HIT-YOU at TSAR 2025 Shared Task: Leveraging Similarity-Based Few-Shot Prompting, Round-Trip Translation, and Self-Refinement for Readability-Controlled Text Simplification

Mao Shimada* **and Kexin Bian*** **and Zhidong Ling and Mamoru Komachi**
Hitotsubashi University
{5123024a@g,dm240020@g,dd250009@g,mamoru.komachi@r}.hit-u.ac.jp

Abstract

We describe our submissions to the TSAR 2025 shared task on readability-controlled text simplification, which evaluates systems on their ability to adjust linguistic complexity to specified CEFR levels while preserving meaning and coherence. We explore two complementary frameworks that both build on LLMs and incorporate feedback from the shared-task CEFR classifiers. The first is an ensemble approach, which uses multiple LLMs to generate diverse candidates through zero-shot prompting, similarity-based few-shot prompting, and round-trip translation. The generated candidates are filtered by predicted CEFR level, and the final output is selected by an LLM judge. The second is a self-refinement loop, which uses a single LLM that begins with one candidate and iteratively revises it based on classifier feedback until it meets the target level or reaches a maximum iteration limit. Both approaches achieved competitive performance in the shared task. To our knowledge, this is among the first studies to apply round-trip translation and iterative self-refinement to controlled simplification, expanding the toolkit for reliable readability control.

1 Introduction

Text simplification aims to reduce the complexity of text while preserving meaning, thereby improving accessibility for language learners and readers with limited proficiency. The **TSAR 2025 Shared Task** (Alva-Manchego et al., 2025) focuses on *readability-controlled simplification*, where English passages at CEFR level B2 or above must be rewritten to a specified target level (A1, A2, or B1). Systems are evaluated on CEFR compliance, semantic similarity to the original text and references, with the challenge that no parallel training data is provided.

*These authors contributed equally to this work.

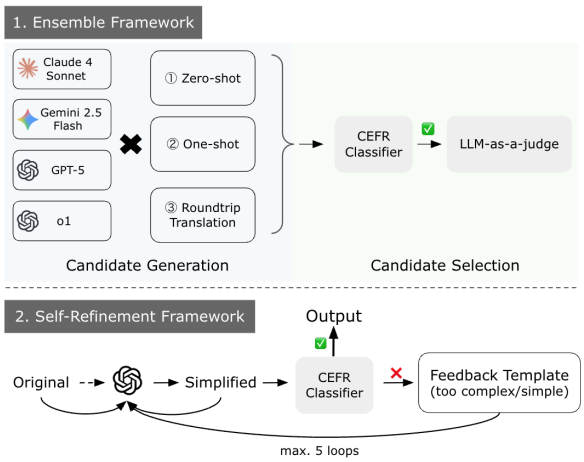

Figure 1: Illustration of the two frameworks used in our system. (1) The **Ensemble Framework** aggregates diverse model–prompt candidates through classifier filtering and LLM-based selection, while (2) the **Self-Refinement Framework** iteratively adjusts a single model's output using classifier feedback.

Over the past years, research has evolved from traditional rule-based methods (Shardlow, 2014) to neural sequence-to-sequence and pretrained Transformer models (Sheang and Saggion, 2021; Li et al., 2024). More recently, large language models (LLMs) with controllable and instruction-based generation have become the dominant trend in text simplification. However, ensuring precise readability control remains a core challenge (Barayan et al., 2025; Tran et al., 2025).

Previous research has explored generation mechanisms that encourage structural and lexical reformulation that might facilitate controllable simplification. Cross-lingual and translation-based pipelines, for example, have been shown to induce diverse edits such as synonym substitutions and word reorderings (Stahlberg et al., 2022; Vlantis et al., 2024), effects that can be leveraged to achieve finer readability control. In another line of work, iterative refinement frameworks such as

Proceedings of the Fourth Workshop on Text Simplification, Accessibility and Readability (TSAR 2025), pages 231–241

SELF-REFINE (Madaan et al., 2023) improve LLM outputs through self-feedback guided by external evaluators, offering a general way of enforcing generation constraints such as readability control.

Building on these insights, we explore two frameworks for readability-controlled simplification, both guided by the shared-task CEFR classifier (Figure 1).

- The **ensemble framework** integrates multiple prompting strategies to generate diverse candidates, including a novel similarity-based few-shot prompting and round-trip translation. Candidates are filtered by the shared-task CEFR classifier, and the final output is selected by an LLM judge. We submitted two ensemble runs that differ in the choice of judge model.

- The **self-refinement framework** employs iterative simplification, where a single candidate is repeatedly revised under classifier feedback until the target CEFR level is reached.

