ReadCtrl: Personalizing text generation with readability-controlled instruction learning

Hieu Tran [*1], **Zonghai Yao** [*1], **Lingxi Li** [1], **Hong Yu**[1,2]

[1] Manning College of Information and Computer Sciences, UMass Amherst, MA, USA
[2] Miner School of Computer and Information Sciences, UMass Lowell, MA, USA
`{hieutran, zonghaiyao}@umass.edu`

Abstract

Content generation conditioning on users' readability is an important application for personalization. In an era of large language models (LLMs), readability-controlled text generation based on LLMs has become increasingly important. This paper introduces a novel methodology called "Readability-Controlled Instruction Learning (ReadCtrl)," which aims to instruction-tune LLMs to tailor users' readability levels. Unlike the traditional methods, which primarily focused on categorical readability adjustments—typically classified as high, medium, and low or expert and layperson levels—with limited success, ReadCtrl introduces a dynamic framework that enables LLMs to generate content at various (near continuous level) complexity levels, thereby enhancing their versatility across different applications. Our results show that the ReadCtrl-Mistral-7b models significantly outperformed strong baseline models such as GPT-4 and Claude-3, with a win rate of 52.1%:35.7% against GPT-4 in human evaluations. Furthermore, Read-Ctrl has shown significant improvements in automatic evaluations, as evidenced by better readability metrics (e.g., FOG, FKGL) and generation quality metrics (e.g., BLEU, SARI, SummaC-Factuality, UniEval-Consistency and Coherence). These results underscore Read-Ctrl's effectiveness and tenacity in producing high-quality, contextually appropriate outputs that closely align with targeted readability levels, marking a significant advancement in personalized content generation using LLMs [1].

1 Introduction

Existing personalization methods mainly focus on the semantics of the content that individuals need, such as retrieving information based on individuals' search queries (Chen et al., 2023; Kirk et al., 2024;
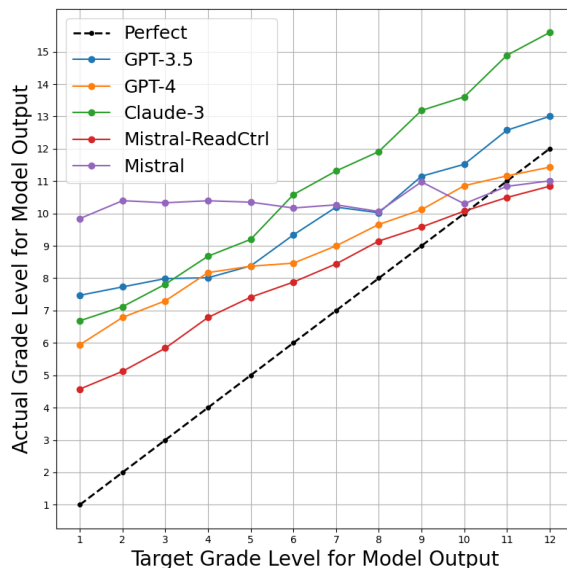

Figure 1: ReadCtrl instruction following ability. While current SOTA LLMs such as GPT and Claude (under the few-shot setting) show an upward trend in aligning their output with the target grade level, they fall significantly short of the 'perfect' adherence curve. Other weaker LLMs like Mistral-7b demonstrate little to no capacity to adjust to ReadCtrl instructions, as indicated by the flat line parallel to the x-axis. Notably, Mistral-ReadCtrl's performance closely approaches 'perfect', showcasing its advanced capability to tailor output to the specified readability level as set out by ReadCtrl instructions.

