
	LENS_SALSA	42.2	**72.4**
	PO (Ours)	**43.9**	<u>70.3</u>

Table 2: Sentence-level results (**bold**: the best under each model group, <u>underline</u>: the second best.)

Prompt engineering is not robust. Prompt engineering alone does not provide consistent gains. We found that sometimes few-shot prompting can degrade performance: for example, -1.6 LENS for Qwen14B on ASSET, -3.2 Agg-LENS for Llama8B on OSE. The gains from few-shot prompting also vary across models: For example, on Turk, it improves Phi3-3.8B by 6.7 SARI but Llama8B by only 1.5. On OSE, it improves Qwen7B by 5.4 points on Agg-LENS, but degrades Llama8B. These suggest that in-context examples can introduce model-specific biases rather than providing reliable guidance. Iterative feedback is also unstable.

[8] We used only one reference per sentence, as preliminary experiments showed performance degradation with multiple references during training.

[9] Because the code and data are not publicly available, we replicate PL†, the best-performing setting reported in the paper, using the algorithms and prompts described by the authors.

Model	Method	OSE LENS	OSE BERT	Newsela LENS	Newsela BERT
GPT-5	zs	54.7	59.4	33.7	25.4
	fs	52.4	56.1	34.9	28.0
Agent-Simp	–	34.0	16.4	44.3	19.9
Qwen7B	Vanilla-zs	41.6	27.3	33.7	20.6
	Vanilla-fs	47.0	40.0	39.6	34.0
	Vanilla-iter1	42.2	28.8	34.7	22.6
	Vanilla-iter2	42.3	29.4	34.8	23.2
	Parallel	–	–	**50.9**	45.4
	PO (Ours)	**53.7**	**50.1**	50.7	**52.0**
Llama8B	Vanilla-zs	47.5	35.0	34.7	25.1
	Vanilla-fs	44.3	33.6	39.5	33.2
	Vanilla-iter1	47.5	35.2	35.4	26.5
	Vanilla-iter2	47.6	35.8	36.2	28.0
	Parallel	–	–	52.4	50.6
	PO (Ours)	**53.9**	**51.5**	**52.6**	**55.6**

(a) Automatic evaluation results. LENS denotes Agg-LENS, and BERT denotes Agg-BERT.

Model	Method	OSE-L	OSE-D	News-E	News-D
human ref	–	26.9	47.4	16.9	48.2
GPT-5	zs	+10.4	−21.7	−3.6	+7.0
	fs	+7.0	−17.8	−3.9	+5.6
Agent-Simp	–	−13.9	+5.9	+16.7	−29.1
Qwen7B	Vanilla-zs	−11.5	+17.4	−9.0	+24.2
	Vanilla-fs	−9.2	+9.5	−6.4	+16.7
	Vanilla-Iter1	−11.4	+16.1	−8.7	+22.7
	Vanilla-Iter2	−11.5	+14.9	−8.5	+22.0
	Parallel	–	–	**−1.5**	**−2.0**
	PO (Ours)	**−2.2**	**−4.4**	−3.0	+5.1
Llama8B	Vanilla-zs	−10.6	+8.0	−6.5	+20.1
	Vanilla-fs	−15.1	+15.5	−6.4	+21.0
	Vanilla-Iter1	−10.6	+6.1	−5.5	+17.7
	Vanilla-Iter2	−10.7	+5.3	−4.6	+15.7
	Parallel	–	–	**−0.7**	**−2.0**
	PO (Ours)	**−2.7**	**−3.6**	−2.5	+2.1

(b) Edit percentage difference (%) relative to the *human ref* values: values closer to zero indicate better performance. Values are computed as model minus reference. L: lexical, D: deletion, E: elaboration

Table 3: Document-level results (**bold**: the best under each model group, underline: the second best.)

At the sentence level, it can improve Turk-SARI, but consistently degrades ASSET-LENS. We find that iterative feedback tends to make models more conservative, leading to fewer meaningful simplification edits. At the document level, iterative feedback brings little to no improvement for either policy. In contrast, after our preference training, all models reach consistently strong performance.

LLM-Judge outperforms crowd-sourced parallel corpus. Our method surpasses SFT trained on the crowd-sourced Turk and ASSET parallel corpora at the sentence level, while remaining comparable with SFT trained on the expert-written Newsela corpus. Two factors contribute: (1) *Scalability:* LLM-Judge enables creating large-scale preference data easily and efficiently, whereas SFT is constrained by the scale of human efforts. (2) *Quality Control:* Human references are not always consistent and may even be surpassed by advanced LLMs (Xu et al., 2024; Liu et al., 2024; Qiang et al., 2025). Appendix A.5.3 further analyzes the effects of data sizes. Even when trained on the same number of samples (2k / 1.5k), our method consistently outperforms the Parallel baseline.

Model	Lex_P	Lex_R	Lex_F1
GPT-5-zs	0.21 (0.09)	0.36 (0.15)	0.27 (0.11)
GPT-5-fs	0.21 (0.08)	0.33 (0.12)	0.26 (0.10)
Agent-Simp	0.22 (0.08)	0.12 (0.05)	0.16 (0.06)
Qwen7B_PO	0.25 (0.11)	0.24 (0.11)	0.24 (0.11)
Llama8B_PO	0.24 (0.11)	0.21 (0.10)	0.22 (0.10)

(a) OSE

Model	Ela_P	Ela_R	Ela_F1
GPT-5-zs	0.06 (0.02)	0.08 (0.02)	0.07 (0.02)
GPT-5-fs	0.06 (0.01)	0.07 (0.02)	0.07 (0.02)
Agent-Simp	0.04 (0.01)	0.15 (0.03)	0.07 (0.02)
Qwen7B_PO	0.11 (0.03)	0.08 (0.02)	0.09 (0.02)
Llama8B_PO	0.11 (0.03)	0.07 (0.02)	0.08 (0.03)

(b) Newsela

Table 4: Precision, recall, and F1 scores of edit patterns (Lex: lexical edits, Ela: elaboration edits). Scores in parentheses are exact-matching.

PO aligns edit distributions more closely with human references and reduces over-deletion. On both OSE and Newsela, our models achieve edit ratios that closely match those of the human references. These results indicate that our method not only improves document-level simplification quality but also effectively steers model behavior toward the desired edit policy. Consistent with prior work (Nandiraju et al., 2025; Fang et al., 2026), vanilla LLMs tend to over-rely on deletion at the document level. Our method substantially alleviates this tendency. Under the same zero-shot prompting setting, on OSE, the deletion rate drops from 64.8% to 43.0% for Qwen7B and from 55.5% to 43.8% for Llama8B, moving both closer to the human reference of 47.4%. On Newsela, it decreases from 72.4% to 53.3% for Qwen7B and from 68.3% to 50.3% for Llama8B, again approaching the reference value of 48.2%. These improvements are also substantially larger than those achieved by prompting strategies.

4.5 Further Analysis

We evaluate LLMs' edit patterns by computing precision, recall, and F1 against human-edit spans.[10] Because semantically equivalent edits sometimes differ in span boundaries (e.g., 'air force' versus 'air force fields'), we report a soft-matching, in which two spans are considered as a match when at least 50% of the words overlap in both spans, in addition to exact matching. Appendix A.3.2 describes details of edit span recognition.

[10]We evaluate in a single-reference setting (see Section 4.2), which may have lowered precision, recall, and F1 scores.

Table 4 shows that our method achieves competitive F1 scores against GPT-5; however, elaboration is clearly more challenging than lexical editing, as reflected in the lower scores on Newsela. This trend is also consistent with the comparison against the parallel baseline in Table 3; under the Doc-Elaboration policy, our method does not outperform the baseline trained on the professionally edited Newsela corpus. Elaboration requires inferring what information to add and how much explanation is appropriate. Although Agent-Simp introduces a general-purpose explainer agent, we find that it often over-explains and adds unnecessary details. Together, these results suggest that effective elaboration requires stronger awareness of the target reader and more explicit modeling of pedagogical judgment.

In addition to the analysis above, experiments in Appendix A.6 demonstrate that our method transfers effectively across domains. Appendix A.5 provides a detailed ablation study.

5 Conclusion and Future Work

We propose a framework for adapting text simplification to various edit policies, which is critical for real-world applications. By leveraging LLM-Judge, our method consistently enhances the policy alignment of small open-source LLMs, achieving comparable or even higher performance than the large proprietary LLMs.

Future work could include a thorough human evaluation to assess whether LLM-generated simplifications meet educational expectations in practice. Moreover, elaboration remains challenging as discussed above. Future work could extend our framework to support finer-grained control by conditioning the LLM judges on target-audience characteristics or reading goals and incorporating expert knowledge into the judging process.

Limitations

First, although we designed several simplification policies for L2 education based on prior work in second language acquisition, their educational effectiveness remains to be validated in real learning settings. In particular, it is necessary to deploy these policies in real educational use and examine their long-term impact on learners' comprehension, vocabulary growth, and reading development.

Second, as discussed in the paper, although our method can steer model editing behavior toward the target policy, a gap still remains between model edits and human edits, especially for elaboration. This gap likely stems from the inherent natures of LLMs, which may not always align with how human experts would perform. Future work could therefore explore more fine-grained control over elaboration, for example, by conditioning on learners' knowledge background.

Acknowledgement

This work was supported by NTT, Inc. and JSPS KAKENHI Grant Number JP25K03233.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Sweta Agrawal, Weijia Xu, and Marine Carpuat. 2021. A non-autoregressive edit-based approach to controllable text simplification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3757–3769, Online. Association for Computational Linguistics.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020a. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. EASSE: Easier automatic sentence simplification evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 49–54.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020b. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics*, 46(1):135–187.

Yuki Arase, Han Bao, and Sho Yokoi. 2023. Unbalanced optimal transport for unbalanced word alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3966–3986, Toronto, Canada. Association for Computational Linguistics.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron

McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, and 32 others. 2022. Constitutional AI: Harmlessness from ai feedback. *Preprint*, arXiv:2212.08073.

Jan Bakker and Jaap Kamps. 2024. Beyond sentence-level text simplification: Reproducibility study of context-aware document simplification. In *Proceedings of the Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC-COLING 2024*, pages 27–38, Torino, Italia. ELRA and ICCL.

Abdullah Barayan, Jose Camacho-Collados, and Fernando Alva-Manchego. 2025. Analysing zero-shot readability-controlled sentence simplification. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6762–6781, Abu Dhabi, UAE. Association for Computational Linguistics.

Sofia Blinova, Xinyu Zhou, Martin Jaggi, Carsten Eickhoff, and Seyed Ali Bahrainian. 2023. SIMSUM: Document-level text simplification via simultaneous summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9927–9944, Toronto, Canada. Association for Computational Linguistics.

Meng Cao, Lei Shu, Lei Yu, Yun Zhu, Nevan Wichers, Yinxiao Liu, and Lei Meng. 2024. Enhancing reinforcement learning with dense rewards from language model critic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9119–9138, Miami, Florida, USA. Association for Computational Linguistics.

Mei-Hua Chen. 2019. Phrasal paraphrase learning: Exploring an effective strategy to consolidate vocabulary knowledge. *Taiwan Journal of TESOL*, 16(1):41–66.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 6621–6642. PMLR.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2023a. Context-aware document simplification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13190–13206, Toronto, Canada. Association for Computational Linguistics.

Liam Cripwell, Joël Legrand, and Claire Gardent. 2023b. Document-level planning for text simplification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 993–1006, Dubrovnik, Croatia. Association for Computational Linguistics.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, and 26 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *Preprint*, arXiv:2412.04261.

Dengzhao Fang, Jipeng Qiang, Xiaoye Ouyang, Yi Zhu, Yunhao Yuan, and Yun Li. 2025. Collaborative document simplification using multi-agent systems. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 897–912, Abu Dhabi, UAE. Association for Computational Linguistics.

Dengzhao Fang, Jipeng Qiang, Yi Zhu, Yunhao Yuan, Wei Li, and Yan Liu. 2026. Progressive document-level text simplification via large language models. *New Generation Computing*, 44(1):4.

Yingqiang Gao, Kaede Johnson, David Fröhlich, Luisa Carrer, and Sarah Ebling. 2026. Evaluating direct preference optimization for personalizing German automatic text simplifications for persons with intellectual disabilities. In *Proceedings of the Sixth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 43–62, Virtual (Online). Association for Computational Linguistics.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

David Heineman, Yao Dou, Mounica Maddela, and Wei Xu. 2023. Dancing between success and failure: Edit-level simplification evaluation using SALSA. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3466–3495.

Yukie Horiba. 1996. Comprehension processes in L2 reading: Language competence, textual coherence, and inferences. *Studies in Second Language Acquisition*, 18(4):433–473.