2 System Description

2.1 Ensemble Models

Our first two submissions use a multi-stage ensemble pipeline to produce a single simplification from a diverse pool of candidates. The pipeline is designed to combine the strengths of different LLMs and prompting strategies. It proceeds in two stages: (i) candidate generation, and (ii) candidate selection through filtering and final judgment.

2.1.1 Candidate Generation

To encourage diversity, we generate candidate simplifications using four proprietary LLMs: GPT-5, Gemini 2.5 Flash, Claude Sonnet 4, and o1.[1] Each model is prompted under three strategies described below (see Appendix A for full prompt texts). For each model–prompt configuration, we draw four samples at temperature 1, yielding a total of $4 \times 3 \times 4 = 48$ candidates for each test input.

Zero-shot We designed zero-shot prompts through observation of the trial data and iterative prompt engineering, adapting instructions to each target CEFR level. All prompts specified both the source and target CEFR levels. For A2, the prompt additionally required the model to simplify the paragraph into a direct subject–verb–object structure. For B1, we first annotated each word in the original paragraph with CEFR levels using `CEFRpy`[2], and then instructed the model to only replace words above B2 with simpler alternatives by including the corresponding wordlist in the prompt. We further constrained the output by limiting each sentence to a maximum of 29 words.

Similarity-based Few-shot We extended the zero-shot prompts by inserting in-context demonstrations selected from the trial data based on source similarity. [3] For each test input, cosine similarity was computed between its sentence embedding and those of 20 trial sources using `Alibaba-NLP/gte-large-en-v1.5` (Zhang et al., 2024; Li et al., 2023). The k most similar sources and their paired references at the target level were then added to the prompt as demonstrations, so a k-shot setup corresponds to including the top-k most similar source–reference sets. In this work, we set $k = 1$, yielding a one-shot setup.

Round-trip We employ a round-trip translation approach, using different intermediate languages depending on the target CEFR level. For A2, the original English paragraph is first translated into Chinese while being simplified to the target level. For B1, it is translated into German without explicit simplification at this stage. In both cases, only the translated paragraph is then used as input to translate it back into English, during which the model is instructed to simplify it to the target level.

2.1.2 Candidate Selection

After candidate generation, we apply a two-step selection process consisting of a rule-based filter and an LLM judge.

Filtering Candidates are first scored by the official CEFR classifier ensemble (Alva-Manchego et al., 2025), with the final label taken from the classifier with the highest confidence, exactly as provided in the shared task evaluation script. We retain those predicted to match the target level exactly; if none remain, we fall back to candidates that are one level above or below the target. This

[1] Model versions: `gpt-5-2025-08-07`, `gemini-2.5-flash`, `claude-sonnet-4-20250514`, and `o1-2024-12-17`.

[2] `CEFRpy` is a Python module built on spaCy that tokenizes text and maps words to CEFR levels based on wordlist resources. Documentation available at `https://maximax67.github.io/cefrpy/`.

[3] We include a comparison with random sampling in Appendix D

step ensures that the judge compares only plausibly compliant outputs, reducing the risk of selecting a fluent but level-mismatched candidate and streamlining the decision to a more competitive set.

LLM-as-a-judge The surviving candidates are then presented to an LLM judge, which is instructed to select the simplification closest in meaning to the original paragraph. To explore how different LLM architectures perform as judges, we experiment with two variants, corresponding to our first two submissions:

- **Gemini ensemble (run1_gemini_ensemble).** On the trial datasets, Gemini 2.5 Flash demonstrates good performance in both the candidate generation and selection stages, and is therefore incorporated into our set of official submissions.
- **GPT-5 ensemble (run2_gpt_ensemble_4).** GPT-5, used as the judging model, demonstrated reliable and consistent candidate selection on the trial data, performing comparably to other high-performing models while being more efficient to deploy. It is thus adopted as one of our official submissions.

2.2 Self-refinement (run3_self_refine)

In addition to the ensemble pipelines, we submitted a self-refinement system that relies on a single model, GPT-5. Unlike the ensemble approach, this system iteratively adjusts one candidate under feedback guidance. While in principle the feedback signal could be derived from various metrics, we chose to use the CEFR classifier, since readability control is one of the two core evaluation criteria and, in our experience, the aspect more frequently failed by baseline outputs.

Initialization The process begins with a zero-shot simplification produced by the model (as described in § 2.1.1), which serves as the initial candidate for refinement.

Iteration At each step, the CEFR classifier predicts the level of the current candidate. If the predicted level is above the target, the feedback instructs the model to simplify vocabulary and sentence structure; if it is below, it encourages the use of slightly more complex constructions. In both cases, the feedback explicitly requires the model to preserve the original meaning and maintain natural, coherent text. The exact prompt template and feedback generation rules are provided in Appendix A. This feedback, together with the original document and the candidate text, is then provided to the model, which generates a revised simplification. The loop continues for up to five iterations.

