Iterative Critique-Driven Text Simplification: Targeted Enhancement of Complex Definitions with Small Language Models

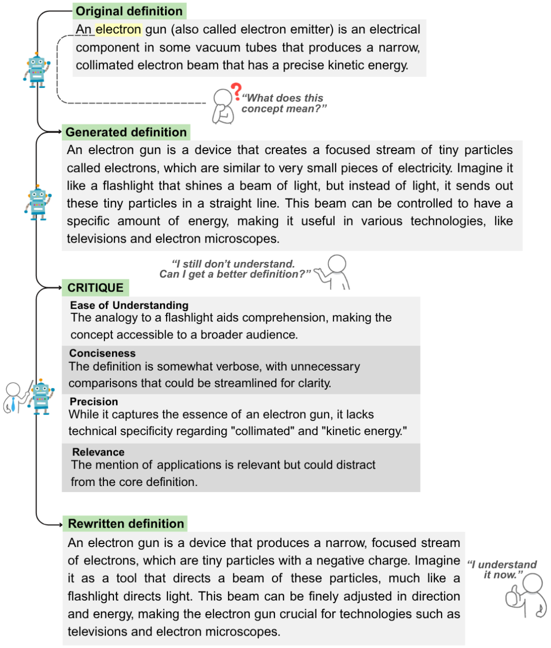
Veer Chheda **Avantika Sankhe** **Aaditya Ghaisas**
Dwarkadas J. Sanghvi College of Engineering, India
{veerchheda3525, avantikasankhe1 , aadityaghaisas0703}@gmail.com,

Abstract

Difficult and unfamiliar concepts often hinder comprehension for lay audiences, especially in technical and educational domains. This motivates the usage of large language models (LLMs) for the process of text simplification (TS). In this work, we propose an iterative refinement framework that aims to simplify definitions by carefully handling complex terminology and domain-specific expressions. The obtained definition is reprocessed based on the critique, making refinements in successive iterations. We emphasize the use of small language models (SLMs) due to their faster response times and cost-efficient deployment. Human evaluations of the definitions produced at each refinement stage indicate consistent improvements in our specified evaluation criteria. We evaluate both LLM-as-a-judge score and human assessments along with automated metrics like BERTScore, BLEU-4, which provided supporting evidence for the effectiveness of our approach. Our work highlights the use of LLMs mimicking human-like feedback system in a TS task catering to a reader's specific cognitive needs. Thus, we find that an iterative, critique-driven method can be an effective strategy for the simplification of dense or technical texts, particularly in domains where jargon impedes understanding.

1 Introduction

Text simplification (TS) is a core task in natural language processing which aims to reduce the lexical and semantic complexity of the text while preserving its original meaning, making the content more accessible to lay audiences. It has a wide application in various fields such as education, law, healthcare, and overall information access; enabling a wide range of people, including children, language learners, individuals with cognitive or reading disabilities, etc., to readily comprehend complex or unfamiliar content (Espinosa-Zaragoza et al., 2023).



Original definition

An electron gun (also called electron emitter) is an electrical component in some vacuum tubes that produces a narrow, collimated electron beam that has a precise kinetic energy.

Generated definition

An electron gun is a device that creates a focused stream of tiny particles called electrons, which are similar to very small pieces of electricity. Imagine it like a flashlight that shines a beam of light, but instead of light, it sends out these tiny particles in a straight line. This beam can be controlled to have a specific amount of energy, making it useful in various technologies, like televisions and electron microscopes.

CRITIQUE

Ease of Understanding
The analogy to a flashlight aids comprehension, making the concept accessible to a broader audience.

Conciseness
The definition is somewhat verbose, with unnecessary comparisons that could be streamlined for clarity.

Precision
While it captures the essence of an electron gun, it lacks technical specificity regarding "collimated" and "kinetic energy."

Relevance
The mention of applications is relevant but could distract from the core definition.

Rewritten definition

An electron gun is a device that produces a narrow, focused stream of electrons, which are tiny particles with a negative charge. Imagine it as a tool that directs a beam of these particles, much like a flashlight directs light. This beam can be finely adjusted in direction and energy, making the electron gun crucial for technologies such as televisions and electron microscopes.

"What does this concept mean?"
"I still don't understand. Can I get a better definition?"
"I understand it now."

Figure 1: An example of a definition from WikiDomains dataset containing a potentially 'difficult concept', refined through iteration using critique-based LLM feedback.

Recent advancements have demonstrated that state-of-the-art LLMs significantly outperform traditional approaches (Kew et al., 2023; Zhang and Lapata, 2017; Feng et al., 2023). Even a casual perusal of content involves complex cognitive processes (Evans et al., 2024). Moreover, human cognitive systems are innately limited in their ability to process complex or unfamiliar information (Sweller, 1988; Kahneman, 2011). For readers encountering such domain-specific content, the complex linguistic structures can add to the cognitive workload, overwhelming their working memory (Song, 2011). Context is necessary to understand any concept upon reading, and struggles commonly arise due to overly complex language, jargon, lack of examples or analogies, and insuffi-

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1081–1096
December 20-24, 2025 ©2025 Association for Computational Linguistics

cient or overly detailed explanations (Asthana et al., 2024). Reader's information needs differ substantially even within a single discipline; becoming more distant across varied domains (Guo et al., 2023).

Iterative refinement is a promising approach to improve text generation in NLP tasks, including TS where balancing simplicity and meaning often requires multiple adjustment cycles. Human editors typically approach revision similarly, applying a variety of edit types over multiple passes and document segments to ensure the text is easy to understand (Laban et al., 2023). Here, editors may also be conceptualized as "critics" that provide targeted feedback catered to the reader's needs. Computational models that mirror this process can potentially achieve more effective results.

Motivated by this, we propose a novel iterative refinement framework for TS focusing on generating definitions of terms with *difficult concepts* (Asthana et al., 2024), that are easy to understand and tailored according to human cognitive principles. Unlike prior work, we appoint a 'critic' LLM that simulates human-guided simplification to provide feedback to the text generator. We define criteria stated in Table 1 for it to evaluate the generated definition iteratively to tailor best possible results, as illustrated in Figure 1. To evaluate the performance of our framework, we use WIKIDOMAINS dataset (Asthana et al., 2024) which is a collection of 22k domain-specific definitions spanning 13 academic domains from Wikipedia, with each definition having a *difficult concept* identified and selected by the creators of this dataset.

The remainder of this paper is structured as follows: Section 2 reviews related work; Section 3 describes our methodology; Section 4 covers experimental setup and evaluation; Section 5 presents results; Section 6 delineates human evaluation.