Koki Horiguchi, Tomoyuki Kajiwara, Yuki Arase, and Takashi Ninomiya. 2024. Evaluation dataset for Japanese medical text simplification. In *Proceedings*

of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop), pages 219–225, Mexico City, Mexico. Association for Computational Linguistics.

Masaya Hosoda. 2016. The interplay of text cohesion and L2 reading proficiency in different levels of text comprehension among EFL readers. *ARELE: Annual Review of English Language Education in Japan*, 27:201–216.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hui Huang, Xuanxin Wu, Muyun Yang, and Yuki Arase. 2026. Reasoning model is superior llm-judge, yet suffers from biases. *Preprint*, arXiv:2601.03630.

Yichen Huang and Ekaterina Kochmar. 2024. REFeREE: A REference-FREE model-based metric for text simplification. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13740–13753, Torino, Italia. ELRA and ICCL.

Tannon Kew, Alison Chi, Laura Vásquez-Rodríguez, Sweta Agrawal, Dennis Aumiller, Fernando Alva-Manchego, and Matthew Shardlow. 2023. BLESS: Benchmarking large language models on sentence simplification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13291–13309.

Tom Kocmi and Christian Federmann. 2023. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation*, pages 768–775, Singapore. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. SOSP '23, page 611–626, New York, NY, USA. Association for Computing Machinery.

Philippe Laban, Jesse Vig, Wojciech Kryscinski, Shafiq Joty, Caiming Xiong, and Chien-Sheng Wu. 2023. SWiPE: A dataset for document-level simplification of Wikipedia pages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10674–10695, Toronto, Canada. Association for Computational Linguistics.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2024. RLAIF vs. RLHF: scaling reinforcement learning from human feedback with AI feedback. In *The Forty-First International Conference on Machine Learning*, ICML'24. JMLR.org.

Guanlin Li, Yuki Arase, and Noel Crespi. 2025. Aligning sentence simplification with ESL learner's proficiency for language acquisition. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 492–507, Albuquerque, New Mexico. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Preprint*, arXiv:1907.11692.

Yixin Liu, Kejian Shi, Katherine He, Longtian Ye, Alexander Fabbri, Pengfei Liu, Dragomir Radev, and Arman Cohan. 2024. On learning to summarize with large language models as references. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8647–8664, Mexico City, Mexico. Association for Computational Linguistics.

Mounica Maddela and Fernando Alva-Manchego. 2025. Adapting sentence-level automatic metrics for document-level simplification evaluation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6444–6459, Albuquerque, New Mexico. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408.

Louis Martin, Éric de la Clergerie, Benoît Sagot, and Antoine Bordes. 2020. Controllable sentence simplification. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4689–4698.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Abhay Kumara Sri Krishna Nandiraju, Gondy Leroy, David Kauchak, and Arif Ahmed. 2025. Automated feedback loops to protect text simplification with generative AI from information loss. In *Intelligent Systems and Applications*, pages 501–513, Cham. Springer Nature Switzerland.

Tong Niu, Shafiq Joty, Ye Liu, Caiming Xiong, Yingbo Zhou, and Semih Yavuz. 2024. Judgerank: Leveraging large language models for reasoning-intensive reranking. *Preprint*, arXiv:2411.00142.

Sun-Young Oh. 2001. Two types of input modification and efl reading comprehension: Simplification versus elaboration. *TESOL Quarterly*, 35(1):69–96.

OpenAI. 2023. GPT-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *36th Conference on Neural Information Processing Systems*.

Gustavo Paetzold and Lucia Specia. 2016. Understanding the lexical simplification needs of non-native speakers of English. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 717–727, Osaka, Japan. The COLING 2016 Organizing Committee.

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.

Qwen Team. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Vipul Raheja, Dhruv Kumar, Ryan Koo, and Dongyeop Kang. 2023. CoEdIT: Text editing by task-specific instruction tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5274–5291, Singapore. Association for Computational Linguistics.

Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 712–718, Melbourne, Australia. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Patricia Cain Smith and L. M. Kendall. 1963. Retranslation of expectations: An approach to the construction of unambiguous anchors for rating scales. *Journal of Applied Psychology*, 47(2):149–155.

Yishen Song, Qianta Zhu, Huaibo Wang, and Qinhua Zheng. 2024. Automated essay scoring and revising based on open-source large language models. *IEEE Transactions on Learning Technologies*, 17:1880–1890.

Neha Srikanth and Junyi Jessy Li. 2021. Elaborative simplification: Content addition and explanation generation in text simplification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5123–5137, Online. Association for Computational Linguistics.

Renliang Sun, Hanqi Jin, and Xiaojun Wan. 2021. Document-level text simplification: Dataset, criteria and baseline. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7997–8013, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of LM alignment. *Preprint*, arXiv:2310.16944.

Sowmya Vajjala and Ivana Lučić. 2018. OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 297–304, New Orleans, Louisiana. Association for Computational Linguistics.

Laura Vásquez-Rodríguez, Matthew Shardlow, Piotr Przybyła, and Sophia Ananiadou. 2023. Document-level text simplification with coherence evaluation. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 85–101, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024a. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450, Bangkok, Thailand. Association for Computational Linguistics.

Tianduo Wang, Shichen Li, and Wei Lu. 2024b. Self-training with direct preference optimization improves chain-of-thought reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11917–11928, Bangkok, Thailand. Association for Computational Linguistics.

Rodrigo Wilkens, Bruno Oberle, and Amalia Todirascu. 2020. Coreference-based text simplification. In

Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI), pages 93–100, Marseille, France. European Language Resources Association.

Xuanxin Wu and Yuki Arase. 2026. An in-depth evaluation of large language models in sentence simplification with error-based human assessment. *ACM Transactions on Intelligent Systems and Technology*, 17(4).

Haoran Xu, Kenton Murray, Philipp Koehn, Hieu Hoang, Akiko Eriguchi, and Huda Khayrallah. 2025. X-ALMA: Plug & play modules and adaptive rejection for quality translation at scale. In *The Thirteenth International Conference on Learning Representations*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Akifumi Yanagisawa, Stuart Webb, and Takumi Uchihara. 2020. How do different forms of glossing contribute to L2 vocabulary learning from reading?: A meta-regression analysis. *Studies in Second Language Acquisition*, 42(2):411–438.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Yang Zhong, Chao Jiang, Wei Xu, and Junyi Jessy Li. 2020. Discourse level factors for sentence deletion in text simplification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9709–9716.

A Appendix

A.1 Preliminaries: Preference Optimization

Preference Optimization (PO) was first popularized by InstructGPT (Ouyang et al., 2022) using proximal policy optimization (PPO) (Schulman et al., 2017). However, PPO suffers from instability, high variance, and complexity, as it requires a reward model and online reinforcement learning. To address these limitations, Direct Preference Optimization (DPO) (Rafailov et al., 2023) was proposed as a lightweight alternative, removing the need for an explicit reward model.

Building on DPO, Contrastive Preference Optimization (CPO) (Xu et al., 2024) further improves memory efficiency through a simpler preference loss combined with a behavior cloning loss, reducing reliance on a reference model required by DPO. CPO has been shown to be effective for machine translation. In parallel, Simple Preference Optimization (SimPO) (Meng et al., 2024) also offers a reference-free but more stable formulation, incorporating length normalization and a target reward margin. CPO and SimPO can be combined to CPO-SimPO for improved performance and stability.

In this study, we use ARPO, which is a variant of CPO designed to mitigate its tendency to overly penalize dispreferred responses that are only marginally worse than preferred ones. Following CPO-SimPO, we also incorporate SimPO into the ARPO loss. We implemented based on the authors' implementation[11] and paper (Xu et al., 2025). Formally, given a policy model π_θ and a preference dataset $\mathcal{D}$ consisting of source texts x, preferred candidates y_w, and dispreferred candidates y_l, we optimize the objective:

$$\mathcal{L} = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\mathcal{L}_{\text{pref}}(x,y_w,y_l) + \mathcal{L}_{\text{BC}}(x,y_w)\right] \quad (4)$$

$$\mathcal{L}_{\text{pref}}(x,y_w,y_l) = \log\sigma\Big(\frac{\beta}{|y_w|}\log\pi_\theta(y_w \mid x) - \tau_\theta(y_w,y_l)\cdot\frac{\beta}{|y_l|}\log\pi_\theta(y_l \mid x) - \gamma\Big) \quad (5)$$

$$\mathcal{L}_{\text{BC}}(x,y_w) = \alpha\log\pi_\theta(y_w \mid x) \quad (6)$$

where β is a hyperparameter, σ is the sigmoid function. τ_θ is determined by the similarity between y_w and y_l, ranging from 0 to 1. $|y_w|$ and $|y_l|$ represent

[11] https://github.com/fe1ixxu/ALMA/tree/master

the length normalization for the preferred and dispreferred candidates, respectively. γ denotes the target reward margin, and $\mathcal{L}_{\text{BC}}(x, y_w)$ denotes the behavior cloning loss.

A.2 Experiment Setting

We used LoRA (Hu et al., 2022) for both PO and SFT training, with $\alpha = 32$ and $r = 16$. For PO, we set the learning rate to $1\text{e}{-4}$. Total batch size was set to 128 for the sentence level and 4 for the document level. We set $\beta = 0.1$ and $\gamma = 1.5$ for the SimPO loss (Meng et al., 2024), while α is fixed to 1. For SFT, we used the LLaMA-Factory package (Zheng et al., 2024). We set the learning rate to $2e-4$ for sentence-level training with 1 epoch, and to $5e-5$ for document-level training with 3 epochs.

We used the vLLM package (Kwon et al., 2023)[12] for inference with open-source LLMs. We set the decoding parameters to temperature $= 0$, top-$p = 1.0$, and top-$k = -1$. For Qwen3-32B in thinking mode, which was used only for LLM-Judge, we followed the official settings[13] with temperature $= 0.6$ and top-$p = 0.95$. For the document-level SFT models, we also used temperature $= 0.6$ and top-$p = 0.95$ to avoid repetitive outputs. Sentence-level experiments are on a single NVIDIA A6000 Ada 48GB GPU. Document-level experiments are on a single NVIDIA H100 SXM5 94GB GPU.

For OTAlign, we used the supervised setting of the authors' implementation[14], using $\tau = 0.88$ and a threshold of 0.40. SARI was computed with the EASSE package (Alva-Manchego et al., 2019). LENS, LENS-SALSA[15], Agg-LENS, and Agg-BERT were computed with the authors' implementation.[16]

A.3 Implementation of Edit Analysis

In this section, we describe our implementation of the edit-category distribution and edit-pattern analysis. All implementations are based on the edit classifier from Laban et al. (2023).[17]

[12] https://github.com/vllm-project/vllm

[13] https://huggingface.co/Qwen/Qwen3-32B

[14] https://github.com/yukiar/OTAlign

[15] https://github.com/Yao-Dou/LENS

[16] https://github.com/cardiffnlp/document-simplification

[17] https://github.com/salesforce/simplification

A.3.1 Edit-category Distribution

Given a source document and its simplified version, the edit classifier assigns each edit to one of 19 fine-grained categories (see the original paper for details). We group the policy-relevant categories into three broader edit types for analysis: *deletion* (semantic deletion, syntactic deletion, and nonsim noise deletion), *elaboration* (semantic elaboration example, semantic elaboration background, and semantic elaboration generic), and *lexical* (lexical generic and lexical entity). All remaining categories are kept unchanged. The classifier is based on RoBERTa-large (Liu et al., 2019) and is limited to inputs of at most 512 tokens. To process longer documents, we first align source and simplified sentences using gpt-oss-120b (OpenAI, 2025) in a one-shot setting (Figures 13 and 14). We then merge aligned sentence pairs into larger chunks whenever they share a source or simplified sentence, and apply the classifier to each chunk. Source sentences with no aligned simplified counterpart are labeled as *deletion*.

Because the classifier predicts span-level edits, raw edit counts are not directly comparable: for instance, deleting a single word and deleting an entire sentence would each count as one edit. To make the statistics comparable, we convert predictions to the word level by assigning each word in a labeled span or deleted sentence its predicted edit category. We then compute the proportion of each category as the number of words assigned to that category divided by the total number of edited words.

A.3.2 Edit-pattern Analysis

Given a source text and its simplification, the authors' implementation first derives edit operations *insert*, *delete*, and *equal* between the two texts, which indicate how the simplification is performed. These operations are then converted into a marked text sequence, where inserted and deleted spans are wrapped with special boundary tokens. The classifier outputs a category to each inserted or deleted span.