Stopping criterion The loop terminates early if the classifier predicts that the target level has been reached. Otherwise, the final candidate after the fifth iteration is returned. We adopt this policy based on the assumption that the last iteration represents the most refined version.

3 Results

Table 1 shows the performance of our three submitted runs. We report results on the official TSAR-2025 shared task metrics: **RMSE** for CEFR compliance, and MeaningBERT (Beauchemin et al., 2023) for semantic similarity to the original (**mBERT-Orig**) and reference (**mBERT-Ref**) texts, respectively, as defined by the organizers (Alva-Manchego et al., 2025).

When considering only the best run per team, our team **ranks third out of 20 teams** overall. Our GPT-5 and Gemini ensemble runs achieve AUTORANK [4] values of 3.61 and 3.67, respectively, among 48 submitted runs, demonstrating competitive performance across all official metrics. The Self-Refinement run ranks at AUTORANK = 5.46, showing slightly weaker CEFR control and meaning preservation, but operates with much lower computational cost, relying on a single model rather than multi-model ensembling and repeated generations.

4 Discussion

Effectiveness of Ensemble We demonstrate the contribution of the ensemble process by comparing its performance against the strongest single-model baselines, shown in the upper part of Table 1, which represent an empirical upper bound of individual model overall performance across the prompting configurations. The ensemble outperforms these baselines across all dimensions. Although the substantial improvement in level control is partly enforced by design through classifier-based filtering, these results indicate that incorporating diverse model–prompt pairs captures complementary strengths beyond any individual configuration.

[4] Linearly rescaled rank where 1 denotes the best performing system; see (Alva-Manchego et al., 2025) for details.

System	RMSE ↓	mBERT-Orig ↑	mBERT-Ref ↑
Claude $_{(0\text{-shot})}$	0.595	0.836	0.820
GPT-5 $_{(0\text{-shot})}$	0.620	**0.848**	0.821
Gemini $_{(0\text{-shot})}$	**0.497**	0.812	0.811
o1 $_{(RT)}$	0.630	0.828	**0.835**
Gemini Ensemble	0.187	**0.863**	0.833
GPT-5 Ensemble	**0.158**	0.852	**0.835**
Self-Refinement	0.245	0.822	0.820

Table 1: Comparison of best single-model baselines (top) and submitted runs (bottom) on the official TSAR-2025 metrics. For each model, best configurations were determined via a weighted composite of normalized scores following median–interpercentile scaling (Alva-Manchego et al., 2025), averaged across four generations. Values in parentheses indicate the prompting configuration in which the best overall score was achieved.

While we did not perform full ablation studies, we examined the distribution of final candidates chosen by the LLM judges to assess the necessity of using multiple models and prompt variants. We find that the selections were distributed across all four models and three prompting strategies (Appendix B), indicating that the ensemble benefited from the diversity of system outputs. Overall, the gains from combining multiple model–prompt pairs suggest that different systems may excel on different types of inputs or aspects of simplification, leading to complementary effects when aggregated, a possibility we plan to explore in future work.

Effectiveness of Self-refinement Compared to its zero-shot GPT-5 counterpart (Table 1), the self-refinement process substantially reduces readability-classification error through its classifier-guided feedback loop, while maintaining comparable similarity to human-written references. This demonstrates that automatic feedback can effectively steer outputs toward the intended readability level without a major loss in fluency and coherence.

Self-refinement successfully corrected most outputs that initially missed the target level. Out of 200 instances, 129 (64.5%) met the target immediately, but a further 54 (27.0%) converged only through iterative refinement, confirming the value of the approach in recovering difficult cases. However, the distribution of refinement depth shows diminishing returns: 31 (15.5%), 14 (7.0%), 7 (3.5%), and 2 (1.0%) cases requiring one to four iterations, suggesting that classifier feedback yields most of its benefit in early steps. However, a small fraction of cases (8.5%) never converged, underscoring that the current feedback signal alone is not sufficient for all inputs.

Another limitation is reflected in the trade-off between readability control and meaning preservation (mBERT-Orig), which decreased from 0.848 to 0.822 across all instances. For the subset that underwent refinement, the drop was more pronounced ($0.85 \rightarrow 0.79$), suggesting that repeated simplification can impair content preservation when guided solely by readability feedback. Future extensions could incorporate additional signals to balance readability, meaning, and naturalness more holistically during refinement.

Model behavior across prompting strategies To better understand how prompting design influences readability control and meaning preservation, we analyze model behavior across the three setups within the ensemble framework (zero-shot, one-shot, and round-trip). Detailed results are provided in Appendix C. Overall, we find that models varied substantially in their behavior.