2 Related Work

Recent work in TS has increasingly emphasized cognitive accessibility and the simplification of targeted domain-specific concepts, with LLMs at the forefront. Frameworks like PAIR (Hua and Wang, 2020) have shown notable gains in generation quality, and incorporating cognitive principles has led to improvements even on general simplification benchmarks (Chamovitz and Abend, 2022). To help readers grasp complex definitions, targeted concept simplification has been proposed (Asthana et al., 2024), definitions are simplified by carefully addressing difficult concepts that might hinder an individual's learning. Parallelly, research has explored readability-controlled simplification in zero-shot settings using instruction-tuned LLMs, though these face challenges in handling extreme simplification and reveal limitations in automatic evaluation metrics (Barayan et al., 2025). Addressing the rigidity of single-pass models, Laban et al. (2023) proposed aligning simplification with human-like iterative refinement. Building on this, self-refinement techniques (Madaan et al., 2023), where an LLM critiques and revises its own outputs, have shown notable gains. Critique-guided decoding (Kim et al., 2023) further improved output quality. Mondal et al. (2024) brought advances in simplification of Indic languages via a multilingual critic in low-resource settings. Complementing the above, Ke et al. (2024) proposed CritiqueLLM, a model trained to generate informative and fine-grained critiques.

Criterion	**Description**
Ease of Understanding	Is it easy for someone with no background to follow?
Precision	Is the explanation specific and correct? Avoid vagueness or technical jargon.
Conciseness	Is it clear and brief, without unnecessary elaboration?
Relevance	Does it stick closely to the original definition without adding unrelated ideas?

Table 1: Evaluation criteria for assessing the quality and critiquing rewritten definitions.

3 Proposed Methodology

We propose an iterative refinement framework for targeted simplification through LLM critic to mimic human-in-the-loop revisions. The generator model generates a candidate simplification and the critic evaluates it on a set of parameters. The critique is used to rewrite the definition for further fine-grained simplification. This loop continues for a specified number of times, yielding different fine-grained iterations of the same definition.

Let d be the original definition, c the difficult concept, and t the target term. Let LLM_θ be a generative model and critic_ϕ a critic model.

$$r_0 = \text{LLM}_\theta(d, c, t), \quad s_0 = \text{critic}_\phi(d, r_0, c, t)$$

r_0 is the initial rewrite of the original definition that allows the generative model to use an appro-

Model	BLEU-4 (↑)			Density (↑)			ΔLength (↓)			BERTScore (↑)			ΔAoA (↑)			ΔFRE (↑)		
	Base	I1	I2	Base	I1	I2	Base	I1	I2	Base	I1	I2	Base	I1	I2	Base	I1	I2
LlaMa 3.2 1B	0.05	**0.52**	**0.75**	0.27	**0.78**	**0.88**	58.93	33.31	29.41	0.86	0.83	0.94	1.31	1.25	1.50	20.02	16.59	20.33
LlaMa 3.2 3B	0.06	0.31	0.51	0.29	0.60	0.75	43.02	**32.84**	**27.89**	0.87	0.92	0.94	1.75	1.84	1.92	11.58	12.67	13.81
Gemma 3 1B	**0.09**	0.20	0.45	**0.37**	0.50	0.69	**23.21**	58.55	57.51	0.87	0.90	0.90	1.19	1.94	2.07	13.65	11.04	12.97
Gemma 3 4B	0.03	0.15	0.19	0.23	0.52	0.56	76.13	79.72	80.78	0.86	0.90	0.91	**1.89**	1.97	2.08	**20.37**	16.26	18.48
Qwen3 1.7B	**0.09**	0.46	0.67	0.34	0.74	0.85	51.19	60.40	63.30	**0.89**	**0.95**	0.97	0.88	1.16	1.27	16.85	17.46	18.31
Qwen3 4B	0.08	0.35	0.48	0.31	0.68	0.76	53.70	51.09	51.25	**0.89**	0.94	0.96	1.84	1.12	1.96	19.22	18.49	19.87
GPT 4o	0.08	0.25	0.42	0.33	0.59	0.73	41.47	34.03	30.05	0.86	0.93	**0.98**	0.54	**2.12**	**2.27**	19.26	**20.15**	**20.43**

Table 2: Performance of simplification models across metrics and refinement iterations (Base, I1, I2). Arrows indicate desirable trends: ↑ = higher is better, ↓ = lower is better. Bold indicates the best score per column.

priate strategy for simplification. Instead of using a particular strategy (simplify, explain, give an example, use an analogy, etc. (Asthana et al., 2024)), our framework is strategy agnostic. LLM$_\theta$ uses a strategy appropriate to the definition, which is then refined by critic$_\phi$. This allows for more flexible generations based on the context of the definition. Critic$_\phi$ evaluates the rewrites based on the parameters present in Table 1 and returns a feedback for guiding the rewrite. We define the refinement function $\psi_{\theta,\phi}$:

$$(r_i, s_i) = \psi_{\theta,\phi}(r_{i-1}, s_{i-1}; d, c, t)$$

$$= (\text{LLM}_\theta(r_{i-1}, c, t, s_{i-1}),\ \text{critic}_\phi(d, r_i, c, t))$$

To ensure that each iteration leads to a meaningful improvement in readability, we employ the Flesch Reading Ease (FRE) score as a stopping criterion. The iterative loop is terminated if the score does not improve relative to the previous iteration. To maintain the experimental simplicity and feasibility of human evaluation, the maximum number of refinement iterations is capped at two.

4 Experimentation

4.1 Setup

All tests were run on 2 NVIDIA Tesla T4 GPUs. We report the inference time and memory usage of models in Appendix. We used QLoRA quantization (Dettmers et al., 2023) via the Unsloth framework (Han and team, 2023), significantly reducing memory and computation needs, allowing for scalable experimentation. We utilized the WIKIDOMAINS test set which contains 3304 definitions.

4.2 Models

We focus primarily on open-weight smaller language models (SLMs) because they can be efficiently deployed on local, on-premises GPUs, enabling cost-effective fine-tuning on different reading grades and simplification styles. We use LlaMa 3.2's 1B and 3B variants (Van Der Maaten et al., 2024) , Gemma3's 1B and 4B variants (Kamath and team, 2025), Qwen-3 1.7B and 4B variants(Yang and Qwen Team, 2025) and GPT-4o as a SOTA baseline, critic and LLM-as-a-judge due to its strong reasoning abilities and intelligence (Hurst and Team, 2024). We also experimented using SLMs as critics but found that their feedback was often less consistent and caused cascading errors due to hallucinations. The rewritten definitions shifted focus from explaining the term to defining the difficult concept. This is digressing from our primary task and hence, we do not report this experimentation.

4.3 Metrics

We follow Asthana et al. (2024)'s suite of automated metrics that include **BLEU-4** (Papineni et al., 2002), **Density** (Grusky et al., 2018), **BERTScore** (Zhang et al., 2020), **Age of Acquisition (AoA)** (Kuperman et al., 2012), **text length** and **Flesch Reading Ease** (Flesch, 1948).