For comparison, we extract spans as follows: for Elaboration, we use the inserted span in the simplified text; for Deletion, the deleted span in the source text; and for Lexical edits, the deleted span in the source text, since the same complex expression may be simplified in multiple valid ways. For each category, we compare spans from the model output with the human reference and identify correspondences.

A.4 Prompts

A.4.1 LLM-Judge

Evaluation guidelines for LLM-Judge are provided in the system message of the prompts. They include the task description, the materials provided to the judge, the evaluation principles, and the required output format. The sentence-level and document-level guidelines are shown in Figures 7 and 11, respectively. The sentence-level prompt also includes three examples (Figures 8, 9, and 10). In each example, the input contains a source sentence along with four simplification candidates, accompanied by their corresponding word alignments and syntactic parses. The output includes evaluation analysis and decisions across lexical, structural, and overall dimensions, documented by the authors. For document-level evaluation, the prompt includes one anchor example for each policy (Table 14). Each anchor example contains a source document, a gold simplified document representing the target policy, and edit information derived from sentence alignment and the edit classifier (see A.3.1). The judge model was instructed to make its judgments referring to the anchor examples while considering the overall quality of document simplification (e.g., coherence and coreference).

A.4.2 Simplification Generation

We use prompts for simplification generation at three stages: candidate collection, fine-tuning, and evaluation. The sentence-level prompts are from a previous study (Wu and Arase, 2026) and illustrated in Figure 5. They are carefully designed and validated on the ASSET and Turk development sets. Each prompt contains an instruction followed by three in-context examples, which we employ for candidate collection. For document-level simplification, we design a new set of prompts, shown in Figure 6, based on edit behaviours observed in human references from the OSE and Newsela corpora. Specifically, we use the edit classifier to estimate the distribution of edit categories in each dataset. In OSE, deletion is the most frequent edit type, accounting for about 45% of edits, followed by lexical edits at roughly 25%. In Newsela, deletion is also the most common edit type, making up around 50% of edits, followed by elaboration accounts for about 15%. Based on these patterns, we design prompts that encourage the model to perform the desired edits while still allowing deletion of less important information. For fine-tuning and evaluation on target LLMs, we only use the system instruction part of the prompts.

A.4.3 Parse

We use a 1-shot prompt to obtain parses, as shown in Figure 12. The model receives a system prompt specifying the task and a single input–output demonstration to illustrate the expected format.

A.5 Ablation Study

A.5.1 Evidence-Guided Judge Design

As described in Section 3.3.2, our sentence-level preference construction adopts an evidence-guided Judge design. This design provides the LLM-Judge with human-designed evaluation criteria and external linguistic evidence, including word alignments and parse trees. Rather than asking the judge to output only best/worst labels, we require it to relate observed lexical and syntactic edits to explicit reward or penalty principles. This makes preference-data construction more transparent and traceable, as users can inspect the judge's rationale before the final decision and verify whether the selected preference pairs are consistent with the intended edit policy.

To examine whether this design contributes to the downstream performance, we conduct an ablation study by removing the detailed reward/penalty criteria, word alignments, and parse information from the judge input. As shown in Table 5, models trained on preference data constructed without the evidence-guided design show performance drops, especially on ASSET-LENS. This suggests that evidence-guided judging provides more effective learning signals for preference optimization, particularly for improving overall simplification quality.

Manual inspection of a small number of samples suggests that evidence-guided design helps the judge place more value on edit-level simplicity changes rather than surface-level fidelity to the source. Table 6 provides an illustrative example. **With the evidence-guided design**, the judge explicitly rewards candidate 0 for simplifying the complex construction "lies in an area known as... characterized by" into "is in... where... interact," and for replacing "producing" with the simpler synonym "causing." It also penalizes candidate 1 for reversing the original subject–location relation and using the less precise verb "meet." As a result, candidate 0 is selected as the best overall candidate and candidate 1 as the worst. **Without the evidence-**

Model	Method	Turk-SARI	ASSET-LENS
Phi3.8B	PO	**43.3**	**71.4**
	w/o evidence	43.3	69.3
Qwen7B	PO	**43.7**	**70.8**
	w/o evidence	**43.7**	68.2
Llama8B	PO	**43.7**	**70.2**
	w/o evidence	43.6	68.3
Qwen14B	PO	**43.9**	**70.3**
	w/o evidence	43.6	67.9

Table 5: Ablation study on the effect of evidence-guided judge design on simplification performance.

guided design, the judge still detects the major error in candidate 1, but its analysis is less aligned with the intended edit policy. In particular, it favors candidate 2 over candidate 0 mainly because "frequently" is closer to the source word "frequent," even though this choice does not make the sentence simpler. This leads to a different choice for the best candidate.

A.5.2 PO vs. SFT on the Preferred Candidates

Our PO models are trained with both preferred and dispreferred candidates through preference optimization. To examine whether the preferred responses alone are sufficient for edit-policy alignment, we conduct an additional ablation, SFT_Chosen, where models are fine-tuned only on the preferred responses from the generated preference pairs. As shown in Table 7, SFT_Chosen achieves strong performance, confirming the quality of the selected preferred candidates. At the sentence level, SFT_Chosen and PO obtain comparable Turk-SARI scores, while PO shows a modest advantage on ASSET-LENS across all models. At the document level, PO outperforms SFT_Chosen under both edit policies. Overall, these results suggest that preference optimization can provide more stable gains than training only on preferred candidates, especially for document-level simplification.

A.5.3 Training Corpus Size vs. Performance

We investigate how the size of the preference optimization dataset affects model performance. For sentence-level simplification, we sample subsets of 1k, 2k, 3k, and 5k preference pairs from the full 7k training set. For document-level simplification, we sample subsets of 1.5k and 2.5k preference pairs from the full 3.5k training set. All other training settings are kept fixed.

Policy alignment improves as the preference optimization dataset grows. Sentence-level results are shown in Figure 3, and document-level results are shown in Figure 4. The overall trend is clear: performance improves as the training set grows. At the sentence level, models show substantial gains once the training size reaches 2k, after which improvements become more gradual. One exception is Qwen7B on Sent-Overall, where performance drops when the number of samples increases from 2k to 3k, but rises again with further increases in training size. At the document level, the proportion of desired edits generally increases with training size. Interestingly, for the Doc-Elaboration policy, the two models show different trends from 2.5k to 3.5k: Llama increases its elaboration rate, whereas Qwen shows a slight decline. Nevertheless, the Parallel baseline still achieves better performance. These results suggest that learning to perform elaborative edits is more challenging and may require professionally rewritten simplifications, such as those in Newsela, to provide more effective supervision during training.

A.6 Generalization to Other Domains

To assess our generalisation capability to other domains, we further evaluated our methods on sentence-level simplification on an out-of-domain of Wikipedia (ASSET and Turk).[18] Specifically, we used **SimPA**, a public-administration dataset which contains $1,100$ original sentences paired with two types of references: (1) lexical simplifications, and (2) overall simplifications combined with lexical and syntactic edits. We evaluate our Sent-Lexical model on the former and the Sent-Overall model on the latter. We applied our policy models directly using the same prompts and hyperparameters as in the wiki-domain experiments, without any retraining. Table 8 shows that our policy models consistently outperform the vanilla baselines. It clearly demonstrates that the learned policies are effective across domains.

A.7 Discussion

A.7.1 Qwen3-32B as Candidate Generator, LLM-Judge, and Parser

Qwen3-32B plays three roles in our pipeline: it is used as one of the simplification candidate generators, as the judge model, and as the parser for

[18] Document-level evaluation was impossible due to the lack of available corpora.

Source: Most of the state lies in an area known as Tornado Alley characterized by frequent interaction between cold and warm air masses producing severe weather.				
ID	**Candidate**	**Aspect**	**w/ evidence-guided judge**	**w/o evidence-guided judge**
0	Most of the state is in Tornado Alley, where cold and warm air masses often interact, causing severe weather.	Lexical	Replaces "lies in an area known as... characterized by" with "is in... where... interact" (active voice simplification) → high reward. "Producing" → "causing" (direct synonym) → high reward. Retains key terms ("Tornado Alley," "air masses," "severe weather") → high reward.	Uses "is in Tornado Alley, where cold and warm air masses often interact, causing severe weather." The word "often" is a good substitute for "frequent." "Causing" is correct. Lexically accurate.
		Structural	Flattens the source's complex participial clause ("characterized by... producing") into a relative clause ("where... interact, causing") → high reward.	Structure is similar, uses "where" clause. Good.
1	Tornado Alley is located in the state where cold and warm air masses frequently meet and cause severe weather.	Lexical	Reverses subject-relationship ("Tornado Alley is located in the state" vs. original focus on "most of the state lies in Tornado Alley") → moderate penalty. Replaces "interaction" with "meet" (less precise) → moderate penalty.	Changes the structure a bit. Says Tornado Alley is located in the state, which might be a slight misrepresentation. Also, "meet" is less precise than "interact." Maybe not as good lexically.
		Structural	Inverts subject-object relationship, creating a less intuitive focus (Tornado Alley as subject) → high penalty. Adds "located in" (redundant phrasing) → moderate penalty.	Changes the subject to Tornado Alley, which might not follow the original structure as closely. Structural change.
2	Most of the state is in Tornado Alley, where cold and warm air masses frequently interact, causing severe weather.	Lexical	Identical to 0 except "often" → "frequently" (synonym with no simplification) → neutral.	Similar to 0 but uses "frequently interact" instead of "often interact." "Frequently" is closer to the original's "frequent." So maybe better than 0.
		Structural	Same structure as 0 → high reward.	Same as 0, maybe better with "frequently interact."
3	Most of the state is in Tornado Alley, an area known for frequent clashes between cold and warm air masses that create severe weather.	Lexical	Replaces "interaction" with "clashes" (slightly more vivid but not simpler) → moderate penalty. "Create" is simpler than "producing" → high reward.	Uses "clashes between cold and warm air masses that create severe weather." "Clashes" is a bit more vivid but might not be as precise as "interaction." "Create" is correct, but "clashes" could be a lexical issue.
		Structural	Adds an appositive phrase ("an area known for...") that increases nesting slightly → moderate penalty.	More complex structure with "an area known for..." which adds a bit more. Maybe more complex than needed.

Table 6: Comparison of lexical and structural judge analyses with and without evidence-guided judge design. The analyses are taken from the model output before the final judgments. Final judgments (best/worst) for w/ evidence-guided judge were Lexical: 0/1, Structural: 0/1, Overall: 0/1, while for w/o evidence-guided judge were Lexical: 2/1, Structural: 0/1, Overall: 2/1.

Model	Method	Turk-SARI	ASSET-LENS
Phi3.8B	SFT_Chosen	43.2	69.8
	PO	**43.3**	**71.4**
Qwen7B	SFT_Chosen	**43.7**	70.1
	PO	**43.7**	**70.8**
Llama8B	SFT_Chosen	**43.9**	70.0
	PO	43.7	**70.2**
Qwen14B	SFT_Chosen	**43.9**	70.2
	PO	**43.9**	**70.3**

(a) Sentence-level

		OSE		Newsela	
Model	Method	LENS	BERT	LENS	BERT
Qwen7B	SFT_Chosen	50.3	42.5	50.5	51.5
	PO	**53.7**	**50.1**	**50.7**	**52.0**
Llama8B	SFT_Chosen	**54.6**	49.6	51.7	51.9
	PO	53.9	**51.5**	**52.6**	**55.6**

(b) Document-level

Table 7: Comparison of preference optimization and SFT on our preference data.

Model	Method	L-SARI	O-LENS
Phi3.8B	Vanilla-zs	22.3	60.1
	PO (Ours)	**36.9**	**63.5**
Qwen7B	Vanilla-zs	26.5	56.0
	PO (Ours)	**36.8**	**63.3**
Llama8B	Vanilla-zs	27.7	59.8
	PO (Ours)	**37.1**	**61.2**
Qwen14B	Vanilla-zs	27.5	58.7
	PO (Ours)	**37.5**	**61.9**

Table 8: Out-of-domain evaluation results on SimPA. We report results for two policy types: Sent-Lexical (L) and Sent-Overall (O).

producing parse-tree information. In this section, we first clarify the motivation for this design and then empirically examine whether its circulation usage introduces systematic bias.

Qwen3-32B is included as a simplification candidate because of its strong simplification performance. We use multiple LLMs with different architectures and parameter scales, including Qwen, Phi, and Llama, to increase the diversity of simplification candidates and reduce dependence on a single model family [19]. In preliminary experiments, we evaluated several larger open-source LLMs that could be run locally within our computational environment, including Gemma3-27B (Gemma Team

[19]The inclusion of outputs from models that are also used for training, namely Qwen2.5-7B and Llama3.1-8B, is motivated by prior work showing that self-generated outputs can provide useful learning signals for preference optimization (Chen et al., 2024; Wang et al., 2024b).