In the zero-shot setting, Gemini achieved the lowest RMSE (0.50) but also exhibited the weakest meaning preservation, while o1 showed the opposite pattern, with high fidelity but poor compliance (RMSE = 0.80). Claude and GPT-5 fell in between. We also examined level-specific classification results, which revealed clear level-dependent differences: Gemini and Claude aligned more successfully with A2 than B1, GPT-5 was relatively balanced, and o1 performed better at B1 than A2. These baseline differences are particularly notable given that all models were prompted identically (§2.1.1), indicating that they internalize and act upon level-control instructions in distinct ways.

Adding one-shot demonstrations generally decreased meaning preservation across all models. In terms of reference similarity, o1 benefited the most

from the example, showing consistent gains across both target levels, whereas Gemini and GPT-5 remained relatively stable and Claude experienced a noticeable drop. Effects on compliance were mixed: Claude and o1 showed clear improvement at both levels, GPT-5 improved slightly (mainly due to gains at A2), while Gemini improved at A2 but worsened at B1, increasing its overall error. These results indicate that one-shot examples can enhance level control, particularly at A2. This asymmetry may stem from the nature of the target levels: demonstrations provide clear guidance for simplifying to A2, where shorter sentences and simpler vocabulary reliably signal compliance. By contrast, B1 allows for more diverse realizations, so reliance on a single example can bias the model toward an unrepresentative solution, reducing compliance consistency.

Notably, models with stronger instruction-following or reasoning capabilities, such as o1, appeared to benefit more from demonstrations, as seen in its consistent gains in both reference similarity and compliance across levels. We leave the systematic evaluation of demonstration effects on simplification outcomes and their underlying factors to future work.

The round-trip approach proved generally effective as a way to induce simplification through translation. However, because this setup differs fundamentally in design, we treat its results as descriptive rather than directly comparable to the zero- and one-shot conditions. The choice of intermediate language appears to influence both level control and meaning preservation, likely reflecting differences in linguistic structure and translation bias. We include in Appendix E a brief discussion of intermediate-language choice and the two-step prompting design.

5 Conclusion and Future Work

We presented three systems for the TSAR 2025 shared task on readability-controlled text simplification: two ensemble pipelines that combine diverse generators with an LLM-as-a-judge, and a self-refinement loop guided by classifier feedback. In developing the ensemble system, we explored multiple prompting strategies, including style- and vocabulary-based instruction, similarity-based few-shot prompting, and round-trip translation.

Our analysis highlights systematic differences in how current LLMs interpret and operationalize level-control instructions, as well as the effects of incorporating demonstrations and classifier-guided feedback. Overall, the results indicate that LLMs under our frameworks form a strong foundation for controllable simplification, but that the trade-off between reliable level control and meaning preservation remains unresolved.

Looking ahead, there are several directions for strengthening our two frameworks under the current evaluation setting. For the ensemble, future work could focus on developing more principled aggregation strategies and analyzing the decision behavior of LLM-as-a-judge models, particularly how they balance readability against meaning when selecting outputs. For the self-refinement framework, richer feedback signals beyond classifier guidance (e.g., semantic similarity, stylistic alignment) could help stabilize convergence and better preserve meaning.

At the same time, our observations during system development hinted that the current automatic evaluation may be brittle. Small stylistic or structural variations can produce large metric shifts, even when readability and meaning remain comparable to human readers. For example, we observed that formatting differences such as line breaks could noticeably affect classifier predictions (Appendix F). Revisiting evaluation design to include human- or reader-centered assessments would therefore provide a more reliable view of simplification quality and practical usefulness.

Limitations

Our systems rely on proprietary LLMs whose training data and update histories are not publicly available, limiting reproducibility and interpretability. While we examined the overall effects of prompting and classifier feedback, our analysis was not exhaustive. Further controlled analyses would be needed to isolate the contribution of each model and prompt variations. We also did not conduct full ablation studies or controlled comparisons across intermediate languages, so some observed trends remain descriptive. Finally, we did not conduct a systematic analysis of the LLM-as-a-judge component, including potential biases, decision consistency, or sensitivity to prompt phrasing. As a result, its contribution to overall system performance is not fully understood.

Acknowledgments

This work was supported by the National Institute of Information and Communications Technology (NICT) under the "Research and Development of externally controllable modeling of multimodal information to enhance the accuracy of automatic translation."

Lay Summary

This paper describes our systems for the TSAR 2025 shared task, which challenges participants to rewrite English texts at different levels of difficulty (A2–B1) without losing meaning, so that language learners can better understand them.

We built two kinds of systems.

- An **ensemble** system combines outputs from several models using different prompting strategies, such as adding style and vocabulary instructions, providing examples, or translating text through another language ("round-trip"). It then uses another LLM as a "judge" to select the best simplification.
- A **self-refinement** system gives the model feedback on whether its output is too simple or too complex and lets it revise itself until a readability classifier confirms that it matches the target level.