We also evaluate the quality of our rewrites through a human survey described in Section 6. To complement automated metrics and human ratings, we employ LLM-as-a-judge (Zheng et al., 2023; Fu et al., 2024; Liu et al., 2023) as a reference-free automated judge for evaluating the quality of the outputs based on the parameters present in Table 1. The implementation is described in detail in Appendix A.1.

5 Results

Table 2 and Table 3 display the automated and LLM-as-a-judge metrics respectively. 'Base' indicates initial rewrite, 'I1' and 'I2' indicate the following iterations of the rewritten definition. All models display an increase in BLEU-4, Density and BERTScore demonstrating that refinement helps retain the original syntactic and semantic context. Further, ΔAoA and ΔFRE also increase steadily,

Model	Ease of Understanding			Relevance			Conciseness			Precision		
	Base	I1	I2	Base	I1	I2	Base	I1	I2	Base	I1	I2
LlaMa 3.2 1B	3.52	3.82	3.79	3.06	3.49	3.55	2.86	3.35	3.43	3.03	3.40	3.51
LlaMa 3.2 3B	3.62	3.85	3.90	3.20	4.20	4.25	2.74	3.40	3.63	2.76	4.03	4.11
Gemma 3 1B	3.79	3.46	3.30	3.67	4.06	3.93	**3.56**	3.23	2.91	3.05	4.00	3.98
Gemma 3 4B	3.87	4.02	4.04	3.73	4.61	4.20	2.90	3.57	3.48	3.08	4.41	3.98
Qwen3 1.7B	3.85	4.14	4.00	4.02	4.45	4.47	3.06	3.40	3.29	**3.46**	4.18	4.30
Qwen3 4B	3.73	4.22	4.10	3.74	4.48	4.35	2.84	3.58	3.65	3.12	4.15	4.11
GPT-4o	**4.68**	**4.80**	**4.64**	**4.69**	**5.00**	**4.95**	3.44	**4.15**	**4.22**	3.39	**4.88**	**4.58**

Table 3: LLM-as-a-judge scores of simplification models across iterations (Base, I1, I2) on four dimensions: Ease of Understanding, Relevance, Conciseness, and Precision. old indicates the best score per column.

denoting increased readability without losing technical precision. GPT-4o consistently outperforms other models across all dimensions and iterations, yielding the highest scores. All models benefit from iterative refinement, reflected in progressively higher LLM-as-a-judge scores, particularly for relevance and conciseness. This is also reflected in human evaluation in Section 6. We also report inferential statistics on inference time, memory usage and failure by stopping criteria in Table 7.

6 Human Evaluation

We conducted a human survey using a total of 18 non-expert annotators randomly selected from the general public on a volunteer basis. The distribution of annotators is as follows:

1. **High School**: 5 participants
2. **Bachelor's Degree**: 7 participants
3. **Master's Degree**: 4 participants
4. **Doctoral Degree (PhD)**: 2 participants

Our focus was on assessing the performance of our methodology using standard human evaluation metrics and also conduct the study with a notably higher number of participants compared to prior work (Asthana et al., 2024; Cardon et al., 2022; Espinosa-Zaragoza et al., 2023), where the number of annotators has typically been limited. Each participant was presented with 50 definitions proportionally sampled to reflect the domain distribution outlined in the dataset and rate each rewritten definition on a Likert scale as per the criterion described in Table 1. The rating criteria are detailed further in A.2.2. For each definition, we provided the following:

1. The term
2. The original definition (for context)
3. The identified *difficult concept*
4. Three rewritten definitions (corresponding to Base, I1, and I2 outputs)

The definitions were provided in a randomized order to remove any potential bias that could occur while evaluating. In addition to the ratings, annotators were also asked to select the version of definition which helped them fully understand the term and the *difficult concept*. In the final question *"What would help you understand it better?"*, this selected definition was subjected to further scrutiny where evaluators determined if it was already acceptable according to them or needed any improvements catering to their specific needs. The WIKIDOMAINS dataset (Asthana et al., 2024) has 22k definitions across 13 academic domains. However, the distribution of definitions across the domains is not uniform. Thus, we sampled a set of 50 definitions from the test set preserving the original domain distribution. We followed the breakdown shown in Table 4.

Domain	**#Definitions**	**#Selected**
Food & Drink	1,403	2
Performing arts	322	1
Business & Economics	1,539	3
Politics & Government	2,267	4
Biology	7,200	17
Chemistry	957	2
Computing	2,083	7
Earth and Environment	1,314	2
Mathematics	1,747	4
Medicine & Health	2,939	4
Physics	741	2
Engineering	89	1
Technology	7	1
Total	**22,561**	**50**

Table 4: Domains, total number of definitions, and number of selected definitions in proportion for the human survey from the WIKIDOMAINS dataset.

To assess inter-rater reliability, we calculated

Krippendorff's Alpha which came out to be 0.3752. Low inter-annotator agreement is expected due to the varying educational backgrounds of surveyors. Likewise, a definition concise enough for a PhD graduate might lack sufficient clarity or context for a high-school student. Complete setup of our survey including exact wording of our questions is provided in the Appendix A.2. A series of Kruskal-Wallis H-tests were conducted, revealing statistically significant differences across education levels for all four criteria: Ease of Understanding ($H(3) = 282.998$, $p < .001$), Conciseness ($H(3) = 186.680$, $p < .001$), Relevance ($H(3) = 285.180$, $p < .001$), and Precision ($H(3) = 310.666$, $p < .001$). These findings indicate that at least one education level group significantly differed from others in their perceptions of definition quality.

Suggestion	**Bachelor's**	**Master's**	**High School**	**PhD**
Analogy	8.63	3.07	10.29	24.22
Detailed Explanation	16.24	15.35	25.86	13.28
Example	17.26	7.46	23.48	21.88
Keep concise	0.00	0.44	0.00	0.00
Not needed	41.37	53.51	15.04	28.91
Simplification	15.99	20.18	24.80	11.72
Lacks relevancy	0.00	0.00	0.26	0.00

Table 5: Scope for improvement percentages by education level

Post-hoc Tukey HSD tests, used to identify specific pairwise differences, consistently showed that Bachelor's degree holders generally provided higher ratings across most categories compared to other groups, while High School graduates often provided lower ratings, highlighting a potential need for further simplification for this demographic. Notably, PhD holders also rated most criteria significantly higher than High School and sometimes Master's degree holders, which might reflect a greater comfort with nuanced or technically precise language often encountered in advanced academic pursuits. The specific significant pairwise comparisons, including mean differences and adjusted p-values, are summarized in Table 9.