Shanahan et al., 2023) and summarization based on content representation (Richardson et al., 2023). However, one important aspect of personalization that has not been widely explored is readability-controlled content generation, which is particularly relevant for writers and readers with varying expertise levels(Vajjala, 2021). This involves tailoring content to match individuals' readability levels, which can vary widely due to differences in education, domain familiarity, and writing proficiency(Ribeiro et al., 2023). The emergence of large language models (LLMs) has further advanced this field, ushering in a transformative era of

---

* indicates equal contribution
[1]Our code and data will be released at `https://github.com/bio-nlp/ReadCtrl`

personalized writing assistance(Pu and Demberg, 2023). It is crucial for content generated by these models to be not only accurate and relevant but also adaptable to the stylistic and cognitive preferences of different users (e.g., writers and readers). The emphasis on customized content creation underscores the critical role of LLM alignment methods in digital interactions, especially in environments where users may require varying levels of guidance and editorial control(Zhou et al., 2023; Sun et al., 2023; Qin et al., 2024). At the heart of this innovative area are the principles of readability control instructions, designed to dynamically adapt the output vocabulary distribution to the specific context of each interaction. This can be achieved by analyzing chat history, interpreting user profiles, or responding to direct interaction requests, significantly enhancing LLMs' versatility (Huang et al., 2023).

Previous efforts in the domain of controllable text generation have primarily centered on binary readability adjustments, typically categorized into tasks of simplification, complication, or sibling style transfer (Goldsack et al., 2022; Guo et al., 2021; Luo et al., 2022). Despite their objectives, these approaches often fail to fully address the diverse personalization needs due to the limited variety in training data and a concentrated emphasis on readability. In traditional supervised fine-tuning scenarios, designing multiple readability-level ground truths for training data to facilitate readability control is not scalable. As a result, models may not have sufficient exposure to varied text complexities, limiting their ability to adjust content according to user-specific readability needs accurately. In response, the field of artificial intelligence is shifting towards more dynamic systems that can adapt outputs to meet users' unique preferences and requirements (Kirk et al., 2024). This shift is heralding a new era of personalized content creation that extends beyond standard domains like information retrieval to specialized areas, enhancing user engagement and satisfaction through tailored content.

This paper addresses these challenges by introducing a novel methodology termed "readability-controlled instruction learning (ReadCtrl)." Our findings demonstrate that ReadCtrl can empower LLMs to transform input text into content with controlled readability accurately. As illustrated in Figure 1, our investigation across a range of state-of-the-art LLMs shows varying degrees of compliance with readability-controlled instructions. Mainstream models like GPT (Achiam et al., 2023) and Claude (Anthropic, 2024), despite demonstrating an Upward trend, fall significantly short of the ideal 'perfect' adherence curve—they show a tendency towards compliance but lack precise output control. In contrast, models that previously struggled with readability control, such as Mistral-7b (Jiang et al., 2023)—illustrated almost as a horizontal line in the figure—have shown significant enhancement with the integration of ReadCtrl, such as Mistral-ReadCtrl. These models now nearly meet the ideal performance curve, showcasing their improved ability to customize outputs to specific readability demands. Specifically, ReadCtrl incorporates explicit instruction tuning based on readability and is rigorously tested through tasks designed to evaluate the model's ability to adjust output complexity. These tasks include text simplification, which aims to reduce the output's readability relative to the input; paraphrase generation, which maintains the input's readability; and semantic entailment generation, which adjusts the output's readability, potentially increasing or decreasing it in relation to the input. By employing a clear instruction—"*Given an input text, please output an entailment with a readability score around {target readability score}*"—models like Mistral-ReadCtrl demonstrate the efficacy of ReadCtrl in generating content that is not only semantically coherent but also finely adjusted to meet diverse comprehension needs and preferences.

In our initial experiments, we evaluated the model's performance in a "seen setting," where models were tested using datasets on which they were trained, such as ASSET (Alva-Manchego et al., 2020) for text simplification, PAWS (Zhang et al., 2019) for paraphrase generation, and SNLI (Bowman et al., 2015) for semantic entailment. This setting established a baseline for instruction tuning, confirming that the models could effectively adhere to readability instructions in familiar contexts. Subsequent experiments involved an "unseen setting," testing the models against new datasets they had not encountered during training, such as WikiSmall (Zhu et al., 2010) for text simplification, MRPC (Dolan and Brockett, 2005) for paraphrase generation, and MultiNLI (Williams et al., 2017) for semantic entailment. This phase was critical for assessing the models' adaptability