Both of our systems performed competitively in the shared task. The ensemble system performed better than any of its components alone, possibly because different model–prompt combinations behaved quite differently (some were better at controlling difficulty, while others preserved meaning more faithfully). This suggests that combining multiple models and prompts can produce more balanced results, and that different combinations may work better for different kinds of source texts.

The self-refinement system also achieved strong results, producing texts that matched the intended difficulty level with much less computation. This shows that giving models simple feedback can be an efficient way to make their writing easier or harder when needed.

However, our experiments show that while large language models can simplify text in a controlled way, it is still difficult to achieve both precise level control and full preservation of meaning.

References

Fernando Alva-Manchego, Regina Stodden, Joseph Marvin Imperial, Abdullah Barayan, Kai North, and Harish Tayyar Madabushi. 2025. Findings of the TSAR 2025 shared task on readability-controlled text simplification. In *Proceedings of the Fourth Workshop on Text Simplification, Accessibility, and Readability (TSAR 2025)*, Suzhou, China. Association for Computational Linguistics.

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.

David Beauchemin, Horacio Saggion, and Richard Khoury. 2023. Meaningbert: assessing meaning preservation between sentences. *Frontiers in Artificial Intelligence*, Volume 6 - 2023.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Zihao Li, Samuel Belkadi, Nicolo Micheletti, Lifeng Han, Matthew Shardlow, and Goran Nenadic. 2024. Large language models for biomedical text simplification: Promising but not there yet. *Preprint*, arXiv:2408.03871.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

Matthew Shardlow. 2014. A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications(IJACSA), Special Issue on Natural Language Processing 2014*, 4(1).

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

Felix Stahlberg, Aashish Kumar, Chris Alberti, and Shankar Kumar. 2022. Conciseness: An overlooked language task. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 43–56, Abu Dhabi, United Arab Emirates (Virtual). Association for Computational Linguistics.

Hieu Tran, Zonghai Yao, Lingxi Li, and Hong Yu. 2025. ReadCtrl: Personalizing text generation with readability-controlled instruction learning. In *Proceedings of the Fourth Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2025)*, pages 19–36, Albuquerque, New Mexico, US. Association for Computational Linguistics.

Daniel Vlantis, Iva Gornishka, and Shuai Wang. 2024. Benchmarking the simplification of Dutch municipal text. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2217–2226, Torino, Italia. ELRA and ICCL.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. *arXiv preprint arXiv:2407.19669*.

A Prompts and Feedback Templates

A.1 Zero-shot prompts

Zero-shot prompt at the A2 level.

```
Please simplify the following paragraph
into a direct SVO structure,
transforming it from {ORIGINAL_CEFR}
CEFR level to {TARGET_CEFR} CEFR level,
making it easier to read and understand
for {TARGET_CEFR} CEFR level English
learners. Return only the simplified
paragraph, without explanation or extra
text.
paragraph: {PARAGRAPH}
```

Zero-shot prompt at the B1 level.

```
Please simplify the following
paragraph, transforming it from
{ORIGINAL_CEFR} CEFR level to
{TARGET_CEFR} CEFR level, making it
easier to read and understand for
{TARGET_CEFR} CEFR level English
learners.
Only replace difficult words with
easier alternatives. Use the provided
list of difficult words for
replacements.
-Keep the overall sentence structure.
-Don't change meaning and delete any of
the given information.
-Ensure that each sentence has no more
than 29 words.
Return only the simplified paragraph,
without explanation or extra text.
paragraph: {PARAGRAPH}
list of difficult words: {WORD_LIST}
```

A.2 Few-shot prompts

The placeholder `{few_shot_examples}` corresponds to the pair of original and simplified paragraph retrieved in the format:

```
Original: [paragraph retrieved]
Simplified: [corresponding reference
paragraph at target CEFR level]
```

Few-shot prompt at the A2 level.

```
Please simplify the following paragraph
into a direct SVO structure,
transforming it from {ORIGINAL_CEFR}
CEFR level to {TARGET_CEFR} CEFR level,
making it easier to read and understand
for {TARGET_CEFR} CEFR level English
learners.
Here are some examples of how to
simplify text for this level:
{few_shot_examples}.
Return only the simplified paragraph,
without explanation or extra text.
paragraph: {PARAGRAPH}
```

Few-shot prompt at the B1 level.

```
Please simplify the following
paragraph, transforming it from
{ORIGINAL_CEFR} CEFR level to
{TARGET_CEFR} CEFR level, making it
easier to read and understand for
{TARGET_CEFR} CEFR level English
learners.
Here are some examples of how to
simplify text for this level:
{few_shot_examples}.
Only replace difficult words with
easier alternatives. Use the provided
list of difficult words for
replacements.
-Keep the overall sentence structure.
-Don't change meaning and delete any of
the given information.
-Ensure that each sentence has no more
than 29 words.
Return only the simplified paragraph,
without explanation or extra text.
paragraph: {PARAGRAPH}
list of difficult words: {WORD_LIST}
```

A.3 Round-trip prompts

Round-trip prompt at the A2 level.