Participants' suggestions for improving definitions varied across education levels, as detailed in Table 5. A prominent finding was the proportion of Not needed suggestions, which was highest among Master's (53.51%) and Bachelor's (41.37%) degree holders, suggesting that these groups found the definitions largely satisfactory. In contrast, High School graduates provided 'Not needed' suggestions at a much lower rate (15.04%). They also showed a higher propensity for suggesting Detailed Explanation (25.86%), Example (23.48%), and Simplification (24.80%), indicating a preference for more elaborate and accessible definitions. PhD holders, while also frequently suggesting Not needed (28.91%), showed a comparatively high percentage for Analogy (24.22%) and Example (21.88%), potentially reflecting their analytical approach to understanding concepts.

EL	**I**	**EoU**	**Conciseness**	**Relevance**	**Precision**
Bachelor's	Base	3.25 ± 1.51	3.05 ± 1.31	3.22 ± 1.45	3.17 ± 1.29
	I1	3.13 ± 1.54	3.13 ± 1.32	3.14 ± 1.47	3.25 ± 1.31
	I2	**3.35 ± 1.59**	**3.33 ± 1.30**	**3.43 ± 1.44**	**3.55 ± 1.29**
High School	Base	4.35 ± 0.84	3.88 ± 0.98	4.24 ± 0.81	3.96 ± 0.83
	I1	4.34 ± 0.72	3.93 ± 0.91	4.28 ± 0.75	4.20 ± 0.77
	I2	**4.36 ± 0.72**	**4.03 ± 0.87**	**4.34 ± 0.71**	**4.30 ± 0.68**
Master's	Base	**4.33 ± 0.78**	**3.76 ± 1.19**	**4.07 ± 0.88**	**4.24 ± 0.75**
	I1	4.26 ± 0.88	3.76 ± 1.14	4.00 ± 0.78	4.20 ± 0.84
	I2	4.31 ± 0.77	3.70 ± 1.04	4.01 ± 0.74	4.21 ± 0.76
PhD	Base	3.81 ± 1.20	3.50 ± 0.83	**3.70 ± 0.75**	3.60 ± 0.89
	I1	4.01 ± 1.10	**3.62 ± 1.01**	3.59 ± 0.87	3.68 ± 0.96
	I2	**4.02 ± 1.07**	3.62 ± 0.85	3.69 ± 0.84	**3.70 ± 0.87**

Table 6: Mean ± standard deviation of survey responses across iterations and education levels for four text quality dimensions. (EoU - Ease of Understanding , EL - Education Level , I - Iteration)

7 Conclusion

We presented a novel iterative refinement framework for targeted text simplification with a particular focus on improving accessibility for lay audiences. As opposed to a plain rewrite of a definition, we appointed an LLM to critique the said definition on the basis of accessibility criteria closely tied with cognitive principles and provide human-like feedback. By iteratively refining the generated outputs, we demonstrated significant improvements in TS. We experimented on domain-specific definitions compiled in the WikiDomains dataset across three SLM families and included GPT-4o as a baseline. Furthermore, our human evaluation involving a diverse pool of participants validated the effectiveness of our methodology. Thus, our work highlights the importance of critique-driven iterative refinement in the TS; as well as the need to user-specific simplification strategies in general.

8 Limitations

A key limitation of our study is that the human evaluation was conducted with only 18 participants from diverse educational backgrounds (high school to PhD). While this lends some credence to our study, it may restrict the generalization to a population. Furthermore, the limited number of annotators may not fully capture the diversity of backgrounds, reading abilities or information needs present in the broader target population. As such, conclusions drawn from these human evaluations should be interpreted with caution and future work should seek to involve a larger and more representative pool of participants to strengthen the validity and reliability of the results. Further, we only noticed correlation of semantic and syntactic metrics (BLEU-4, BERTScore and Density) with LLM-as-a-judge scores. There was no definitive correlation with Human Evaluation which calls for development of robust metrics that take the background of the *difficult concept* into account. While strategy agnostic and iterative frameworks enhance the readability of definitions, there is scope to include the reader's educational background and reading preferences into the methodology.

Ethics Statement

This work uses only public domain datasets and does not make use of any personal data. The human evaluators were volunteers from a diverse pool of educational levels. Our system is intended solely for informational and research purposes.

Acknowledgements

We would like to express our sincere appreciation to Unsloth for providing an efficient and accessible framework that enabled low-resource fine-tuning and quantization of large language models. Their contributions were instrumental in scaling our experiments across different model sizes while maintaining computational feasibility. We also extend our gratitude to the human annotators who participated in our survey.

References

Sumit Asthana, Hannah Rashkin, Elizabeth Clark, Fantine Huot, and Mirella Lapata. 2024. Evaluating llms for targeted concept simplification for domain-specific texts. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, page 6208–6226. Association for Computational Linguistics.

Abdulaziz Barayan et al. 2025. Zero-shot readability-controlled sentence simplification with instruction-tuned large language models. In Proceedings of the 30th International Conference on Computational Linguistics (COLING 2025).

Rémi Cardon, Adrien Bibal, Rodrigo Wilkens, David Alfter, Magali Norré, Adeline Müller, Watrin Patrick, and Thomas François. 2022. Linguistic corpus annotation for automatic text simplification evaluation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 1842–1866, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Eytan Chamovitz and Omri Abend. 2022. Cognitive simplification operations improve text simplification. In Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL), pages 241–265, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Isabel Espinosa-Zaragoza, José Abreu-Salas, Elena Lloret, Paloma Moreda, and Manuel Palomar. 2023. A review of research-based automatic text simplification tools. In Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, pages 321–330, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Isabel Espinosa-Zaragoza, José Abreu-Salas, Paloma Moreda, and Manuel Palomar. 2023. Automatic text simplification for people with cognitive disabilities: Resource creation within the ClearText project. In Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability, pages 68–77, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Paul Evans, Maarten Vansteenkiste, Patrick Parker, Aaron Kingsford-Smith, and Shanshan Zhou. 2024. Cognitive load theory and its relationships with motivation: a self-determination theory perspective. Educational Psychology Review, 36(1):7.

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

Rudolf Flesch. 1948. A new readability yardstick. Journal of Applied Psychology, 32(3):221–233.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2024. GPTScore: Evaluate as you desire. 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 6556–6576, Mexico City, Mexico. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Yue Guo, Joseph Chee Chang, Maria Antoniak, Erin Bransom, Trevor Cohen, Lucy Lu Wang, and Tal August. 2023. Personalized jargon identification for enhanced interdisciplinary communication.

Han and Unsloth team. 2023. Unsloth.

Xinyu Hua and Lu Wang. 2020. PAIR: Planning and iterative refinement in pre-trained transformers for long text generation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 781–793, Online. Association for Computational Linguistics.

Hurst and OpenAI Team. 2024. Gpt-4o system card.

Daniel Kahneman. 2011. Thinking, fast and slow. Farrar, Straus and Giroux, New York.

Kamath and Gemma team. 2025. Gemma 3: Technical Report. arXiv preprint arXiv:2503.19786. Published Mar122025.