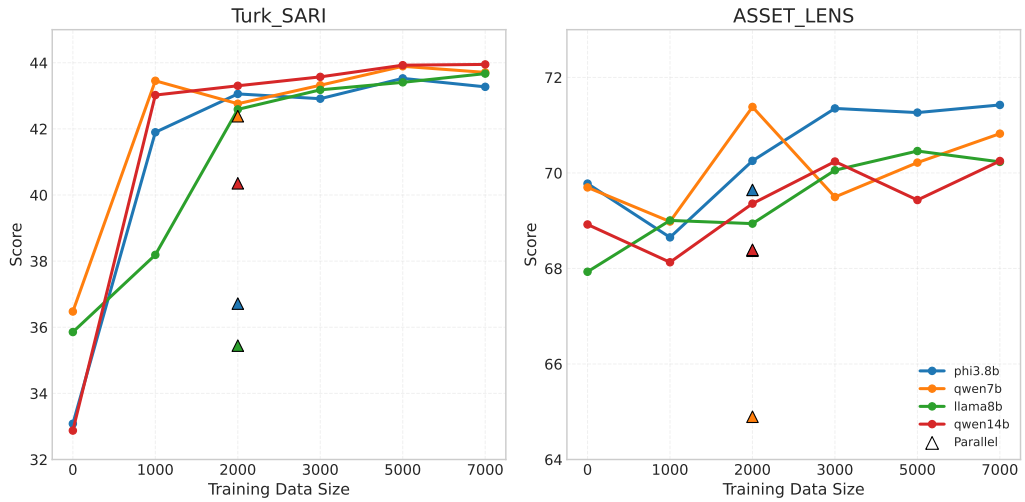


Figure 3: Impact of training sample size in sentence simplification. Colored lines show models trained on our preference data; colored triangles show models trained on $2k$ human-written parallel data. Overlap in triangles is due to the nearly identical LENS scores from Qwen14B and Llama8B.

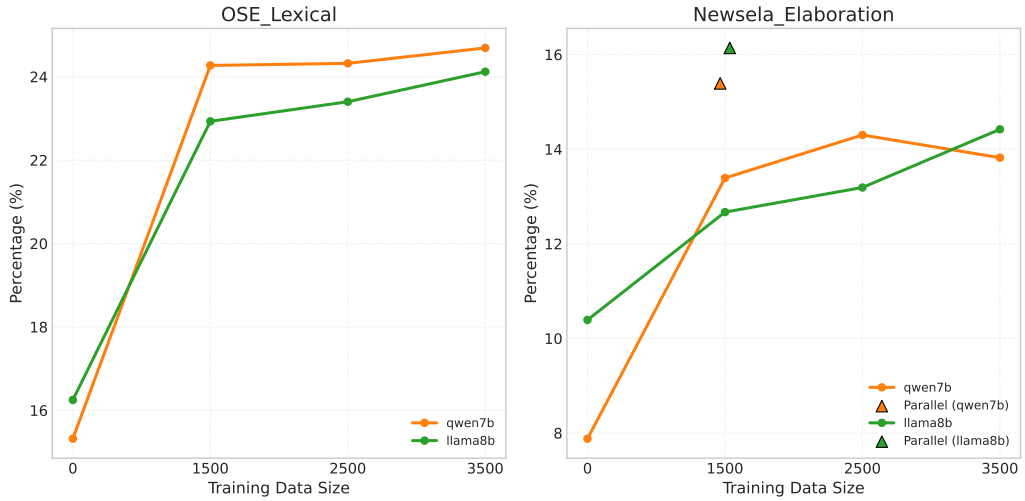


Figure 4: Impact of training sample size in document simplification.

et al., 2025), OSS-20B (OpenAI, 2025), and Aya-Expanse-32B (Dang et al., 2024). Among these models, Qwen3-32B achieved the strongest overall simplification performance. Under the same few-shot prompting setting, Qwen3-32B outperformed the other candidate models on both Turk and ASSET, as shown in Table 9. We therefore include Qwen3-32B as one of the simplification candidate models, expecting it to enrich the candidate pool with higher-quality simplifications.

Qwen3-32B is selected as the judge due to its strong judge ability. The choice of judge model is critical for constructing reliable preference data that aligns with the target edit policies. We compare Qwen3-32B and Gemma3-27B as judge models by using the preference data selected by each model to train downstream models, while keeping all other settings unchanged. As shown in Table 10, models trained on Qwen3-32B-selected preference data consistently outperform those trained on Gemma3-27B-selected data, particularly at the document level. These results suggest that Qwen3-32B provides higher-quality preference annotations for our setting, which in turn improves downstream editing performance.

Qwen-generated parse trees do not bias the judge toward Qwen3-32B outputs. We examine whether using Qwen3-32B to generate parse-tree information biases the judge toward Qwen3-32B outputs. We conduct an ablation study in which only the parse-tree information is removed, while all other components remain unchanged. If

Model	Turk-SARI	ASSET-LENS
Gemma3-27B	39.5	67.9
OSS-20B	41.7	66.3
Aya-Expanse-32B	40.4	67.5
Qwen3-32B	**43.0**	**71.0**

Table 9: Comparison of larger open-source LLMs under the same few-shot prompting setting.

Model	Method	Turk-SARI	ASSET-LENS
Phi3.8B	PO_Gemma	42.9	70.5
	PO_Qwen	**43.3**	**71.4**
Qwen7B	PO_Gemma	43.2	70.2
	PO_Qwen	**43.7**	**70.8**
Llama8B	PO_Gemma	42.9	69.5
	PO_Qwen	**43.7**	**70.2**
Qwen14B	PO_Gemma	43.1	70.0
	PO_Qwen	**43.9**	**70.3**

(a) Sentence-level

		OSE		Newsela	
Model	Method	LENS	BERT	LENS	BERT
Qwen7B	PO_Gemma	50.7	45.6	49.1	49.0
	PO_Qwen	**53.7**	**50.1**	**50.7**	**52.0**
Llama8B	PO_Gemma	51.9	47.7	51.1	52.4
	PO_Qwen	**53.9**	**51.5**	**52.6**	**52.6**

(b) Document-level

Table 10: Comparison between Gemma3-27B and Qwen3-32B as the judge.

Qwen-generated parse trees induced a preference for Qwen3-32B outputs, we would expect the preference fraction of Qwen3-32B to increase when parse-tree information is included. However, we observe the opposite pattern as shown in Table 11. Adding parse-tree information slightly decreases the preferred fraction of Qwen3-32B in both Sent-Lexical and Sent-Overall settings: from 32.2% to 31.9% and from 29.6% to 28.8%, respectively. This indicates that the use of Qwen-generated parse trees does not lead the judge to favor Qwen3-32B outputs.

Qwen3-32B as the judge does not systematically favor its own simplification outputs. We examine whether using Qwen3-32B as both the judge and one of the candidate generators introduces model-selection bias. If the judge systematically favored its own generations, we would expect outputs generated by Qwen3-32B to dominate the selected candidates across edit policies. To test this, we analyze the model-wise distribution of preferred candidates selected by Qwen3-32B. As shown in Table 11, the preferred fraction of Qwen3-32B outputs ranges from 28.8% to 44.2% across policies, and Qwen3-32B is not consistently the most frequently selected candidate model. For example, Phi4-14B is selected more often than Qwen3-32B under both sentence-level policies and also receives high preferred fractions in the document-level settings. These results suggest that Qwen3-32B, when used as the judge, does not simply select its own outputs by default.

A.7.2 Model Outputs vs. Human

To better understand the discrepancies between edit patterns in PO model outputs and human references discussed in Section 4.5, we manually examined a small set of examples. The case analysis shows that human references and PO model outputs often differ in both what they simplify and how they simplify it. For example, human references sometimes simplify domain-specific terms, while PO models tend to preserve them. PO models also often paraphrase the surrounding context more broadly than human references. An illustrative example from the Doc-Lexical policy is shown below.

Source: ...the 1961 Single Convention on Narcotic Drugs, the framework that governs international drugs policy...

Human: ...the 1961 Convention on Drugs, the agreement that controls international drugs policy...

Qwen7B: ...the 1961 Single Convention on Narcotic Drugs, which sets rules for international drug policy...

Llama8B: ...the 1961 Single Convention on Narcotic Drugs, which is the main rule for international drug policy...

The human reference simplifies the domain-specific term *1961 Single Convention on Narcotic Drugs* to *1961 Convention on Drugs*, while both PO models preserve the full term. The human reference also makes localized lexical substitutions, such as *framework* → *agreement* and *governs* → *controls*. In contrast, the PO models paraphrase the surrounding phrase more broadly, using expressions such as *sets rules for* and *is the main rule for*. These differences can reduce measured span overlap, even when the model outputs constitute valid simplifications.

Model	Sentence-level				Document-level	
	w/o Parse		w/ Parse		Doc-Lexical	Doc-Elaboration
	Sent-Lexical	Sent-Overall	Sent-Lexical	Sent-Overall		
Qwen2.5-7B	24.0/22.3	25.0/22.9	22.9/22.3	25.1/22.2	19.5/17.3	21.1/15.5
Llama3.1-8B	12.1/50.5	14.3/48.7	13.2/49.9	13.7/48.8	5.9/63.1	6.4/68.3
Phi4-14B	31.6/12.1	31.2/11.3	32.0/12.3	32.5/11.4	30.3/12.3	30.5/10.2
Qwen3-32B	32.2/15.0	29.6/17.2	31.9/15.5	28.8/17.6	44.2/7.2	42.0/6.0

Table 11: Preference distribution under the Qwen3-32B judge. Each value is reported as preferred/dispreferred ratio. For sentence-level criteria, we also report ablation results without parse-tree information.

```
System:
You are required to simplify the original sentence by using simpler concepts, words, or
phrases. Pleasekeep the meaning the same. Only provide one result.

User:
San Francisco Bay is located in the U.S. state of California, surrounded by a
contiguous region known as the San Francisco Bay Area, dominated by the large cities
San Francisco, Oakland and San Jose.

Assistant:
San Francisco Bay is located in the U.S. state of California, surrounded by a
contiguous region known as the San Francisco Bay Area, influenced by the large cities,
San Francisco, Oakland and San Jose.

User:
The book chronicles events which take place in the fictional space colony of Windhaven.

Assistant:
The book chronicles events which take place in the space colony of Windhaven.

User:
Some academic journals do refer to Wikipedia articles, but are not elevating it to the
same level as traditional references.

Assistant:
Some academic journals do refer to Wikipedia articles, but are not using it to the same
level as common references.

User:
{Input}
```

(a) Sent-Lexical

```
System:
You are required to simplify the original sentence by applying different
transformations. Please keep the meaning the same. Only provide one result.

User:
Rollins retired in 1962 and opted to become a coach.

Assistant:
Rollins retired in 1962. He then chose to become a coach.

User:
Tourism is concentrated in the mountains, particularly around the towns of Davos /
Arosa, Laax and St. Moritz / Pontresina.

Assistant:
Tourism takes place in the mountains around the towns of Davos / Arosa, Laax and St.
Moritz / Pontresina.

User:
First Fleet is the name given to the 11 ships which sailed from Great Britain on 13 May
1787 with about 1,487 people to establish the first European colony in New South Wales.

Assistant:
11 ships sailed from Great Britain on 13 May 1787 carrying about 1,487 people. These
ships aimed to establish the first European colony in New South Wales. These 11 ships
were named First Fleet.

User:
{Input}
```

(b) Sent-Overall

Figure 5: Prompts used for sentence-level simplification generation (from Wu and Arase (2026)).