- **English → Chinese + simplify**

```
Please translate the following English
paragraph into Chinese and simplify it
for {TARGET_CEFR} CEFR level learners
of Chinese. Return only the Chinese
paragraph, without explanation or extra
text.
English paragraph: {PARAGRAPH}
```

- **Chinese → Englsih + simplify**

```
Please translate the following Chinese
paragraph into English and simplify it
for {TARGET_CEFR} CEFR level learners
of English. Return only the translated
and simplified paragraph, without
explanation or extra text.
Chinese paragraph: {PARAGRAPH}
```

Round-trip prompt at the B1 level.

- **English → German**

```
Please translate the following paragraph
into German.  Return only the German
paragraph, without explanation or extra
text.
English paragraph: {PARAGRAPH}
```

- **German → Englsih + simplify**

```
Please translate the following German
paragraph into English, making it easier
to read and understand by {TARGET_CEFR}
CEFR level English learners.  Return
only the translated and simplified
paragraph, without explanation or extra
text.
German paragraph: {PARAGRAPH}
```

A.4 Candidate Selection prompt

```
YOU MUST FOLLOW OUTPUT RULES EXACTLY.
Choose the one closest to the original
paragraph. Output MUST be a single
digit 1-{len(candidates)} on its own
line. No other text.
Original paragraph: {ORIGINAL}
Simplified paragraphs:
{candidates_text}
```

A.5 Self-Refinement

Refinement prompt.

```
You are an expert in text
simplification. Your previous attempt
to simplify the document was not
successful and did not meet the quality
criteria.

Original Document: {document_text}
Your Previous Attempt that failed:
{previous_step_output}
Feedback on Previous Attempt:
{feedback_message}

Please use this feedback to generate a
new, improved simplification from the
original document. Reply only with the
simplified text. Do not add
explanations, labels, or extra
comments.
Simplified text:
```

Feedback template. The classifier output is converted into natural-language feedback according to the predicted level:

Too complex (predicted above target): "The simplified text is too complex. It was predicted as CEFR {predicted_cefr} but the target is {target_cefr}. Please simplify vocabulary and sentence structure while preserving the original meaning and keeping the text natural and coherent."

Too simple (predicted below target): "The simplified text is too simple. It was predicted as CEFR {predicted_cefr} but the target is {target_cefr}. Please use slightly more complex vocabulary and sentence structures while preserving the original meaning and keeping the text natural and coherent."

B Distribution of final candidates

Table 2 shows the distribution of final candidates chosen by the LLM judges across prompting strategies and models.

	Zero-shot	Few-shot	Round-trip	**Total**
Gemini-as-judge				
Claude	30	3	12	**45**
GPT-5	33	19	10	**62**
Gemini	18	7	22	**47**
o1	27	14	5	**46**
Total	**108**	**43**	**49**	**200**
GPT-as-judge				
Claude	12	4	12	**28**
GPT-5	51	20	13	**84**
Gemini	15	8	21	**44**
o1	27	7	10	**44**
Total	**105**	**39**	**56**	**200**

Table 2: Distribution of final candidates selected under Gemini- and GPT-as-judge.

C Model–prompt performance by target level

This appendix provides detailed results across all prompting configurations (zero-shot, one-shot, and round-trip). Table 3 summarizes overall test performance across the four models, averaged over five runs. Tables 4 and 5 present level-specific results for A2 and B1.

D Effect of Sample Selection Strategies for Few-shot

On the trial data, we also explored sample selection strategies with GPT-5, comparing random versus similarity-based one-shot examples (Table 6). Here, both random and similarity-based one-shot examples reduced RMSE relative to zero-shot, but random examples achieved the lowest RMSE with relatively low variance, while similarity-based examples performed comparably. In both cases, the gain in compliance came with a clear drop in meaning preservation.