and accuracy in novel contexts, reflecting their generalizability and real-world applicability. We utilized several readability assessment metrics, including the Gunning Fog Index (Gunning, 1952) and Flesch-Kincaid Grade Level (Kincaid et al., 1975), to quantitatively evaluate the complexity of the generated texts. Additionally, we employed a range of automatic generation metrics for generation quality evaluation, such as BLEU (Papineni et al., 2002), SARI (Xu et al., 2016), Factuality (Laban et al., 2022), Consistency and Coherence (Zhong et al., 2022), to assess the quality of the generated texts, aiming to balance readability, information retention, factuality, consistency, and coherence during evaluation.

These evaluations confirmed the effectiveness of our ReadCtrl methodology across a diverse range of tasks and datasets. Particularly, Mistral-ReadCtrl excelled in both seen and unseen settings, showcasing robust performance metrics. For instance, in the unseen MRPC dataset, Mistral ReadCtrl achieved the lowest readability gap (1.66), the highest factuality (0.8184), and excellent BLEU (0.3798) and SARI (44.4327) scores, significantly outperforming competitors like GPT-4 and Claude-3. In the WikiSmall dataset, it recorded a readability gap of just 2.09, the highest coherence score (0.9763), and a strong SARI score of 42.1033. Furthermore, detailed human and LLM-as-a-judge (Lan et al., 2024) evaluations were conducted to compare Mistral-ReadCtrl with GPT-4 across different tasks and readability requirements. Mistral-ReadCtrl demonstrated superior performance, achieving a win rate of 52.1% in human evaluations and 58.3% in AI assessments, compared to GPT-4's 35.7% and 38.4%, respectively. Notably strong results were observed in tasks involving WikiSmall (62.5% in Human Eval, 67.7% in AI Eval) and SNLI (66.7% in Human Eval).

2 Methodology

2.1 Task Overview

Our methodology is designed to evaluate the effectiveness of instruction tuning conditional on readability across a suite of tasks, specifically focusing on text simplifications, paraphrase generation, and semantic entailment generation. These tasks are strategically chosen to test the model's capability in adjusting the complexity of its output to match specified readability levels. They serve a broad spectrum of applications, from enhancing educational material accessibility to refining technical documentation for diverse audiences.

- **Text Simplifications:** Here, the aim is to reduce the readability level of the given input text, making it more accessible to a wider audience or readers with varying comprehension skills. This task challenges the model to simplify complex text while preserving its essential content and meaning, demonstrating the ability to decrease textual complexity upon demand.

- **Paraphrase Generation:** In this task, the model is tasked with rewording the given text to produce a paraphrase that maintains the original's readability level. This requires a nuanced understanding of language to ensure the output remains true to the input's complexity and style, facilitating content reformulation without altering its accessibility.

- **Semantic Entailment Generation:** This involves creating text that semantically follows from the given input, with the flexibility to increase or decrease the readability level. The model must grasp the underlying meaning of the input text and generate output that logically entails the input, demonstrating versatility in producing content with adjustable complexity levels.

We employ the instruction tuning approach conditional on readability for all these tasks. This method provides explicit instructions to the model to control the output text's readability score, ensuring that the generated content aligns with the intended complexity level for the target audience. This approach underlines our belief that these tasks can all contribute to readability control generation, where, depending on the task—be it text simplification, paraphrase generation, or semantic entailment generation—the model is calibrated to generate output with the desired readability level. In text simplification, the goal is to lower the readability of the output relative to the input, while in paraphrase generation, the output's readability should mirror the input's. For the semantic entailment generation task, the output's readability may vary, being either higher or lower than the input's, thereby offering a versatile tool for adjusting text complexity across a wide range of contexts.