Pei Ke, Bosi Wen, Zhuoer Feng, Xiao Liu, Xuanyu Lei, Jiale Cheng, Shengyuan Wang, Aohan Zeng, Yuxiao Dong, Hongning Wang, Jie Tang, and Minlie Huang. 2024. Critiquellm: Towards an informative critique generation model for evaluation of large language model generation.

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, Singapore. Association for Computational Linguistics.

Minbeom Kim, Hwanhee Lee, KangMin Yoo, Joonsuk Park, Hwaran Lee, and Kyomin Jung. 2023. Critic-guided decoding for controlled text generation. In Findings of the Association for Computational Linguistics: ACL2023, pages 4598–4612, Toronto, Canada. Association for Computational Linguistics.

Victor Kuperman, Hans Stadthagen-Gonzalez, and Marc Brysbaert. 2012. Age-of-acquisition ratings for 30,000 english words. Behavior Research Methods, 44(4):978–990.

Philippe Laban, Jesse Vig, Wojciech Kryscinski, Shafiq Joty, Caiming Xiong, and Chien-Sheng Jason 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.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 2511–2522, Singapore. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Stephen Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shruti Prabhumoye, Yiming Yang, Sonal Gupta, Bodhisattwa Prasad Majumder, Karl Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. arXiv preprint arXiv:2303.17651.

Sneha Mondal, Ritika Ritika, Ashish Agrawal, Preethi Jyothi, and Aravindan Raghuveer. 2024. Dimsim: Distilled multilingual critics for indic text simplification. In Findings of the Association for Computational Linguistics: ACL 2024, pages 16093–16109, Bangkok, Thailand. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pages 311–318.

Minjung Song. 2011. Effects of Background Context and Signaling on Comprehension Recall and Cognitive Load: The Perspective of Cognitive Load Theory. Ph.d. dissertation, University of Nebraska - Lincoln, Lincoln, Nebraska. Advisor: Roger Bruning.

John Sweller. 1988. Cognitive load during problem solving: Effects on learning. Cognitive Science, 12(2):257–285.

Laurens Van Der Maaten et al. 2024. The llama 3 herd of models. arXiv preprint arXiv:2407.21783.

Yang and Qwen Team. 2025. Qwen3 technical report.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In International Conference on Learning Representations.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

A Appendix

A.1 Implementation Details

A.1.1 Model Cards

We utilize open-source large language models, all converted with Unsloth's efficient 4-bit quantization.Below, we provide links to official model cards for each model:

- unsloth/LlaMa-3.2-1B-Instruct-unsloth-bnb-4bit
- unsloth/LlaMa-3.2-3B-Instruct-unsloth-bnb-4bit
- unsloth/Qwen3-1.7B-unsloth-bnb-4bit
- unsloth/Qwen3-4B-unsloth-bnb-4bit
- unsloth/gemma-3-1b-it-unsloth-bnb-4bit
- unsloth/gemma-3-4b-it-unsloth-bnb-4bit
- OpenAI GPT-4o

A.1.2 Inference Settings

Models were loaded with a maximum sequence length of 4096 tokens. Generation was performed using greedy decoding for deterministic outputs by setting `do_sample=False`. Since OpenAI API does not support greedy decoding, we try to employ deterministic outputs by setting `temperature=0`. The inference time per model is given in Table 7.

A.1.3 Instruction Template

Our prompting strategy follows a structured, instruction-driven approach to guide the model in rewriting complex definitions. Each prompt specifies the term, the identified difficult concept, and clear directions to simplify the target definition by the most appropriate strategy according to the model. For the initial rewrite, the model receives the following prompt:

```
Rewrite the definition of the term f‘{term}’ by making the difficult concept f‘{difficult_concept}’ easier to understand. You may simplify the wording, explain in more detail, give an analogy, or provide an example - use whatever is most effective. Return only a definition and nothing else.
```

For generating subsequent iterations, iterative refinement is enabled by incorporating a structured critique, generated either by the following prompt:

```
You are critiquing how well the rewritten definition of the term f‘{term}’ having the concept f‘{difficult_concept}’ elucidates the intended meaning of the original definition. Your goal is to judge how effectively it communicates the intended meaning to a layperson.

Original definition (intended meaning): f‘{original_definition}’

Rewritten definition: f‘{rewritten_definition}’

Judge the rewritten definition on these 4 criteria:

1. Ease of Understanding - Is it easy for someone with no background to follow?
2. Precision - Is the explanation specific and correct? Avoid vagueness or technical jargon.
3. Conciseness - Is it clear and brief, without unnecessary elaboration?
4. Relevance - Does it stick closely to the original definition without adding unrelated ideas?
```

Iterative prompt for each subsequent iteration is as follows:

```
Rewrite the definition of the term f‘{term}’ having a difficult concept, to understand the original meaning of the definition better. Use the
```

Model	Base (s)	I1 (s)	I2 (s)	Base (MB)	I1 (MB)	I2 (MB)
LlaMa 3.2 1B	2.74 ± 0.06	1.95 ± 0.03	1.97 ± 0.04	1129.3 ± 17.4	1142.5 ± 6.8	1148.5 ± 10.9
LlaMa 3.2 3B	2.36 ± 0.19	2.18 ± 0.06	1.70 ± 0.02	2395.0 ± 35.6	2411.3 ± 31.7	2429.6 ± 47.2
Qwen3 1.7B	5.30 ± 0.08	5.36 ± 0.09	5.34 ± 0.05	1474.0 ± 16.0	1523.7 ± 7.7	1523.3 ± 21.4
Qwen3 4B	6.73 ± 0.17	6.93 ± 0.13	7.06 ± 0.11	3557.8 ± 31.2	3612.0 ± 71.6	3613.6 ± 40.3
Gemma 3 1B*	2.99 ± 0.06	3.40 ± 0.65	3.64 ± 0.47	990.9 ± 5.5	1006.0 ± 14.3	1008.1 ± 14.2
Gemma 3 4B*	10.24 ± 0.15	15.76 ± 0.29	16.56 ± 0.25	4483.7 ± 29.1	4540.1 ± 62.1	4548.2 ± 55.3
OpenAI GPT-4o	1.20 ± 0.03	0.91 ± 0.02	1.81 ± 0.03	–	–	–

Table 7: Timings (seconds, mean ± std) and peak GPU memory usage (MB, mean ± std) per generation call for each model over 10 runs on a fixed sample input. Platform: Tesla T4, Linux, CUDA 12.6, Torch 2.7.1. * - Model forced to float32.

> critique below to improve the definition. You may simplify the wording, explain in more detail, give an analogy, or provide an example - use whatever is most effective.
>
> Original definition: f‘{definition}’
>
> Difficult concept: f‘{difficult_concept}’
>
> Critique: f‘{critique}’
>
> Only return the rewritten definition and nothing else. Don’t explain the definition.