Model	Sentence
Source 1	One side of the armed conflicts is composed mainly of the Sudanese military and the Janjaweed, a Sudanese militia group recruited mostly from the Afro-Arab Abbala tribes of the northern Rizeigat region in Sudan.
Phi3B-Lexical	One side of the armed conflicts is mainly made up of the Sudanese military and the Janjaweed, a Sudanese militia group mostly recruited from the Afro-Arab Abbala tribes in the northern Rizeigat region of Sudan.
Phi3B-Overall	The armed conflicts involve the Sudanese military and the Janjaweed, a Sudanese militia group mostly made up of the Afro-Arab Abbala tribes from the northern Rizeigat region in Sudan.
Qwen7B-Lexical	One side of the armed conflicts is mainly made up of the Sudanese military and the Janjaweed, a Sudanese militia group mostly from the Afro-Arab Abbala tribes in the northern Rizeigat region of Sudan.
Qwen7B-Overall	One side of the armed conflicts includes the Sudanese military and the Janjaweed, a militia group from the northern Rizeigat region.
Llama8B-Lexical	One side of the conflicts is made up mainly of the Sudanese military and the Janjaweed, a militia group from the Afro-Arab Abbala tribes in northern Sudan.
Llama8B-Overall	One side of the armed conflicts includes the Sudanese military and the Janjaweed, a militia group from the northern Rizeigat region, mostly from the Afro-Arab Abbala tribes.
Qwen14B-Lexical	One side of the conflict is made up mainly of the Sudanese military and the Janjaweed, a Sudanese militia group mostly recruited from the Afro-Arab Abbala tribes in the northern Rizeigat region of Sudan.
Qwen14B-Overall	One side of the armed conflicts includes the Sudanese military and the Janjaweed, a Sudanese militia group from the northern Rizeigat region.
Source 2	The spacecraft consists of two main elements: the NASA Cassini orbiter, named after the Italian-French astronomer Giovanni Domenico Cassini, and the ESA Huygens probe, named after the Dutch astronomer, mathematician and physicist Christiaan Huygens.
Phi3B-Lexical	The spacecraft has two main parts: the NASA Cassini orbiter, named after Italian-French astronomer Giovanni Domenico Cassini, and the ESA Huygens probe, named after Dutch astronomer, mathematician, and physicist Christiaan Huygens.
Phi3B-Overall	The spacecraft has two main parts: the NASA Cassini orbiter and the ESA Huygens probe.
Qwen7B-Lexical	The spacecraft has two main parts: the NASA Cassini orbiter, named after the Italian-French astronomer Giovanni Domenico Cassini, and the ESA Huygens probe, named after the Dutch astronomer, mathematician, and physicist Christiaan Huygens.
Qwen7B-Overall	The spacecraft has two main parts: the NASA Cassini orbiter, named after Giovanni Domenico Cassini, and the ESA Huygens probe, named after Christiaan Huygens.
Llama8B-Lexical	The spacecraft has two main parts: the NASA Cassini orbiter, named after Italian-French astronomer Giovanni Domenico Cassini, and the ESA Huygens probe, named after Dutch astronomer, mathematician, and physicist Christiaan Huygens.
Llama8B-Overall	The spacecraft has two main parts: the NASA Cassini orbiter, named after Giovanni Domenico Cassini, and the ESA Huygens probe, named after Christiaan Huygens.
Qwen14B-Lexical	The spacecraft has two main parts: the NASA Cassini orbiter, named after the Italian-French astronomer Giovanni Domenico Cassini, and the ESA Huygens probe, named after the Dutch astronomer, mathematician, and physicist Christiaan Huygens.
Qwen14B-Overall	The spacecraft has two main parts: the NASA Cassini orbiter, named after Giovanni Domenico Cassini, and the ESA Huygens probe, named after Christiaan Huygens.
Source 3	Each entry has a datum (a nugget of data) which is a copy of the datum in some backing store.
Phi3.8B-Lexical	Each entry has a piece of data (a nugget of data) that is a copy of the data in some storage.
Phi3.8B-Overall	Each entry has a piece of data that is a copy from a storage.
Qwen7B-Lexical	Each entry has a datum, which is a copy of the data from some storage.
Qwen7B-Overall	Each entry has a copy of the data from a backing store.
Llama8B-Lexical	Each entry has a copy of the data from the backing store.
Llama8B-Overall	Each entry has a copy of the data from a backing store.
Qwen14B-Lexical	Each entry has a piece of data that is a copy of the data in a backup storage.
Qwen14B-Overall	Each entry is a copy of a datum from a backing store.

Table 12: Examples of Sent-Lexical and Sent-Overall simplifications by our models.

Model	Document
Source 1	Dame Sally Davies, the Chief Medical Officer for England, has likened the problem of antibiotic resistance to the risks presented by international terrorism. While this might sound like an exaggeration, the threat was actually, if anything, understated. Each year, the global number of deaths to which bacterial resistance contributes far outstrips those caused by terrorist attacks. While it is difficult to track the global impact of antibiotic resistance across all bacterial species, the World Health Organization estimates that for tuberculosis alone multi-drug resistance accounts for more than 150,000 deaths each year. Antibiotic resistance is no longer an abstract risk: this is now a war. In the past hundred years, our expectations of life and survival have changed beyond all recognition...
Qwen7B-Lexical	Dame Sally Davies, the Chief Medical Officer for England, has compared the threat of antibiotic resistance to international terrorism. This comparison is not an exaggeration; the problem is even more serious than it might seem. Each year, the number of deaths caused by antibiotic-resistant bacteria is much higher than those caused by terrorist attacks. While it's hard to track the global impact of antibiotic resistance, the World Health Organization estimates that multi-drug resistant tuberculosis alone causes more than 150,000 deaths annually. Antibiotic resistance is no longer just a potential risk; it's a real and urgent threat. In the past century, our expectations of life and survival have changed dramatically...
Llama8B-Lexical	Dame Sally Davies, the Chief Medical Officer for England, has compared the problem of antibiotic resistance to the risks of international terrorism. While this might seem like an exaggeration, the threat is actually more serious. Each year, the number of deaths caused by bacterial resistance is much higher than those caused by terrorist attacks. The World Health Organization estimates that for tuberculosis alone, multi-drug resistance causes more than 150,000 deaths each year. Antibiotic resistance is no longer just a risk; it's a real war. In the past hundred years, our expectations of life and survival have changed dramatically...
Source 2	Standing at the edge of space above the deserts of New Mexico, Felix Baumgartner paused slightly. It was a small step away from the capsule, but a 24-mile drop back down to Earth. Our guardian angel will take care of you, said mission control, and the man known as Fearless Felix jumped. Ten heart-stopping minutes later the Austrian landed back on Earth, after reaching speeds of up to 725mph, and breaking three world records, including becoming the worlds first supersonic skydiver by breaking the sound barrier at Mach 1.24. We love you Felix, cheered the control room as his mother, Ava Baumgartner, wept. Baumgartner, who claimed the records for the highest altitude manned balloon flight and the highest altitude skydive, raised his arms in a victory salute to thank his team...
Qwen7B-Lexical	Felix Baumgartner stood at the edge of space above the deserts of New Mexico. He was about to jump from a capsule 24 miles above the Earth. Mission control said, "Our guardian angel will take care of you." Baumgartner jumped. Ten minutes later, he landed safely on Earth, having reached speeds of up to 725 mph and breaking three world records, including becoming the first supersonic skydiver by breaking the sound barrier at Mach 1.24. The control room cheered, and Baumgartner's mother, Ava Baumgartner, cried. He raised his arms in a victory salute to thank his team...
Llama8B-Lexical	Felix Baumgartner, known as Fearless Felix, stood at the edge of space above the deserts of New Mexico. He was about to jump from a capsule 24 miles above the Earth. Mission control told him, "Our guardian angel will take care of you." Felix jumped, and 10 minutes later, he landed back on Earth, reaching speeds of up to 725mph. He broke three world records, including becoming the first person to break the sound barrier in free-fall. The control room cheered, and his mother, Ava Baumgartner, cried. Felix raised his arms in a victory salute to thank his team...
Source 3	A major international row with wide-ranging 1 implications for global drugs policy has erupted over the right of Bolivias indigenous Indian tribes to chew coca leaves, the principal ingredient in cocaine. Bolivia has obtained a special exemption from the 1961 Single Convention on Narcotic Drugs, the framework that governs international drugs policy, allowing its indigenous people to chew the leaves. Bolivia had argued that the convention was 2 in opposition to its new constitution, adopted in 2009, which obliges it to protect native and ancestral coca as cultural patrimony and maintains that coca in its natural state is not a narcotic. South American Indians have chewed coca 3 leaves for centuries...
Qwen7B-Lexical	A big international argument has started over the right of Bolivia's indigenous people to chew coca leaves, which are used to make cocaine. Bolivia has asked to be allowed to keep this tradition because it is part of their culture. The 1961 Single Convention on Narcotic Drugs, which sets rules for international drug policy, usually doesn't allow this. But Bolivia got a special permission to let its indigenous people chew the leaves. This is because their new constitution, from 2009, says they should protect coca as part of their culture and says coca in its natural form is not a drug.South American Indians have chewed coca leaves for centuries...
Llama8B-Lexical	A big international argument has started about the right of Bolivia's indigenous people to chew coca leaves, which are the main ingredient in cocaine. Bolivia has asked for a special exception from the 1961 Single Convention on Narcotic Drugs, which is the main rule for international drug policy. This rule says that coca leaves are illegal, but Bolivia says that in its natural state, they are not a drug and are important to its culture. The country's new constitution, adopted in 2009, says that coca is a part of its cultural heritage and should be protected. South American Indians have been chewing coca leaves for centuries, and they are...

Table 13: Examples of Doc-Lexical simplifications by our models. Documents are truncated for readability.

		Content
Source		Unveiling a car with a top speed of 25mph, two seats, and no pedals or steering wheel might not make much of an impression at a motor show. But Google, in the US, sent a minor earthquake through the car and taxi industries as it unveiled the latest version of its driverless car. The electrically powered vehicle, which Google has begun testing around its headquarters in Mountain View, California, dispenses with all the normal controls, including foot pedals. Instead, it has a smartphone app that summons it and tells it the destination, and a single STOP button mounted between the two front-facing seats in case the occupants need to override the computer. *[truncated]*
Simplified		A car with a maximum speed of 25 miles per hour, two seats, and no pedals or steering wheel does not sound very interesting. But Google, in the US, shocked the car and taxi industries when it unveiled the latest version of its driverless car. Google has begun testing the electric car at its headquarters in Mountain View, California. The car does not have the normal controls, such as foot pedals. Instead, it has a smartphone app that calls it and tells it the destination, and a STOP button between the two seats in case the passengers need to override the computer. *[truncated]*
Edit Category	**Lexical Generic**	*Location 1* "source": "Unveiling a car with a top … at a motor show." "simplified": "A car with a maximum … very interesting." "edit": "A" ← "Unveiling a"; "maximum" ← "top"; … *[truncated]* … *[truncated]* *Location 21* "source": "The company points out that … due to human error." "simplified": "The company says that … by human mistakes." "edit": "says" ← "points out"; "mistakes" ← "error"
	Nonsim Format	*Location 1* "source": "Unveiling a car with … at a motor show." "simplified": "A car with a maximum … very interesting." "edit": "25 miles per hour" ← "25mph"; … *[truncated]*
	Discourse Reordering	*Location 1* "source": "The electrically powered vehicle, which Google has begun testing around its headquarters in Mountain View, California, dispenses with all the normal controls, including foot pedals." "simplified": "Google has begun testing the electric car at its headquarters in Mountain View, California." "edit": "The electrically powered vehicle, which"; … *[truncated]*
	… *[truncated]*	...

Table 14: Anchor example for Doc-Lexical policy used in the prompt for LLM-Judge. This example is a held-out sample from the OSE dataset. For readability, we truncate each field because the documents and edit annotations are very long.

```
System:
You are required to simplify the original document.
Simplify the document by:
    - Mainly focus on lexical simplification: replace complex words/phrases with simpler equivalents.
    - Remove only minor redundancies if needed.
    - Preserve the content as much as possible.
    - Allow discourser reordering when it improves clarity.
    - Output a single simplified document.

User:
Unveiling a car with a top speed of 25mph, two seats and no pedals or steering wheel might not make much of an
impression at a motor show. But Google, in the US, sent a minor earthquake through the car and taxi industries
as it unveiled the latest version of its driverless car. The electrically powered vehicle, which Google has
begun testing around its headquarters in Mountain View, California, dispenses with all the normal controls,
including foot pedals. Instead ...

Assistant:
A car with a maximum speed of 25 miles per hour, two seats and no pedals or steering wheel does not sound very
interesting. But Google, in the US, shocked the car and taxi industries when it unveiled the latest version of
its driverless car. Google has begun testing the electric car at its headquarters in Mountain View, California.
The car does not have all the normal car controls, such as foot pedals. Instead ...

User:
Not just the identity of the man in the car park with the twisted spine, but the appalling last moments and
humiliating treatment of the naked body of Richard III in the hours after his death have been revealed at an
extraordinary press conference at Leicester University. There were cheers when Richard Buckley, Lead
Archaeologist on the hunt for the kings body ...

Assistant:
A team at Leicester University has told the world that the body they found under a local car park is the body
of King Richard III. There were cheers when Richard Buckley, leader of the team of archaeologists, finally said
that they were certain they had found the body of the king ...

User:
Clay Cockrell is sitting in his of ce at Columbus Circle, across the street from 1 Central Park West, which
houses Trump International Hotel and Tower. In front of the tower is Central Park, where Cockrell holds his
popular walk and talk therapy sessions. Cockrell, a former Wall Street worker turned therapist, spends large
parts of his days walking through Central Park or the Battery Park in downtown Manhattan near Wall Street, as a
con dant and counsellor to some of New Yorks wealthiest people ...

Assistant:
Clay Cockrell is sitting in his of ce opposite the Trump International Hotel and Tower. In front of the tower
is Central Park, where Cockrell holds his popular walk and talk therapy sessions. Cockrell is a former Wall
Street worker who is now a therapist. He spends large parts of his days walking in Central Park or the Battery
Park in downtown Manhattan near Wall Street, talking to some of New Yorks wealthiest people ...

User:
{Input}
```

(a) Doc-Lexical

```
System:
You are required to simplify the original document.
Simplify the document by:
    - Delete less important details to enhance readability.
    - Add brief clarifications when helpful (light elaboration).
    - Keep the main content.
    - Replace complex words/phrases with simpler equivalents if needed.
    - Allow discourser reordering when it improves clarity.
    - Output a single simplified document.


User:
{Example Input 1}

Assistant:
{Example Output 1}

User:
{Example Input 2}

Assistant:
{Example Output 2}

User:
{Example Input 3}

Assistant:
{Example Output 3}

User:
{Input}
```

(b) Doc-Elaboration

Figure 6: Prompts used for document-level simplification generation. The instructions are designed to reflect the reference editing behavior observed in the OSE and Newsela corpora.