2.2 Instruction Design for Readability Control

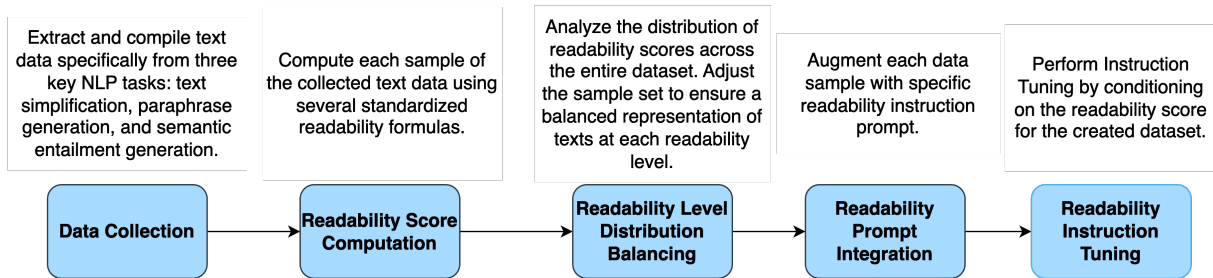To achieve the desired readability level across various tasks, we employ straightforward and singular

Figure 2: Overview of ReadCtrl data construction.

instruction. This approach emphasizes the model's ability to tailor its output to meet specific readability goals, demonstrating its versatility and effectiveness in readability control. The instruction is as follows:

"Given an input text, please output an entailment with a readability score around target readability score."

This concise instruction mandates the model to generate content that not only semantically follows from the given input but also aligns with a specified readability level, showcasing the model's capacity to produce targeted outputs that cater to diverse comprehension needs and preferences.

2.3 Implementation and Readability Scoring

The readability of the generated text is quantitatively evaluated using a suite of established readability metrics. We calculate the following readability scores [2]:

- **Gunning Fog Index:** Estimates the years of formal education required to understand the text on the first reading (Gunning, 1952).

- **Flesch-Kincaid Grade Level:** Translates the US grade level needed to comprehend the text (Kincaid et al., 1975).

- **Automated Readability Index:** Outputs a score correlating to the US grade level necessary for understanding (Senter and Smith, 1967).

- **Coleman-Liau Index:** Estimates the US grade level needed to comprehend the text using letter count instead of syllable count (Coleman and Liau, 1975).

These metrics are selected for their diverse approaches to assessing text complexity, offering a comprehensive understanding of the text's readability. Subsequently, an average Reading Grade Level (RGL) is derived from these scores to represent the text's overall readability. The integration of these

readability assessments into our methodology allows a nuanced approach to generating text that meets the specified readability criteria. By adjusting the instruction based on the target RGL, we can fine-tune the complexity of the output, making our approach adaptable to a wide range of applications, from educational content to technical documentation. This process underscores the importance of readability in tailoring content to specific audience needs, a critical factor in communication effectiveness across various domains.

3 Experiments

3.1 Dataset

Our experimental framework is designed to assess the model's performance across various tasks, specifically focusing on text simplification, paraphrase generation, and semantic entailment generation. To facilitate a comprehensive evaluation, we utilize six distinct datasets, two for each task, which enables us to explore the model's capabilities in both seen and unseen settings. The datasets employed in our experiments are outlined as follows:

- **Text Simplification:** For this task, we use the ASSET (Alva-Manchego et al., 2020) and WikiSmall (Zhu et al., 2010) datasets. ASSET is a diverse corpus for automatic sentence simplification, providing high-quality simplifications with multiple references per source sentence, making it ideal for instruction tuning and evaluation in seen settings. WikiSmall serves as an additional dataset for evaluating performance in an unseen setting, offering a different collection of simplified sentences derived from Wikipedia articles.

- **Paraphrase Generation:** We utilize the PAWS (Zhang et al., 2019) (Paraphrase Adversaries from Word Scrambling) and MRPC (Microsoft Research Paraphrase Corpus) (Dolan and Brockett, 2005) datasets. PAWS contains pairs of sentences paraphrasing each other, including those

---

[2]More details can be found in Appendix A.

constructed through controlled word scrambling, making it suitable for training and the seen setting evaluations. MRPC offers a collection of sentence pairs labeled as paraphrases or not, sourced from online news sources, to test the model's paraphrasing ability in unseen settings.