We utilised the following prompt for LLM-as-a-judge evalution of definitions:

> You are evaluating how well the rewritten definition of the term ‘{term}’ having the concept ‘{difficult_concept}’ elucidates the intended meaning of the original definition. Your goal is to judge how effectively it communicates the intended meaning to a layperson.
> Original definition (intended meaning): ‘{original_definition}’
> Rewritten Definition: ‘{rewritten_definition}’
> Rate the rewritten definition on these 4 criteria, from 1 to 5 (5 = excellent, 1 = poor):
> 1. Ease of Understanding - Is it easy for someone with no background to follow?
> 2. Precision - Is the explanation specific and correct? Avoid vagueness or technical jargon.
> 3. Conciseness - Is it clear and brief, without unnecessary elaboration?
> 4. Relevance - Does it stick closely to the original definition without adding unrelated ideas?
> Be strict. Carefully judge each dimension separately. Use the full scale. A perfect 5 is rare. A 3 is average. A 1 shows serious issues.
> Return ONLY a JSON object in this format:

```
{{
"ease_of_understanding": score,
"precision": score,
"conciseness": score,
"relevance": score,
}}
```

Model	Base	I1	I2
LlaMa 3.2 1B	3298	1958	1894
LlaMa 3.2 3B	3301	2971	2862
Gemma 3 1B	3263	3259	2781
Gemma 3 4B	3304	3201	3194
Qwen3 1.7B	3304	3203	3172
Qwen3 4B	3304	3226	3218
GPT-4o	3304	3247	3235

Table 8: Number of successful iterations per model and iteration type. Each cell gives the number of rewritten definitions that have a positive ΔFRE.

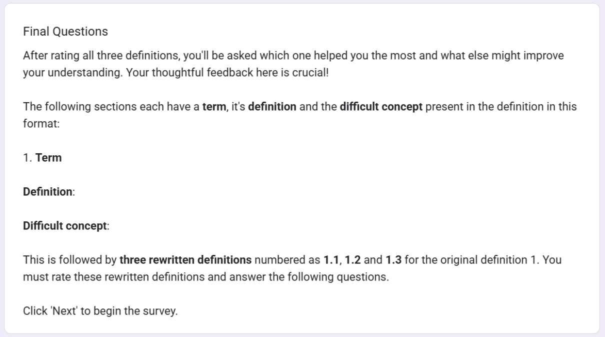
A.2 Human Evaluation

A.2.1 Setup

We recruited 18 non-expert annotators from the general population with diverse educational backgrounds (high school to PhD) to evaluate 50 randomly selected defintions each.

Our objective was to conduct the human evaluation survey on a larger number of people with varied information and accessibility needs, across various domains and levels of education. The participants would be tasked with rating the rewritten definitions on the criteria common to the "critic" in our methodology (Section 3). This allowed us to assess the quality of our generated definitions on a more holistic scale.

The annotators were instructed to read and rate each rewritten definition individually based on specific criteria using a 5-point Likert scale. As shown in Figure 2, we provided: 1) the term, 2) the original definition, 3) the *difficult concept*, and three rewritten definitions based on our methodology (Base, I1, and I2) for each definition.



Final Questions

After rating all three definitions, you'll be asked which one helped you the most and what else might improve your understanding. Your thoughtful feedback here is crucial!

The following sections each have a **term**, it's **definition** and the **difficult concept** present in the definition in this format:

1. **Term**

Definition:

Difficult concept:

This is followed by **three rewritten definitions** numbered as **1.1**, **1.2** and **1.3** for the original definition 1. You must rate these rewritten definitions and answer the following questions.

Click 'Next' to begin the survey.

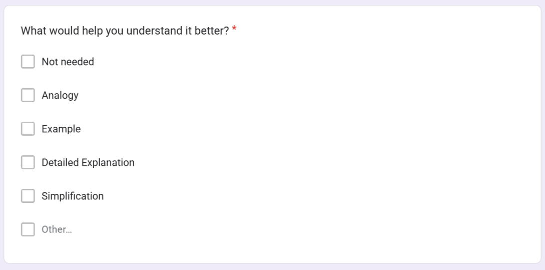
Figure 2: Screenshot of the skeleton of our human evaluation form instructing participants.

Additionally, we asked participants to choose the definition that helped them understand the *difficult concept* the most, and if it showed room for improvement. Figure 3 shows the options given to choose from which could possibly improve the generated definition (Asthana et al., 2024).

A.2.2 Rating Criteria

Each rating criteria chosen addresses a core aspect of how effective and accessible a rewritten definition is for non-expert readers.

The annotators were asked to rate each rewritten definition on a Likert scale as shown in Figure 4 They criteria are detailed below:



What would help you understand it better? *

□ Not needed

□ Analogy

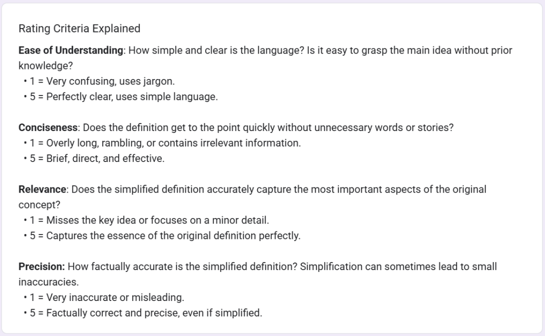
□ Example

□ Detailed Explanation

□ Simplification

□ Other…

Figure 3: Screenshot of the final question in our human evaluation form asking the reader if their favoured definition has scope for improvement. If yes, then the required options must be chosen facilitating the annotator's personalised information needs.



Rating Criteria Explained

Ease of Understanding: How simple and clear is the language? Is it easy to grasp the main idea without prior knowledge?
- 1 = Very confusing, uses jargon.
- 5 = Perfectly clear, uses simple language.

Conciseness: Does the definition get to the point quickly without unnecessary words or stories?
- 1 = Overly long, rambling, or contains irrelevant information.
- 5 = Brief, direct, and effective.

Relevance: Does the simplified definition accurately capture the most important aspects of the original concept?
- 1 = Misses the key idea or focuses on a minor detail.
- 5 = Captures the essence of the original definition perfectly.

Precision: How factually accurate is the simplified definition? Simplification can sometimes lead to small inaccuracies.
- 1 = Very inaccurate or misleading.
- 5 = Factually correct and precise, even if simplified.

Figure 4: Screenshot of description provided for the chosen rating criteria in our human evaluation form.