E Effect of Intermediate Language for Round-trip

This appendix provides the results for the Round-trip model on the trial datasets. Table 7 summarizes

Model	RMSE ↓			mBERT-Orig ↑			mBERT-Ref ↑		
	0-shot	1-shot	RT	0-shot	1-shot	RT	0-shot	1-shot	RT
Claude	0.595 (.018)	**0.536** (.027)	0.606 (.020)	0.836 (.002)	0.796 (.001)	0.809 (.000)	0.820 (.002)	0.811 (.005)	0.825 (.010)
GPT-5	0.620 (.029)	0.597 (.009)	0.569 (.020)	0.848 (.005)	0.803 (.001)	0.793 (.000)	**0.821** (.004)	0.821 (.002)	0.819 (.010)
Gemini	**0.497** (.018)	0.574 (.020)	**0.539** (.020)	0.812 (.004)	0.769 (.005)	0.793 (.000)	0.811 (.003)	0.808 (.008)	0.819 (.000)
o1	0.797 (.013)	0.706 (.015)	0.630 (.030)	**0.868** (.002)	**0.839** (.004)	**0.828** (.000)	0.818 (.004)	**0.826** (.003)	**0.835** (.000)

Table 3: Average performance under zero-shot, similarity-based one-shot, and round-trip prompting on test data across all models (5 runs per setting). Standard deviations shown in parentheses.

Model	A2 RMSE ↓			A2 mBERT-Orig ↑			A2 mBERT-Ref ↑		
	0-shot	1-shot	RT	0-shot	1-shot	RT	0-shot	1-shot	RT
Claude	0.444 (.040)	0.433 (.060)	0.548 (.050)	0.754 (.000)	0.748 (.000)	0.756 (.000)	0.810 (.000)	0.801 (.010)	0.795 (.010)
GPT-5	0.595 (.020)	0.551 (.030)	0.436 (.020)	0.792 (.010)	0.776 (.000)	0.727 (.010)	0.819 (.010)	0.818 (.000)	0.789 (.010)
Gemini	**0.359** (.040)	**0.266** (.040)	**0.364** (.050)	0.730 (.000)	0.715 (.010)	0.722 (.000)	0.788 (.010)	0.783 (.010)	0.788 (.010)
o1	0.857 (.020)	0.743 (.030)	0.601 (.040)	**0.840** (.010)	**0.818** (.010)	**0.779** (.010)	**0.820** (.010)	**0.827** (.000)	**0.806** (.000)

Table 4: A2-level results under zero-shot, one-shot, and round-trip prompting.

Model	B1 RMSE ↓			B1 mBERT-Orig ↑			B1 mBERT-Ref ↑		
	0-shot	1-shot	RT	0-shot	1-shot	RT	0-shot	1-shot	RT
Claude	0.714 (.010)	**0.620** (.010)	**0.656** (.060)	**0.917** (.000)	0.843 (.000)	0.863 (.000)	0.831 (.000)	0.821 (.000)	0.854 (.000)
GPT-5	0.644 (.040)	0.638 (.030)	0.676 (.020)	0.904 (.000)	0.830 (.000)	0.858 (.000)	0.823 (.000)	0.824 (.000)	0.849 (.010)
Gemini	**0.603** (.020)	0.767 (.020)	0.669 (.030)	0.894 (.010)	0.824 (.000)	0.864 (.000)	**0.833** (.000)	**0.834** (.010)	0.849 (.000)
o1	0.733 (.030)	0.666 (.030)	0.658 (.030)	0.896 (.000)	**0.859** (.010)	**0.877** (.000)	0.816 (.000)	0.824 (.000)	**0.865** (.000)

Table 5: B1-level results under zero-shot, one-shot, and round-trip prompting.

Setting	RMSE ↓	mBERT-Orig ↑	mBERT-Ref ↑
Zero-shot	0.709 ± 0.056	$\mathbf{0.853 \pm 0.008}$	$\mathbf{0.793 \pm 0.006}$
Random 1-shot	$\mathbf{0.532 \pm 0.024}$	0.801 ± 0.010	0.783 ± 0.006
Similarity 1-shot	0.540 ± 0.097	0.799 ± 0.008	0.779 ± 0.007

Table 6: Average performance ($\pm$ standard deviation) of GPT-5 with zero-shot, random one-shot, and similarity-based one-shot prompting on trial data (3 runs per setting).

the performance of each intermediate language, measured by CEFR accuracy, meaning preservation score, and similarity to the reference score, averaged across five runs. Also, Table 8 reports a comparison of simplified paragraphs with and without simplification in English-to-Chinese translation, based on MeaningBERT (mBERT in short).

Round-trip and comparing with other intermediate languages. We implemented round-trip prompts with multiple intermediate languages on the trial datasets. Besides Chinese and German (included in the submission file), we also employed Spanish, Japanese, French, and Indonesian. Our findings are: from Table 7 , at A2, Chinese achieved the highest accuracy but the poorest meaning preservation and similarity-to-reference scores, whereas French and Indonesian showed the lowest accuracy but the highest Orig-BERT score. At B1, the highest accuracy was obtained by Chinese, German, Spanish, and Japanese; however, Chinese also showed the lowest accuracy, the same as Indonesian, indicating a large variance in its performance. Spanish achieved the highest scores for both meaning preservation and similarity to reference and German also achieves adequately high. These results indicate the different tendencies of intermediate languages in matching the target CEFR level at A2 and B1.