- **Semantic Entailment Generation:** For this task, the SNLI (Stanford Natural Language Inference) (Bowman et al., 2015) and MultiNLI (Multi-Genre Natural Language Inference) (Williams et al., 2017) datasets are employed. SNLI is a large collection of sentence pairs annotated with textual entailment information, used for instruction tuning and seen setting evaluation. MultiNLI extends this to a broader range of genres and contexts, providing a robust challenge for the model in unseen settings.

In our experimental setup, instruction tuning is performed on the training sets of ASSET, PAWS, and SNLI to align the model's output with specific readability goals. The effectiveness of this approach is then evaluated in two distinct settings: a *seen setting*, using the test sets of ASSET, PAWS, and SNLI, and an *unseen setting*, using the test sets of WikiSmall, MRPC, and MultiNLI. This methodology allows us to not only measure the model's immediate response to the instruction tuning but also its generalizability and adaptability to different textual contexts and tasks.

3.2 Evaluation Metrics

To comprehensively evaluate the model's performance across the different tasks, we employ a multifaceted set of metrics that assess various aspects of the generated texts. These metrics enable us to gauge the model's effectiveness in adjusting readability, maintaining factual accuracy, and ensuring textual coherence and consistency. The following metrics are used:

- **Average Readability Score:** This metric calculates the average readability level of the generated texts, providing insight into the overall accessibility of the content produced by the model.

- **Readability Gap (Delta):** The readability gap is measured as the difference between the requested readability level and the actual readability level of the generated text. This metric assesses the model's precision in hitting target readability levels.

- **Factuality:** Factuality is evaluated based on the methodology from the SummaC (Laban et al.,

2022) work, which offers a means to assess the factual alignment of the generated text with the source content or input.

- **Consistency and Coherence:** These aspects are measured using criteria from the UniEval (Zhong et al., 2022) framework, which provides standardized metrics for evaluating the logical consistency and coherence of the text, ensuring that the generated content is not only readable but also logically structured and coherent.

- **SARI:** The SARI (System output Against References and the Input sentence) (Xu et al., 2016) metric is utilized to assess the quality of text simplification. It measures the model's ability to produce simplified text that is both accurate and helpful, comparing the generated output against both the original text and reference simplifications.

- **BLEU:** The BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002) metric is applied to evaluate paraphrase generation and semantic entailment tasks. It quantifies the linguistic similarity between the generated texts and reference texts, indicating the model's capability to produce coherent and contextually appropriate content.

These metrics collectively offer a robust framework for assessing the nuanced performance of the model across various dimensions of text generation, readability adjustment, and content quality.

3.3 Evaluated Models

In our study, we evaluate a diverse set of models to understand their efficacy in handling tasks related to term definition generation, text simplification, and text complication, particularly focusing on adjusting text complexity according to specified readability levels. The models include:

- **GPT-3.5:** As a precursor to GPT-4, GPT-3.5 has demonstrated substantial capabilities in generating human-like text across various tasks. It serves as a baseline to understand the incremental improvements brought about by its successors and other models.

- **GPT-4:** The latest iteration from OpenAI's GPT series at the time of our study, GPT-4, represents a significant leap in language model performance, offering improved comprehension and generation capabilities over its predecessors.

- **Claude-3:** As a model known for its understanding and generation abilities, Claude-3 has been in-