1. **Ease of Understanding**: Measures how comprehensible and cognitively clear the definition is for general non-expert readers without prior knowledge of the domain.*"Is it easy for someone with no background to follow?"*

2. **Conciseness**: Assesses whether the definition conveys the necessary information using as few words as possible while still retaining it's complete meaning. A concise definition is free from redundant phrases, irrelevant details, or overly lengthy explanations. *"Is it clear and brief, without unnecessary elaboration?"*

3. **Relevance**: Evaluates if the definition focuses on the key aspects of the term and its difficult concept, rather than introducing irrelevant or tangential information.*"Does it stick closely to the original definition without adding unrelated ideas?"*

4. **Precision**: Measures the accuracy and faithfulness of the definition to the underlying concept, ensuring that no critical details are omitted or incorrectly represented.*"Is the explanation specific and correct?"*

Rating Category	Group 1	Group 2	Mean Diff.	p-adj
Ease of Understanding	Bachelor’s	High School	1.10	$< .001$
	Bachelor’s	Master’s	1.05	$< .001$
	Bachelor’s	PhD	0.70	$< .001$
	High School	PhD	-0.40	$< .001$
	Master’s	PhD	-0.35	0.0001
Conciseness	Bachelor’s	High School	0.77	$< .001$
	Bachelor’s	Master’s	0.57	$< .001$
	Bachelor’s	PhD	0.41	$< .001$
	High School	Master’s	-0.21	0.004
	High School	PhD	-0.37	$< .001$
Relevance	Bachelor’s	High School	1.03	$< .001$
	Bachelor’s	Master’s	0.76	$< .001$
	Bachelor’s	PhD	0.40	$< .001$
	High School	Master’s	-0.26	0.0001
	High School	PhD	-0.63	$< .001$
	Master’s	PhD	-0.37	$< .001$
Precision	Bachelor’s	High School	0.83	$< .001$
	Bachelor’s	Master’s	0.89	$< .001$
	Bachelor’s	PhD	0.34	$< .001$
	High School	PhD	-0.49	$< .001$
	Master’s	PhD	-0.55	$< .001$

Table 9: Significant Pairwise Comparisons of Rating Scores by Education Level (Post-hoc Tukey HSD)

A.3 Examples

We provide some examples of rewritten definitions according to our methodology. We cover every model within the three SLM families we conduct our experimentation on as well as our baseline. They are listed as below:

1. **LLaMA family**: Llama3.2 1B, Llama3.2 3B
2. **Gemma family**: Gemma3 1B, Gemma3 4B
3. **Qwen family**: Qwen3 1.7B, Qwen3 4B
4. **Baseline model**: GPT-4o

Table 10 refers to example one where we provide the rewritten the definitions of the term "Chusquea", with *difficult concept* "genus" and it’s corresponding original definition (from WIKIDOMAINS dataset) as "Chusquea is a genus of evergreen bamboos in the grass family."

Table 11 refers to example one where we provide the rewritten the definitions of the term "Reserved word", with *difficult concept* "word" and it’s corresponding original definition (from WIKIDOMAINS dataset) as "In a computer language, a reserved word (also known as a reserved identifier) is a word that cannot be used as an identifier, such as the name of a variable, function, or label – it is "reserved from use"."

Model	**Base Definition**	**I1 Definition**	**I2 Definition**
Llama3.2 1B	Chusquea is a genus of plants that are characterized by their ability to grow in dense, evergreen forests, much like a dense, green carpet.	Chusquea is a genus of plants that are characterized by their ability to grow in dense, evergreen forests, much like a dense, green carpet.	Chusquea is a genus of evergreen bamboos that grow in thick, green forests.
Llama3.2 3B	Chusquea is a group of plants that are similar to bamboo, but are actually a type of grass. Think of it like a big family of grasses that are often mistaken for bamboo due to their similar appearance.	Chusquea is a genus of grasses that are often mistaken for bamboo due to their similar appearance.	Chusquea is a genus of bamboo, characterized by its unique growth patterns and characteristics that distinguish it from other bamboo species.
Gemma3 1B	Chusquea is a group of bamboo species belonging to the grass family. Think of it like a family of bamboo varieties – they share similar characteristics like their bamboo-like leaves and growth habits, but they're distinct from each other.	Rewritten Definition: Chusquea is a specialized group of bamboo species characterized by their distinctive, pleated leaves and a particular, slow growth pattern – they resemble a tightly woven, layered structure, much like a miniature, naturally formed bamboo forest.	Rewritten Definition: Chusquea is a specialized group of bamboo species renowned for its tightly arranged, layered leaves that create a remarkably dense, almost sculpted appearance, resembling a miniature, organically formed bamboo forest with a unique, slow growth rhythm.
Gemma3 4B	Chusquea is a group of closely related bamboo species – think of it like a family of bamboos. These bamboos, all part of the grass family, share many similarities in how they grow, look, and reproduce. They're recognized as a distinct "family" within the bamboo world, just like how a family is a group of people who are related and share common traits.	Chusquea is a *genus* – a specific grouping – within the grass family (Poaceae) that includes various bamboo species. Think of a genus like a club: it's a category that contains many different, but closely related, members. All species within the *Chusquea* genus share fundamental characteristics – like their growth patterns, stem structure, and method of reproduction – that distinguish them from other types of bamboo and grasses, even though they can appear quite diverse in size and appearance.	Chusquea is a group – a kind of family – within the large family of grasses (Poaceae), much like how a dog is a type of animal. All the different types of bamboo we call "Chusquea" are closely related because they share basic ways of growing and reproducing. They might look very different – some are tall and slender, others are short and bushy – but they all have shared features that set them apart from other grasses and bamboos.

Model	Base Definition	I1 Definition	I2 Definition
Qwen3 1.7B	Chusquea is a genus of evergreen bamboos in the grass family, which means they are a group of plants that are evergreen (they remain green throughout the year) and belong to the grass family. Think of it as a category or group of related plants that share certain characteristics. For example, like how a family of cats includes different breeds but shares common traits, Chusquea is a group of bamboos with similar features but different species.	Chusquea is a genus of evergreen bamboos in the grass family, characterized by their evergreen nature and placement in the grass family. It is a group of related plants that share certain traits, such as being evergreen and belonging to the grass family, but differ in specific species. Like a group of related cats, Chusquea includes different types of bamboos that share common features but vary in specific characteristics.	Chusquea is a genus of evergreen bamboos in the grass family, characterized by their evergreen nature and their placement in the grass family. It is a group of related plants that share common traits, such as being evergreen and belonging to the grass family, but differ in specific characteristics. Like a group of related plants, Chusquea includes different species of bamboos that are closely related but vary in specific features.
Qwen3 4B	Chusquea is a group of evergreen bamboo plants that are part of the grass family. A "genus" is like a category of plants that share common features, such as having leaves, roots, and the ability to grow tall. Chusquea is one such group, and all the plants in this group are types of bamboo that stay green all year and are part of the larger grass family. For example, just like how dogs, cats, and birds are all animals but belong to different groups, Chusquea is a group of bamboo plants that are closely related to each other.	Chusquea is a genus of evergreen bamboo plants that are part of the grass family. A genus is a category of plants that share common characteristics, like being grasses. All plants in the Chusquea genus are types of bamboo that remain green all year. For example, just as dogs and cats are both mammals but belong to different genera, Chusquea is a specific group of bamboo plants that are closely related to each other.	Chusquea is a genus of evergreen bamboo plants that are part of the grass family. A genus is a group of related species that share common features. Chusquea includes several species of bamboo that stay green all year. For example, like how different types of roses are all part of the same family but have different species, Chusquea is a group of closely related bamboo plants.
GPT 4o	Chusquea is a group of evergreen bamboos that belong to the grass family. Think of it like a family of plants that share similar characteristics, just like how all dogs belong to the same family of animals but come in different breeds.	Chusquea is a genus of evergreen bamboos that belong to the grass family, characterized by their similar traits and growth habits.	Chusquea is a group of evergreen bamboos that are part of the grass family, known for their similar characteristics.