You will be provided with the following:
    A source sentence.
    Four simplified versions of the source sentence (0, 1, 2, and 3)
    Word alignments between the source and each simplified sentences, structured as:
        words in the source sentence with indices.
        words in the simplified sentence with indices.
        word alignments following the format: sourceIndex_sourceWord-simplifiedIndex_simplifiedWord
    Penn Treebank-style parses for the source and each simplified sentence.

Your task: Act as an evaluation system, choose the best and the worst among four simplified sentences. Analyze each simplified sentence across three aspects:
    Lexical: Refer to word alignments.
    Structural: Refer to the parse trees.
    Overall: Consider both lexical and structural aspects.
    Follow the evaluation principles strictly.

Lexical Evaluation Principles:
    Replacing difficult words with easier ones without changing the meaning, or only slightly changing it. → High reward
    Replacing easy words with even simpler ones without changing the meaning, or only slightly changing it. → Moderate reward
    Replacing words with more complex ones → High penalty
    Replacing words in a way that significantly changes the original meaning → High penalty
    Deleting unimportant difficult words → Moderate reward
    Deleting easy words → Moderate penalty
    Deleting important information → High penalty
    Deleting, replacing, or omitting important named entities → High penalty
    Retaining difficult words → Moderate penalty
    Adding new complex words → High penalty

Structural Evaluation Principles:
    Simplifying difficult structures → High reward
    Simplifying easy structures → Moderate reward
    Using more difficult structures → High penalty
    Splitting long sentences → High reward
    Reordering for clarity → High reward
    Edits that do not contribute to simplicity or clarity → Moderate penalty
    Retaining difficult structures → High penalty

Output Instructions:
    For each aspect (Lexical, Structural, Overall), return:
    Aspect: {Lexical / Structural / Overall}, Best: {0, 1, 2 or 3}, Worst: {0, 1, 2 or 3}
    Strictly follow the above format and do not include any extra symbols.
    For the lexical and structural aspect, also provide the analysis of each simplified sentence during your evaluation.

Figure 7: Sentence-level evaluation guidelines provided to the judge LLM

```
Source sentence: We'll slip you through with our cloaking device.
0: With our cloaking devices, we can slip by you.
1: We will hide you using our cloaking device.
2: We can let you go with our cloaking device.
3: We'll cloak you with our device.

0 alignment:
    0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
    0_with 1_our 2_cloaking 3_devices, 4_we 5_can 6_slip 7_by 8_you.
    0_we'll-4_we 1_slip-6_slip 3_through-7_by 4_with-0_with 5_our-1_our 6_cloaking-
2_cloaking 7_device.-8_you.
1 alignment:
    0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
    0_we 1_will 2_hide 3_you 4_using 5_our 6_cloaking 7_device.
    0_we'll-0_we 0_we'll-1_will 1_slip-2_hide 2_you-3_you 4_with-4_using 5_our-5_our
6_cloaking-6_cloaking 7_device.-7_device.
2 alignment:
    0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
    0_we 1_can 2_let 3_you 4_go 5_with 6_our 7_cloaking 8_device.
    0_we'll-0_we 1_slip-2_let 2_you-3_you 3_through-4_go 4_with-5_with 5_our-6_our
6_cloaking-7_cloaking 7_device.-8_device.
3 alignment:
    0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
    0_we'll 1_cloak 2_you 3_with 4_our 5_device.
    0_we'll-0_we'll 1_slip-1_cloak 2_you-2_you 4_with-3_with 5_our-4_our 7_device.-
5_device.

Source parse:
    (S \n  (NP (PRP We))\n (VP\n (MD 'll)\n (VP\n (VB slip)\n (NP (PRP you))\n (PP\n
(IN through)\n (PP\n (IN with)\n (NP\n (NP (PRP$ our) (NN cloaking) (NN device))))))\n
(. .))
0 parse:
    (S \n  (PP (IN With) \n (NP (PRP$ our) (NNS cloaking) (NNS devices)))\n (, ,)\n (NP
(PRP we))\n (VP (MD can) \n (VP (VB slip) \n (PRT (RP by)) \n (NP (PRP you))))\n (. .))
1 parse:
    (S \n  (NP (PRP We)) \n (VP \n (MD will) \n (VP \n (VB hide) \n (NP (PRP you)) \n
(PP \n (VBG using) \n (NP \n (PRP$ our) \n (NN cloaking) \n (NN device)))))\n (. .))
2 parse:
    (S \n  (NP (PRP We)) (VP (MD can) (VP (VB let) (NP (PRP you)) (VP (VB go) (PP (IN
with) (NP (PRP$ our) (NN cloaking) (NN device))))))) (. .))
3 parse:
    (S (NP (PRP We)) (VP (MD 'll) (VP (VB cloak) (NP (PRP you)) (PP (IN with) (NP (PRP$
our) (NN device))))) (. .))
```

(a) Input

```
Aspect: Lexical, Best: 1, Worst: 0
Analysis:
0: 'slip you through'→'slip by you' reverses the core meaning — high penalty. Also
pluralizes "device" to "devices" without justification — light penalty for minor
factual change. The difficult word 'cloaking' is retained — moderate penalty.
1: 'slip you through'→'hide you' shifts nuance, but becomes simpler and preserves most
of the intended meaning — moderate high reward. The difficult word 'cloaking' is
retained — moderate penalty.
2: 'slip you through'→'let you go' may imply release rather than passage — light
penalty. The difficult word 'cloaking' is retained — moderate penalty.
3: "slip you through"→"cloak you" simplifies verb but loses the specific "cloaking
device" label ("our device" is ambiguous) → moderate-high penalty for deleting an
important modifier

Aspect: Structural, Best: 3, Worst: 2
Analysis:
0: Fronting the PP ("With our…devices") aids clarity. However, it fails to simplify the
source's nested PPs; instead, creates a new multi-layered VP (slip → by → you) without
resolving recursion.
1: Collapses two nested PPs ("through with...") with a single gerundive phrase ("using
our cloaking device"), resulting in a shallower syntactic tree.
2: Avoids the nested PP by using dual VP embedding ("let you go"), but less streamlined
than 1: The PP ("with...") modifies the lower verb (go), maintaining a moderately
hierarchical structure.
3: Simple S → NP VP PP, no nesting → high reward

Aspect: Overall, Best: 1, Worst: 0
```

(b) Corresponding output showing evaluations on lexical, structural, and overall dimensions.

Figure 8: Example 1 for sentence-level LLM-Judge

Source sentence: The humans had grown complacent, having faced no concentrated robotic aggression during the last century of uneasy cold war.
0: People were beginning to be satisfied because during the last hundred years of a nervous cold war, they had not had to face any concentrated aggression of robots.
1: The humans became complacent due to a century without robotic aggression.
2: They had become complacent.
3: The humans became complacent, having faced no major robotic attacks in the last century of cold war tensions.

0 alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_people 1_were 2_beginning 3_to 4_be 5_satisfied 6_because 7_during 8_the 9_last 10_hundred 11_years 12_of 13_a 14_nervous 15_cold 16_war, 17_they 18_had 19_not 20_had 21_to 22_face 23_any 24_concentrated 25_aggression 26_of 27_robotes.
1_humans-0_people 2_had-1_were 3_grown-4_be 4_complacent,-5_satisfied 5_having-18_had 6_faced-22_face 7_no-23_any 8_concentrated-24_concentrated 10_aggression-25_aggression 11_during-7_during 12_the-8_the 13_last-9_last 14_century-11_years 15_of-12_of 16_uneasy-14_nervous 17_cold-15_cold 18_war.-27_robotes.
1 alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_the 1_humans 2_became 3_complacent 4_due 5_to 6_a 7_century 8_without 9_robotic 10_aggression.
0_the-0_the 1_humans-1_humans 3_grown-2_became 4_complacent,-3_complacent 7_no-8_without 9_robotic-9_robotic 10_aggression-10_aggression. 12_the-6_a 14_century-7_century 18_war.-10_aggression.
2 alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_they 1_had 2_become 3_complacent.
0_the-0_they 1_humans-0_they 2_had-1_had 3_grown-2_become 4_complacent,-3_complacent. 5_having-1_had 18_war.-3_complacent.
3 alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_the 1_humans 2_became 3_complacent, 4_having 5_faced 6_no 7_major 8_robotic 9_attacks 10_in 11_the 12_last 13_century 14_of 15_cold 16_war 17_tensions.
0_the-0_the 1_humans-1_humans 3_grown-2_became 4_complacent,-3_complacent, 5_having-4_having 6_faced-5_faced 7_no-6_no 8_concentrated-7_major 9_robotic-8_robotic 10_aggression-9_attacks 11_during-10_in 12_the-11_the 13_last-12_last 14_century-13_century 15_of-14_of 17_cold-15_cold 18_war.-17_tensions.

Source parse:
(S (NP (DT The) (NNS humans)) (VP (VBD had) (VP (VBN grown) (ADJP (JJ complacent), (S (VP (VBG having) (VP (VBN faced) (NP (DT no) (JJ concentrated) (JJ robotic) (NN aggression)) (PP (IN during) (NP (DT the) (JJ last) (NN century) (PP (IN of) (NP (JJ uneasy) (JJ cold) (NN war))))))))))) (. .))
0 parse:
(S (NP (NNS People)) (VP (VBD were) (VP (VBG beginning) (S (VP (TO to) (VP (VB be) (VP (VBN satisfied) (SBAR (IN because) (S (PP (IN during) (NP (DT the) (JJ last) (CD hundred) (NNS years) (PP (IN of) (NP (DT a) (JJ nervous) (NNP Cold) (NNP War))))) (, ,) (NP (PRP they)) (VP (VBD had) (RB not) (VP (VBN had) (S (VP (TO to) (VP (VB face) (NP (DT any) (JJ concentrated) (NN aggression) (PP (IN of) (NP (NNS robots))))))))))))))) (. .))
1 parse:
(S \n (NP (DT The) (NNS humans)) (VP (VBD became) (ADJP (JJ complacent) (PP (IN due) (PP (TO to) (NP (DT a) (NN century) (PP (IN without) (NP (JJ robotic) (NN aggression)))))))) (. .))
2 parse:
(S \n (NP (PRP They)) (VP (VBD had) (VP (VBN become) (ADJP (JJ complacent)))) (. .)
3 parse:
(S (NP (DT The) (NNS humans)) (VP (VBD became) (ADJP (JJ complacent)) (, ,) (S (VP (VBG having) (VP (VBN faced) (NP (DT no) (JJ major) (JJ robotic) (NNS attacks)) (PP (IN in) (NP (DT the) (JJ last) (NN century) (PP (IN of) (NP (JJ cold) (NN war) (NNS tensions)))))))) (. .))

(a) Input

Aspect: Lexical, Best: 3, Worst: 2
Analysis:
0: Simplifies "humans" → "people", "century" → "hundred years", "had grown complacent" → "were beginning to be satisfied". High meaning preservation. "robotic aggression"→"aggression of robots", moderate penalty. Adds filler "because," "they had not had to": moderate penalty.
1: Simplifies "had grown" → "became" (reduces verb complexity) → high reward. Omits "concentrated" and "cold war" nuance → moderate penalty. Retains "complacent" (difficult word) → moderate penalty. Good lexical clarity with minor losses.
2: "humans" → "they" (loss of specificity) → high penalty. Retains "complacent" (difficult word) → moderate penalty. Heavy meaning loss despite surface simplicity.
3: "had grown complacent"→"became complacent": moderate reward. Retains "complacent" (difficult word): moderate penalty. "concentrated"→"major," "aggression"→"attacks": simpler synonyms—high reward.