Furthermore, we found a two-step simplification approach as in A2 is also effective. From Table 7 , Chinese achieved an accuracy of 1.0 at A2. The reason for ineffectiveness at B1 is; the two-step simplification did not adjust the output toward the target CEFR level, but rather accumulated the effect of simplification, resulting in an oversimplification beyond B1. Additionally, compared to simple translation, the meaning preservation score

decreases in most paragraphs, but not for similarity to the reference: as illustrated by the example in Table 8, in about 25% of the paragraphs the score was actually higher, with simplified outputs also tending to have shorter sentences, with an average of 7.47 sentences after simplification compared to 6.76 without simplification, while original and reference paragraphs respectively contain 4.35 and 5.55 on average.

F Formatting

We observed that our self-refinement run contained substantially more newlines than the ensemble runs. At first we attributed this to the refinement process itself, hypothesizing that LLMs might exploit formatting as a way to adjust difficulty in the loop. Further inspection of zero-shot baselines revealed that only GPT-5 appeared to use newlines as a mechanism for level control, producing many at the B1 level (3.0 on average per output) but almost none at A2 (0.07), whereas other models inserted almost none across levels (Table 9).

However, we did find that the refinement process affects formatting. Among the instances that underwent refinement, newlines significantly decreased at B1 (0.63) but increased at A2 (1.06). This suggests that the refinement loop not only adjusts lexical and syntactic complexity, but also affects surface formatting.

Notably, removing newlines from the outputs substantially altered the CEFR classifier's predictions, as seen in Table 10, indicating that the classifier is also sensitive to formatting cues.

Intermediate language	Chinese	+simplify	German	+simplify	Spanish	+simplify	Japanese	French	Indonesian
Target CEFR: A2									
Acc-min	0.500	**0.900**	0.600	0.750	0.500	0.800	0.500	0.450	0.450
Acc-avg	0.620	**0.950**	0.625	0.815	0.550	0.900	0.590	0.600	0.530
Acc-max	0.750	**1.000**	0.700	0.850	0.600	0.950	0.700	0.650	0.600
mBERT-Orig	0.776	0.713	0.797	0.715	**0.806**	0.716	0.777	0.803	0.805
mBERT-Ref	0.743	0.721	**0.755**	0.744	0.740	0.734	0.732	0.739	0.750
Target CEFR: B1									
Acc-min	0.350	0.250	**0.500**	0.400	0.450	0.300	0.450	0.400	0.350
Acc-avg	0.500	0.300	**0.563**	0.415	0.540	0.385	0.540	0.460	0.490
Acc-max	**0.650**	0.350	**0.650**	0.450	**0.650**	0.450	**0.650**	0.500	0.600
mBERT-Orig	0.840	0.767	0.858	0.785	**0.869**	0.812	0.837	0.858	0.853
mBERT-Ref	0.827	0.776	0.826	0.792	**0.850**	0.814	0.841	0.836	0.829

Table 7: Comparison of intermediate languages based on CEFR accuracy, meaning preservation score, and similarity to reference score on the trial dataset

	Simplified paragraph	MeaningBERT
Without simplification	Small animals like birds, squirrels, mice, and insects are common in many cities and towns. But recently, news from all over the world talks a lot about wild animals coming into cities. Bears have been seen in parks in Vancouver. Leopards walk on the streets of Mumbai. And wild boars are in gardens in Berlin. What happens when bigger animals come into our cities? Are they welcome? Or do people see them as dangerous or a problem?	MeaningBERT-orig: **0.823** MeaningBERT-ref: 0.700
With simplification	Small animals live in many towns and cities. For example, birds, squirrels, mice, and insects. But now, more wild animals are coming into cities. Newspapers write about it. For example, there are bears in parks in Vancouver. Leopards are on streets in Mumbai. Wild pigs are in gardens in Berlin. What if bigger animals come to cities? Will we welcome them? Or will we think they are dangerous or harmful?	MeaningBERT-orig: 0.701 MeaningBERT-ref: **0.846**

Table 8: Comparison of simplified paragraphs with and without simplification in English-to-Chinese translation.

Model	A2	B1
Claude	0.01	0.00
GPT-5	0.07	3.02
Gemini	0.00	0.00
o1	0.00	0.24

Table 9: Average number of newlines per output in zero-shot simplifications on test data by model and target CEFR level.

Run	Formatting	RMSE	mBERT-Orig
GPT-5 Ensemble	Original	0.1225	0.8511
	\n removed	0.2550	0.8503
Gemini Ensemble	Original	0.0707	0.8621
	\n removed	0.1871	0.8621

Table 10: Effect of removing newline characters (\n) on automatic evaluation metrics.