Table 10: Term: Chusquea, *difficult concept*: genus, Original definition: Chusquea is a genus of evergreen bamboos in the grass family.

Model	Iteration 1 Definition	Iteration 2 Definition	Iteration 3 Definition
Llama3.2 1B	**Definition:** A reserved word is a special word in a programming language that cannot be used as a variable, function, or label, and is used to prevent errors and confusion in the code.	Think of a reserved word as a "name" that is not available for use. Just like how you can't use a name that's already taken by someone else, a reserved word is a name that's already been used by the computer, and it can't be used again.	Reserved word: A reserved word is a special term in programming that has a specific function and cannot be used as a name for variables or functions.
Llama3.2 3B	In simple terms, a reserved word is a special word in a computer language that has a specific job, like a title or a name, and cannot be used as a name for anything else, such as a variable, function, or label.	A reserved word is a predefined term in a programming language that has a specific function, such as a variable name, function name, or label, and cannot be used as a variable, function, or label in any other context.	A reserved word in a programming language is a term that cannot be used as a variable name, function name, or label in any context, due to its inherent meaning or association with the language itself.
Gemma3 1B	A reserved word is like a special name that computers don't allow you to use as a variable or a label – it's like saying "you can't call this variable as a name."	Rewritten Definition: Reserved words are special keywords or symbols in programming languages that the compiler or interpreter explicitly prohibits you from using as variable names or labels. They are carefully crafted to ensure code stability and prevent unintended variable names.	Reserved words are special keywords or symbols in programming languages that the compiler or interpreter explicitly prohibits you from using as variable names or labels. They are carefully crafted to ensure code stability and prevent unintended variable names.

Model	Iteration 1 Definition	Iteration 2 Definition	Iteration 3 Definition
Gemma3 4B	A reserved word is like a word in a dictionary that's permanently set aside for a special purpose. In a computer program, these are words that the computer already knows and uses for its own operations – think of them as built-in commands or instructions. You can't just grab those words and use them to name your own things (like a box to hold a number, or a tool to do a calculation). It's like trying to use the word "jump" to name a variable; the computer already knows "jump" means something else, so you can't reuse it for your own purposes. These words are	A reserved word is a name assigned to a word in a programming language that has a predefined meaning within the language itself. These words aren't available for you to use as names for your own variables, functions, or other identifiers. Think of them like pre-existing, official terms – like "true" or "false" – that the computer already understands and uses for specific tasks. You can't simply *claim* those words for your own; they're permanently set aside for the language's core operations.	A reserved word, also called a keyword, is a specific word in a programming language that has a special job already built into the language itself. It's like using a pre-defined tool – imagine a wrench. You wouldn't try to use a wrench as a screwdriver, would you? Similarly, reserved words like "if," "else," "while," or "for" aren't meant to be used as names for your own things (variables, functions, etc.). They're already taken and always represent something specific within the language's rules – they're essential parts of how the computer understands and executes your code.
Qwen3 1.7B	A reserved word in computer language is a special word that cannot be used as a variable, function, or label name. It is "reserved" because it cannot be reused by the programmer. For example, in programming languages like Python or Java, words like "if," "else," or "for" are reserved words and cannot be used as variable names. They are set aside for specific purposes, such as controlling program flow or defining loops.	A reserved word in computer language is a special word that cannot be used as a variable, function, or label name. It is reserved because these words are set aside for specific purposes, such as controlling program flow or defining loops. For example, in Python or Java, "if," "else," and "for" are reserved words that cannot be used as variable names. They are designated for specific tasks in the language.	A reserved word in computer language is a special word that cannot be used as a variable, function, or label name because it has a predefined purpose in the language. These words are set aside for specific tasks, such as controlling program flow or defining operations. For example, in Python or Java, "if," "else," and "for" are reserved words that cannot be used as variable names because they are designed to perform specific functions in the language.

Model	Iteration 1 Definition	Iteration 2 Definition	Iteration 3 Definition
Qwen3 4B	In a computer language, a reserved word is a special word that the computer uses for specific purposes, like telling the computer what to do. You can't use these words as names for variables, functions, or labels because they are "reserved" for the computer's special tasks, just like how certain words in a dictionary can't be used as regular words. For example, in programming, words like "if," "for," and "while" are reserved words because they tell the computer to make decisions or loop through actions.	In programming, a reserved word is a specific keyword that the computer uses for predefined tasks, such as controlling program flow or structuring code. These words cannot be used as variable names, function names, or labels because they are "reserved" for the computer's internal operations, just like certain words in a dictionary are not meant to be used as regular words. For example, "if," "for," and "while" are reserved words that instruct the computer to make decisions or repeat actions.	A reserved word in programming is a special word that the computer uses for specific tasks, like making decisions or looping. These words can't be used as names for variables, functions, or labels because they're set aside for the computer's built-in functions, just like certain words in a dictionary are not meant to be used as regular words. For example, words like "if," "for," and "while" are reserved words that tell the computer what to do.
GPT-4o	In a computer language, a reserved word is a special word that has a specific meaning and cannot be used for anything else, like naming a variable or function. Think of it like a "no parking" sign on a street; just as you can't park there because it's reserved for a specific purpose, you can't use reserved words for your own names because they are set aside for the language's rules.	In a computer language, a reserved word is a specific term that has a predefined function and cannot be used as an identifier, such as a variable or function name. It is like a designated space that is exclusively allocated for certain operations within the language, ensuring that the rules of the language are followed.	In a computer language, a reserved word is a specific term that has a fixed meaning and cannot be used as a name for variables or functions. These words are set aside for particular operations within the language, ensuring that the language's rules are consistently followed.

Table 11: Term: Reserved word, *difficult concept*: word, Original definition: In a computer language, a reserved word (also known as a reserved identifier) is a word that cannot be used as an identifier, such as the name of a variable, function, or label – it is "reserved from use".