Models	Readability Gap↓	Factuality↑	Consistency↑	Coherence↑	BLEU↑	SARI↑
ASSET (seen) | WikiSmall (unseen) - Text Simplification						
Claude-3	3.6323 | 4.53	0.5221 | 0.4612	0.9301 | 0.9391	0.934 | 0.9396	0.1874 | 0.1606	40.6964 | 32.9996
GPT-3.5	2.8635 | 3.12	0.7231 | 0.6721	0.9641 | 0.9401	0.9648 | 0.9231	0.2739 | 0.194	41.0061 | 33.9842
GPT-4	2.7465 | 2.69	0.6547 | 0.5892	**0.9688** | **0.9556**	**0.9687** | 0.949	0.2061 | 0.1666	39.7319 | 31.4657
Mistral-ReadCtrl	**1.8384** | **2.09**	**0.7687** | **0.7168**	0.9423 | 0.9477	0.9653 | **0.9763**	**0.4317** | **0.4321**	**49.3521** | **42.1033**
SNLI (seen) | MultiNLI (unseen) - Semantic Entailment Generation						
Claude-3	4.6433 | 5.64	0.5102 | 0.3904	0.919 | 0.8292	0.9331 | 0.8346	0.0446 | 0.0303	48.3281 | 44.4344
GPT-3.5	2.8333 | 6.7	0.5176 | 0.3967	0.9049 | 0.8829	0.8982 | 0.896	0.0875 | 0.0378	51.0201 | 44.0607
GPT-4	2.4733 | 3.36	0.5632 | 0.5167	0.9488 | **0.8961**	0.9382 | 0.8879	0.105 | 0.0562	**52.1153** | **46.4204**
Mistral-ReadCtrl	**1.8733** | **2.21**	**0.7406** | **0.6542**	**0.9491** | 0.8804	**0.9437** | **0.9122**	**0.183** | **0.1137**	51.6644 | 43.8289
PAWS (seen) | MRPC (unseen) - Paraphrase Generation						
Claude-3	2.4333 | 2.61	0.5141 | 0.4736	0.921 | 0.9154	0.9183 | 0.9012	0.2393 | 0.1679	38.3459 | 36.7783
GPT-3.5	1.5433 | 2.64	0.7443 | 0.5868	0.9761 | **0.9683**	0.9746 | 0.9679	0.3873 | 0.2059	37.9808 | 37.3417
GPT-4	1.4467 | 2.19	0.7085 | 0.5203	**0.979** | 0.9635	**0.978** | 0.9639	0.3122 | 0.153	34.3525 | 34.8477
Mistral-ReadCtrl	**0.6367** | **1.66**	**0.7871** | **0.8184**	0.9677 | 0.9669	0.9735 | **0.9769**	**0.6649** | **0.3798**	**60.5332** | **44.4327**

Table 1: Main results for seen | unseen tasks in ReadCtrl.

cluded as a baseline for its efficiency in handling various NLP tasks and its purported adaptability to instruction-based prompts, making it a relevant comparison for our instruction-tuned model.

- **Mistral 7B ReadCtrl:** Our proposed model has been instruction-tuned to adjust the readability level of generated texts based on explicit instructions. Mistral 7B is designed to excel in the specific tasks of text simplification, paraphrase generation, and semantic entailment generation, leveraging instruction tuning to achieve precise control over the readability of its outputs.

Each of these models brings unique strengths and capabilities to the table, allowing us to conduct a comprehensive comparison that not only highlights Mistral 7B's advancements in controlling readability but also situates these achievements within the broader context of current NLP technologies. By evaluating Mistral 7B against these established models, we aim to demonstrate its efficacy and potential applications in enhancing readability control in automatic text generation.

3.4 Results

3.4.1 Performance on seen tasks

Table 1 presents a performance comparison of Claude-3, GPT-3.5, GPT-4, and our model, Mistral-ReadCtrl, on seen tasks involving three datasets where instruction tuning was implemented: AS-SET, SNLI, and PAWS. Regarding the Readability Gap, Mistral-ReadCtrl demonstrates superior precision in adhering to target readability levels, achieving the lowest scores across all datasets, indicating effective control over text readability. Factuality scores, which assess the accuracy of content com-

pared to the original, show that Mistral-ReadCtrl maintains higher factual consistency than its counterparts. When evaluating Consistency and Coherence, which measure the logical flow and structural soundness of texts, Mistral-ReadCtrl performs robustly, though it is slightly outperformed by GPT-4 in the PAWS dataset. For BLEU and SARI metrics, critical for evaluating the linguistic and contextual appropriateness of text simplification and paraphrase generation, Mistral-ReadCtrl again posts the highest scores, showcasing its efficacy in producing high-quality, contextually appropriate responses.