Aspect: Structural, Best: 2, Worst: 0
Analysis:
0: Retains complex multi-clause structure: nested VP chains, SBAR ("because"), deeply embedded PPs. Adds a "because" link (small clarity gain), but doesn't fundamentally flatten structure.
1: Flattens source's multi-clause structure into a single VP-ADJP chain. Simplifies "during...century" → "due to a century without" (reduces PP embedding → high reward).
2: Collapses the source into minimal NP-VP-ADJP structure (no clauses or modifiers → high reward).
3: Retains the original complex participial construction ("having faced…"), only substituting simpler words, moderate structural penalty for not simplifying the structure.

Aspect: Overall, Best: 1, Worst: 2

(b) Corresponding output showing evaluations on lexical, structural, and overall dimensions.

Figure 9: Example 2 for sentence-level LLM-Judge

Source sentence: These concerns have intensified due to the actions of China, the predominant supplier.
0: These concerns have gotten larger because of the actions of China.
1: China's actions have intensified these concerns.
2: These are concerns that have been raised by China, who are a major supplier.
3: These concerns have intensified because of China's actions, which is the main supplier.

0 alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_these 1_concerns 2_have 3_gotten 4_larger 5_because 6_of 7_the 8_actions 9_of 10_china.
0_these-0_these 1_concerns-1_concerns 2_have-2_have 3_intensified-3_gotten 3_intensified-4_larger 4_due-5_because 5_to-6_of 6_the-7_the 7_actions-8_actions 8_of-9_of 12_supplier.-10_china.
1 alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_china's 1_actions 2_have 3_intensified 4_these 5_concerns.
0_these-4_these 1_concerns-5_concerns. 2_have-2_have 3_intensified-3_intensified 7_actions-1_actions 9_china,-0_china's 12_supplier.-5_concerns.
2 alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_these 1_are 2_concerns 3_that 4_have 5_been 6_raised 7_by 8_china, 9_who 10_are 11_a 12_major 13_supplier.
0_these-0_these 1_concerns-2_concerns 2_have-4_have 3_intensified-6_raised 8_of-7_by 9_china,-8_china, 10_the-11_a 11_predominant-12_major 12_supplier.-13_supplier.
3 alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_these 1_concerns 2_have 3_intensified 4_because 5_of 6_china's 7_actions, 8_which 9_is 10_the 11_main 12_supplier.
0_these-0_these 1_concerns-1_concerns 2_have-2_have 3_intensified-3_intensified 4_due-4_because 5_to-5_of 7_actions-7_actions, 10_the-10_the 11_predominant-11_main 12_supplier.-12_supplier.

Source parse:
(S (NP (DT These) (NNS concerns)) (VP (AUX have) (VP (VBN intensified) (PP (IN due) (PP (TO to) (NP (DT the) (NNS actions) (PP (IN of) (NP (NNP China) (, ,) (NP (DT the) (JJ predominant) (NN supplier)))))))) (. .))
0 parse:
(S \n (NP (DT These) (NNS concerns)) (VP (VBP have) (VP (VBN gotten) (ADJP (JJR larger)) (SBAR (IN because) (S (PP (IN of) (NP (DT the) (NNS actions))) (PP (IN of) (NP (NNP China)))))))) .)
1 parse:
(S \n (NP (NNP China) (POS 's))\n (VP \n (VBZ actions) \n (VP \n (VBP have) \n (VP \n (VBN intensified) \n (NP \n (DT these) \n (NNS concerns)))))\n (. .))
(S (NP (NNP China) (POS 's)) (VP (VBZ actions) (VP (VBP have) (VP (VBN intensified) (NP (DT these) (NNS concerns))))) (. .))
2 parse:
(S (NP (DT These)) (VP (VBP are) (NP (NNS concerns) (SBAR (WHNP (WDT that)) (S (VP (VBP have) (VP (VBN been) (VP (VBN raised) (PP (IN by) (NP (NNP China))) (, ,) (SBAR (WHNP (WP who)) (S (VP (VBP are) (NP (DT a) (JJ major) (NN supplier))))))))))))
3 parse:
(S (NP (DT These) (NNS concerns)) (VP (VBP have) (VP (VBN intensified) (PP (IN because) (PP (IN of) (NP (NP (NNP China) (POS 's) (NNS actions)) (SBAR (WHNP (WDT which)) (S (VP (VBZ is) (NP (DT the) (JJ main) (NN supplier))))))))) (. .))

(a) Input

Aspect: Lexical, Best: 3, Worst: 2
Analysis:
0: Simplifies "intensified" with "gotten larger, "due to" with "because of" → high reward. Deletes "predominant supplier" - loss of specificity but retains core meaning, moderate penalty.
1: Retains the complex word "intensified", moderate penalty. Deletes "predominant supplier" - loss of specificity but retains core meaning, moderate penalty.
2: "intensified" → "raised" (significant meaning change), high penalty. Simplifies "predominant" -> "major", high reward. However, meaning distortion outweighs its partial simplification.
3: Retains the complex word "intensified", moderate penalty. Simplifies "predominant" with "main", "due to" with "because of", high reward. Keeps most of the original content.

Aspect: Structural, Best: 1, Worst: 2
Analysis:
0: Retains complex SBAR ("because of") and nested PPs, adding no structural improvement.
1: Flattens the source's structure into a clean active-voice NP–VP–NP ("China's actions have intensified concerns") → high reward.
2: Introduces two relative clauses ("that have been raised...", "who are..."), creating deeper SBAR nesting than the source → high penalty.
3: Embeds a relative clause ("which is…"), mirroring the source's nesting—moderate penalty.

Aspect: Overall, Best: 1, Worst: 2

(b) Corresponding output showing evaluations on lexical, structural, and overall dimensions.

Figure 10: Example 3 for sentence-level LLM-Judge

```
You are an evaluation system for document-level simplification. You will be provided with:
1)  One anchor example of document-level simplification.
It contains:
     • an original document
     • its gold simplified version
     • edits that were made to create the simplified version (relative to the original)
   - The anchors demonstrate:
   (a) what categories of edits are typically made
   (b) how frequently and how strongly those edits are applied
2) An original document
3) Four simplified versions of the original document (labeled 0, 1, 2, and 3)
--------------------------------------------------
YOUR TASK

Act as a document-level evaluation system.

Select:
- The BEST simplified document that most closely follows the edit patterns from the anchors,
while making sure the global coherence, discourse relations, and coreference are in good shape.
- The WORST simplified document that least follows the edit patterns from the anchors, while
also considering any major issues in coherence, discourse, or coreference.
--------------------------------------------------
OUTPUT FORMAT (strict):
{{
  "best": <index>,
  "worst": <index>
}}
The values must be integers from 0 to 3.
Strictly follow the above format and do not include any explanation or extra symbols.
```

Figure 11: Document-level evaluation guidelines provided to the judge LLM

```
System:
Please return the Penn Treebank-style constituency parse for the following sentence.
Only return the parse tree. Do not return any additional text or explanation.

User:
We can let you go with our cloaking device.

Assistant:
(S
  (NP (PRP We))
  (VP (MD can)
      (VP (VB let)
          (NP (PRP you))
          (VP (VB go)
              (PP (IN with)
                  (NP (PRP$ our) (NN cloaking) (NN device))))))
  (. .))

User:
{Input}
```

Figure 12: Prompts used for parsing

```
You are given two parallel documents: the original document and its simplified version.
Each sentence in both documents is prefixed with a sentence index in square brackets, e.g., [0],
[1], etc.
Your task is to align the corresponding sentences between the two documents.
One sentence in the original document may align to zero, one, or multiple sentences in the
simplified document.
Create a separate alignment entry for each aligned sentence pair.
If an original sentence does not align to any simplified sentence, include an alignment entry
with sent_simp_id as null and sent_simp as an empty string.


OUTPUT FORMAT (strict):
The output should be a JSON array of alignment entries.
Each alignment entry is a JSON object with the following structure:
{
  "id": <integer, starting from 0 and increasing>,
  "sent_orig_id": <integer, the sentence index in the original document as given in the square
brackets>,
  "sent_simp_id": <integer, the sentence index in the simplified document as given in the square
brackets, or null if no alignment>,
  "sent_orig": "<corresponding sentence from the original document, do not truncate>",
  "sent_simp": "<corresponding sentence from simplified document, do not truncate, or empty
string if no alignment>"
}


Please make sure all keys and values are present and complete as specified, and the output is
valid JSON.
DO NOT include any additional text, explanations, or comments in the output.
```

Figure 13: System messages for sentence alignment

```
Original Document:
[0]_Guglielmo giovanni maria marconi, 1st marquis of marconi (; 25 april 187420 july 1937)
was an italian inventor and electrical engineer, known for his pioneering work on long-
distance radio transmission, development of marconi's law, and a radio telegraph system.
[1]_He is credited as the inventor of radio, and he shared the 1909 nobel prize in physics
with karl ferdinand braun "in recognition of their contributions to the development of
wireless telegraphy".
[2]_Marconi was also an entrepreneur, businessman, and founder of the wireless telegraph &
signal company in the united kingdom in 1897 (which became the marconi company).
[3]_He succeeded in making an engineering and commercial success of radio by innovating and
building on the work of previous experimenters and physicists.
[4]_In 1929, marconi was ennobled as a marchese (marquis) by king victor emmanuel iii of
italy, and, in 1931, he set up vatican radio for pope pius xi.

Simplified Document:
[0]_Guglielmo Marconi was an Italian inventor and engineer, best known for his work in long-
distance radio communication.
[1]_He is often called the inventor of radio and played a key role in developing wireless
telegraphy.
[2]_In 1909, he shared the Nobel Prize in Physics with Karl Ferdinand Braun for their work in
wireless communication.
[3]_He made radio a practical and commercial success by building on the discoveries of
earlier scientists.
[4]_Marconi passed away in 1937, leaving behind a lasting legacy in the field of
communication.
```

(a) Input

```
[
  {
    "id": 0,
    "sent_orig_id": 0,
    "sent_simp_id": 0,
    "sent_orig": "Guglielmo giovanni maria marconi, 1st marquis of marconi (; 25 april 187420 july 1937)
was an italian inventor and electrical engineer, known for his pioneering work on long-distance radio
transmission, development of marconi's law, and a radio telegraph system.",
    "sent_simp": "Guglielmo Marconi was an Italian inventor and engineer, best known for his work in
long-distance radio communication."
  },
  {
    "id": 1,
    "sent_orig_id": 0,
    "sent_simp_id": 4,
    "sent_orig": "Guglielmo giovanni maria marconi, 1st marquis of marconi (; 25 april 187420 july 1937)
was an italian inventor and electrical engineer, known for his pioneering work on long-distance radio
transmission, development of marconi's law, and a radio telegraph system.",
    "sent_simp": "Marconi passed away in 1937, leaving behind a lasting legacy in the field of
communication."
  }
  {
    "id": 2,
    "sent_orig_id": 1,
    "sent_simp_id": 1,
    "sent_orig": "He is credited as the inventor of radio, and he shared the 1909 nobel prize in physics
with karl ferdinand braun 'in recognition of their contributions to the development of wireless
telegraphy'.",
    "sent_simp": "He is often called the inventor of radio and played a key role in developing wireless
telegraphy."
  },
  {
    "id": 3,
    "sent_orig_id": 1,
    "sent_simp_id": 2,
    "sent_orig": "He is credited as the inventor of radio, and he shared the 1909 nobel prize in physics
with karl ferdinand braun 'in recognition of their contributions to the development of wireless
telegraphy'.",
    "sent_simp": "In 1909, he shared the Nobel Prize in Physics with Karl Ferdinand Braun for their work
in wireless communication."
  },
  {
    "id": 4,
    "sent_orig_id": 2,
    "sent_simp_id": null,
    "sent_orig": "Marconi was also an entrepreneur, businessman, and founder of the wireless telegraph &
signal company in the united kingdom in 1897 (which became the marconi company).",
    "sent_simp": ""
  },
  {
    "id": 5,
    "sent_orig_id": 3,
    "sent_simp_id": 3,
    "sent_orig": "He succeeded in making an engineering and commercial success of radio by innovating and
building on the work of previous experimenters and physicists.",
    "sent_simp": "He made radio a practical and commercial success by building on the discoveries of
earlier scientists."
  },
  {
    "id": 6,
    "sent_orig_id": 4,
    "sent_simp_id": null,
    "sent_orig": "In 1929, marconi was ennobled as a marchese (marquis) by king victor emmanuel iii of
italy, and, in 1931, he set up vatican radio for pope pius xi.",
    "sent_simp": ""
  },
]
```