3.4.2 Performance on unseen tasks

Table 1 illustrates the performance of four models — Claude-3, GPT-3.5, GPT-4, and Mistral-ReadCtrl — on unseen tasks, using the datasets WikiSmall for text simplification, MultiNLI for semantic entailment generation, and MRPC for paraphrase generation. These results are crucial for assessing each model's ability to generalize beyond the data types encountered during training.

In the WikiSmall dataset, Mistral-ReadCtrl outperforms other models with the lowest readability gap of 2.09, suggesting superior control aligning with the target readability levels. It also achieves the highest factuality and coherence scores and significantly outstrips the competition in BLEU and SARI scores, indicating its effectiveness in maintaining content quality in text simplification tasks.

Mistral-ReadCtrl again shows notable performance for the MultiNLI dataset, which focuses on semantic entailment generation, with the lowest readability gap of 2.21 and the highest factuality and coherence scores among the models. However, while its BLEU score is the highest, it slightly trails
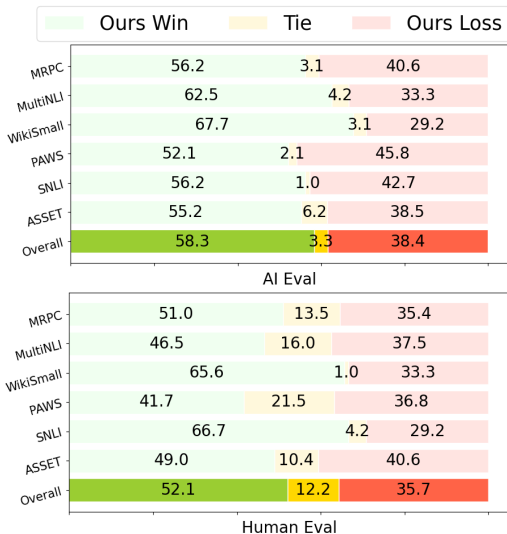
Figure 3: Win rate (%) for Mistral-ReadCtril vs GPT-4 (3 shots) using AI (Claude3 and GPT3.5) and Human evaluation.

Grade 2	
Employ short, straightforward sentence structures	100%
Focus only on essential details, omitting unnecessary complexity	85.7%
Use very simple vocabulary and avoid complex words	76.2%
Break down information into clear sequential steps	35.7%
Grade 5	
Introduce some more varied and content-specific vocabulary	71.4%
Use longer sentences with conjunctions to combine ideas	57.1%
Provide additional context and relevant details	28.6%
Explain concepts more directly instead of narratives	23.8%
Grade 8	
Use complex sentence structures like passive voice	66.7%
Employ richer descriptive language and vivid details	54.8%
Incorporate academic and technical terminology	47.6%
Establish clear logical connections between ideas	21.4%
Grade 11	
Construct elaborate compound-complex sentences	42.9%
Use sophisticated vocabulary from all domains	40.5%
Write with consistent formality and academic tone	33.3%
Employ advanced stylistic techniques like figurative language	23.8%

Table 2: Readability control strategies for Mistral 7B ReadCtrl. The number represents what proportion of the system output in the corresponding grade level uses the corresponding method to adjust the readability.

behind GPT-4 in SARI, demonstrating strong but not leading performance in generating entailments that are semantically aligned with the input.

In the MRPC dataset, which tests the model's ability to generate paraphrases, Mistral-ReadCtrl leads to a readability gap of 1.66, the highest factuality and coherence scores, and the best BLEU and SARI scores. This highlights its exceptional ability to generate paraphrases that not only adhere closely to the desired readability level but also maintain high levels of linguistic and contextual integrity.

Overall, the data from the unseen tasks confirm Mistral-ReadCtrl's robust generalization capabilities across different types of text generation tasks, demonstrating its potential as a versatile tool in NLP applications where adapting to varied content types and maintaining consistent quality is critical.