(b) Corresponding output

Figure 14: One-shot example for sentence alignment

```
You will be provided with the following:
    A source sentence and its simplified version.
    Word alignments between the source and the simplified sentence, structured as:
        words in the source sentence with indices.
        words in the simplified sentence with indices.
        word alignments following the format: sourceIndex_sourceWord-
simplifiedIndex_simplifiedWord

Your task: Act as an evaluation system, evaluate the lexical quality of the simplified
sentence referring to the word alignments and the following lexical evaluation
principles:
    Replacing difficult words with easier ones without changing the meaning, or only
slightly changing it. → High reward
    Replacing easy words with even simpler ones without changing the meaning, or only
slightly changing it. → Moderate reward
    Replacing words with more complex ones → High penalty
    Replacing words in a way that significantly changes the original meaning → High
penalty
    Deleting unimportant difficult words → Moderate reward
    Deleting easy words → Moderate penalty
    Deleting important information → High penalty
    Deleting, replacing, or omitting important named entities → High penalty
    Retaining difficult words → Moderate penalty
    Adding new complex words → High penalty

Please give me an evaluation report.
```

(a) Sent-Lexical

```
You will be provided with the following:
    A source sentence and its simplified version.
    Word alignments between the source and the simplified sentence, structured as:
        words in the source sentence with indices.
        words in the simplified sentence with indices.
        word alignments following the format: sourceIndex_sourceWord-
simplifiedIndex_simplifiedWord
    Penn Treebank-style parses for the source and each simplified sentence.

Your task: Act as an evaluation system, evaluate the overall quality of the simplified
sentence referring to the word alignments, parses, and the following evaluation
principles.
Lexical Evaluation Principles:
    Replacing difficult words with easier ones without changing the meaning, or only
slightly changing it. → High reward
    Replacing easy words with even simpler ones without changing the meaning, or only
slightly changing it. → Moderate reward
    Replacing words with more complex ones → High penalty
    Replacing words in a way that significantly changes the original meaning → High
penalty
    Deleting unimportant difficult words → Moderate reward
    Deleting easy words → Moderate penalty
    Deleting important information → High penalty
    Deleting, replacing, or omitting important named entities → High penalty
    Retaining difficult words → Moderate penalty
    Adding new complex words → High penalty

Structural Evaluation Principles:
    Simplifying difficult structures → High reward
    Simplifying easy structures → Moderate reward
    Using more difficult structures → High penalty
    Splitting long sentences → High reward
    Reordering for clarity → High reward
    Edits that do not contribute to simplicity or clarity → Moderate penalty
    Retaining difficult structures → High penalty

Please give me an evaluation report.
```

(b) Sent-Overall

Figure 15: Guidelines in sentence-level iterative feedback. Contents are from Figure 7, 8, 9, and 10.

Source sentence: We'll slip you through with our cloaking device.
Simplified sentence: With our cloaking devices, we can slip by you.

Alignment:
0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
0_with 1_our 2_cloaking 3_devices, 4_we 5_can 6_slip 7_by 8_you.
0_we'll-4_we 1_slip-6_slip 3_through-7_by 4_with-0_with 5_our-1_our 6_cloaking-2_cloaking 7_device.-8_you.

(a) Input of Example 1

'slip you through'→'slip by you' reverses the core meaning — high penalty. Also pluralizes "device" to "devices" without justification — light penalty for minor factual change. The difficult word 'cloaking' is retained — moderate penalty.

(b) Output of Example 1

Source sentence:. These concerns have intensified due to the actions of China, the predominant supplier.
Simplified sentence: These are concerns that have been raised by China, who are a major supplier.

Alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_these 1_are 2_concerns 3_that 4_have 5_been 6_raised 7_by 8_china, 9_who 10_are 11_a 12_major 13_supplier.
0_these-0_these 1_concerns-2_concerns 2_have-4_have 3_intensified-6_raised 8_of-7_by 9_china,-8_china, 10_the-11_a 11_predominant-12_major 12_supplier.-13_supplier.

(c) Input of Example 2

"intensified" → "raised" (significant meaning change), high penalty. Simplifies "predominant" -> "major", high reward. However, meaning distortion outweighs its partial simplification.

(d) Output of Example 2

Source sentence: The humans had grown complacent, having faced no concentrated robotic aggression during the last century of uneasy cold war.
Simplified sentence: The humans became complacent, having faced no major robotic attacks in the last century of cold war tensions.

Alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_the 1_humans 2_became 3_complacent, 4_having 5_faced 6_no 7_major 8_robotic 9_attacks 10_in 11_the 12_last 13_century 14_of 15_cold 16_war 17_tensions.
0_the-0_the 1_humans-1_humans 3_grown-2_became 4_complacent,-3_complacent, 5_having-4_having 6_faced-5_faced 7_no-6_no 8_concentrated-7_major 9_robotic-8_robotic 10_aggression-9_attacks 11_during-10_in 12_the-11_the 13_last-12_last 14_century-13_century 15_of-14_of 17_cold-15_cold 18_war.-17_tensions.

(e) Input of Example 3

"had grown complacent"→"became complacent": moderate reward. Retains "complacent" (difficult word): moderate penalty. "concentrated"→"major," "aggression"→"attacks": simpler synonyms—high reward.

(f) Output of Example 3

Figure 16: Three-shot examples of Sent-Lexical used for LLM-Judge in the iterative feedback setting.

Source sentence: We'll slip you through with our cloaking device.
Simplified sentence: With our cloaking devices, we can slip by you.

Alignment:
0_we'll 1_slip 2_you 3_through 4_with 5_our 6_cloaking 7_device.
0_with 1_our 2_cloaking 3_devices, 4_we 5_can 6_slip 7_by 8_you.
0_we'll-4_we 1_slip-6_slip 3_through-7_by 4_with-0_with 5_our-1_our 6_cloaking-2_cloaking 7_device.-8_you.

Source parse:
(S \n (NP (PRP We))\n (VP\n (MD 'll)\n (VP\n (VB slip)\n (NP (PRP you))\n (PP\n (IN through)\n (PP\n (IN with)\n (NP\n (NP (PRP$ our) (NN cloaking) (NN device))))))\n (. .))
Simplified parse:
(S \n (PP (IN With) \n (NP (PRP$ our) (NNS cloaking) (NNS devices)))\n (, ,)\n (NP (PRP we))\n (VP (MD can) \n (VP (VB slip) \n (PRT (RP by)) \n (NP (PRP you))))\n (. .))

(a) Input of Example 1

Lexical Analysis:
'slip you through'→'slip by you' reverses the core meaning — high penalty. Also pluralizes "device" to "devices" without justification — light penalty for minor factual change. The difficult word 'cloaking' is retained — moderate penalty.

Structural Analysis:
Fronting the PP ("With our…devices") aids clarity. However, it fails to simplify the source's nested PPs; instead, creates a new multi-layered VP (slip → by → you) without resolving recursion.

(b) Output of Example 1

Source sentence: The humans had grown complacent, having faced no concentrated robotic aggression during the last century of uneasy cold war.
Simplified sentence: The humans became complacent, having faced no major robotic attacks in the last century of cold war tensions.

Alignment:
0_the 1_humans 2_had 3_grown 4_complacent, 5_having 6_faced 7_no 8_concentrated 9_robotic 10_aggression 11_during 12_the 13_last 14_century 15_of 16_uneasy 17_cold 18_war.
0_the 1_humans 2_became 3_complacent, 4_having 5_faced 6_no 7_major 8_robotic 9_attacks 10_in 11_the 12_last 13_century 14_of 15_cold 16_war 17_tensions.
0_the-0_the 1_humans-1_humans 3_grown-2_became 4_complacent,-3_complacent, 5_having-4_having 6_faced-5_faced 7_no-6_no 8_concentrated-7_major 9_robotic-8_robotic 10_aggression-9_attacks 11_during-10_in 12_the-11_the 13_last-12_last 14_century-13_century 15_of-14_of 17_cold-15_cold 18_war.-17_tensions.

Source parse:
(S (NP (DT The) (NNS humans)) (VP (VBD had) (VP (VBN grown) (ADJP (JJ complacent), (S (VP (VBG having) (VP (VBN faced) (NP (DT no) (JJ concentrated) (JJ robotic) (NN aggression)) (PP (IN during) (NP (DT the) (JJ last) (NN century) (PP (IN of) (NP (JJ uneasy) (JJ cold) (NN war))))))))))) (. .))
Simplified parse:
(S (NP (DT The) (NNS humans)) (VP (VBD became) (ADJP (JJ complacent)) (, ,) (S (VP (VBG having) (VP (VBN faced) (NP (DT no) (JJ major) (JJ robotic) (NNS attacks)) (PP (IN in) (NP (DT the) (JJ last) (NN century) (PP (IN of) (NP (JJ cold) (NN war) (NNS tensions)))))))) (. .))

(c) Input of Example 2

Lexical Analysis:
"had grown complacent"→"became complacent": moderate reward. Retains "complacent" (difficult word): moderate penalty. "concentrated"→"major," "aggression"→"attacks": simpler synonyms—high reward.

Structural Analysis:
Retains the original complex participial construction ("having faced…"), only substituting simpler words, moderate structural penalty for not simplifying the structure.

(d) Output of Example 2

Source sentence:. These concerns have intensified due to the actions of China, the predominant supplier.
Simplified sentence: These are concerns that have been raised by China, who are a major supplier.

Alignment:
0_these 1_concerns 2_have 3_intensified 4_due 5_to 6_the 7_actions 8_of 9_china, 10_the 11_predominant 12_supplier.
0_these 1_are 2_concerns 3_that 4_have 5_been 6_raised 7_by 8_china, 9_who 10_are 11_a 12_major 13_supplier.
0_these-0_these 1_concerns-2_concerns 2_have-4_have 3_intensified-6_raised 8_of-7_by 9_china,-8_china, 10_the-11_a 11_predominant-12_major 12_supplier.-13_supplier.

Source parse:
(S (NP (DT These) (NNS concerns)) (VP (AUX have) (VP (VBN intensified) (PP (IN due) (PP (TO to) (NP (DT the) (NNS actions) (PP (IN of) (NP (NNP China) (, ,) (NP (DT the) (JJ predominant) (NN supplier))))))))) (. .))
Simplified parse:
(S (NP (DT These)) (VP (VBP are) (NP (NNS concerns) (SBAR (WHNP (WDT that)) (S (VP (VBP have) (VP (VBN been) (VP (VBN raised) (PP (IN by) (NP (NNP China))) (, ,) (SBAR (WHNP (WP who)) (S (VP (VBP are) (NP (DT a) (JJ major) (NN supplier))))))))))))

(e) Input of Example 3

Lexical Analysis:
"intensified" → "raised" (significant meaning change), high penalty. Simplifies "predominant" -> "major", high reward. However, meaning distortion outweighs its partial simplification.

Structural Analysis:
Introduces two relative clauses ("that have been raised...", "who are..."), creating deeper SBAR nesting than the source → high penalty.

(f) Output of Example 3

Figure 17: 3-shot examples of Sent-Overall used for LLM-Judge in the iterative feedback setting

```
System:
You will be provided a source sentence, a simplified sentence, and an evaluation report
assessing the simplification. Improve the simplified sentence according to the
evaluation. If there are no issues mentioned in the evaluation, return the original
simplified sentence. Return only the revised simplified sentence without numbering or
explanations.

User:
{Input}
```

Figure 18: Prompts used for self-correction in the iterative feedback setting at the sentence level

You are an evaluation system for document-level simplification.
You will be provided with:
1) One anchor example of document-level simplification. It contains:
   - an original document
   - its gold simplified version
   - edits that were made to create the simplified version (relative to the original)
   - The anchors demonstrate:
     (a) what categories of edits are typically made
     (b) how frequently and how strongly those edits are applied
2) An original document
3) A simplified version of the original document

Your task: Act as a document-level evaluation system, evaluate the quality of the simplified version by comparing it with the original document and the anchor example.

Please give me an evaluation report.

Figure 19: Guidelines in document-level iterative feedback.

```
System:
You will be provided a source document, a simplified document, and an evaluation report
assessing the simplification. Improve the simplified document according to the evaluation.
If there are no issues mentioned in the evaluation, return the original simplified document.
Return only the revised simplified document without numbering or explanations.


User:
{Input}
```

Figure 20: Prompts used for self-correction in the iterative feedback setting at the document level