4 Human Evaluation

4.1 Human Evaluation settings

Our human evaluation was conducted by 5 human evaluators and 1 expert evaluator [3]. We randomly sampled 6 data from the test datasets of 6 data sets, and a total of 36 data appeared in the human evaluation. We give detailed instructions to the annotators: "*You are evaluating two systems, both of which are trying to convert inputs to specific readability requirements to produce output suitable for the user. I will show you the input and output of the two systems on grade 2/5/8/11, respectively. Tell me which system's output you prefer by specifying system 1 or system 2 or tie if the quality is the same. Please explain the reason for your preference.*". And they worked using our evaluation system to select preference; see Figure 4 (left). Each time, we randomly shuffle the outputs of two systems (Mistral-ReadCtrl and GPT-4), and they can choose the one that better meets the readability requirements and has higher output quality. If they think the outputs of the two systems are tied, they can choose both. After we get judgments from multiple people per instance, we do not aggregate their labels before calculating the win rate but count them individually. We also used a similar setting of our human preference evaluation for AI evaluation with claude-3-opus-20240229 and gpt-3.5-turbo-0125 as the judge [4].

After preference evaluation, we then worked with one Linguistics expert for the readability control strategies annotation. We summarized 4 different reasons for each grade level (see Table 2) and then asked the expert to use our evaluation system for readability control strategies annotation; see Figure 4 (right). Each time, the expert needed to select all qualified control strategies for the output of our system (Mistral 7B ReadCtrl), where multiple selections are allowed.

---

[3]More details can be found on Appendix B.

[4]More details can be found on Appendix C

4.2 Human Evaluation Results

As shown in Figure 3, the human evaluation prefers Mistral-ReadCtrl with an overall win rate of 49.4% as opposed to GPT-4, while AI evaluation gave us a broader win rate of 58.3%. Specifically, both seen settings (ASSET, SNLI, PAWS) and unseen settings (WikiSmall, MultiNLI, MRPC) exhibit consistent results across human evaluation and AI evaluation. Among these, the lead in WikiSmall and SNLI is most pronounced. Delving further, human annotations shed light on the operational tactics of Mistral-ReadCtrl. For example, when catering to Grade 2 readability, it implemented straightforward sentence structures with 100% adherence, focused on essential details 85.7% of the time, and employed very simple vocabulary in 76.2% of instances. For more advanced Grade 5 and 8 requirements, it adeptly introduced content-specific vocabulary (71.4% for Grade 5) and complex sentence structures (66.7% for Grade 8), illustrating the model's dexterity in scaling complexity according to the readability demands.

5 Related Work

Early efforts for readability control in natural language generation (NLG) included microplanning steps to tailor the text to match different target reading levels (Moraes et al., 2016; Agrawal and Carpuat, 2019; Marchisio et al., 2019). More recent studies, such as those by Luo et al. (2022), have investigated controllable abstractive and extractive approaches for generating summaries tailored for layman and expert audiences from biomedical documents. Concurrently, recent work Pu and Demberg (2023); Rao and Tetreault (2018); Yao and Yu (2021) examined the ability of the language models to adapt its outputs to different target audiences and writing styles, ranging from formal to informal, whereas Imperial (2022) highlighted that GPT2 models struggle with preserving the linguistic complexity of input prompts. Significant developments have also been made in models for Plain Language Summarization (PLS) from scientific papers (Devaraj et al., 2021; Goldsack et al., 2023; Guo et al., 2023), focusing on generating simplified texts that retain the original content's meaning.

Text Simplification aims to enhance the readability of sentences by reducing their linguistic complexity, with various important societal applications, such as increasing accessibility for those with cognitive disabilities and also for patient education, non-native speakers, and children with reading difficulties (Martin et al., 2020; Cao et al., 2020). Various aspects of simplified outputs have been addressed, including adapting to specific levels (Nishihara et al., 2019), incorporating edit operations (Kumar et al., 2020; Mallinson et al., 2020), enforcing lexical and syntactic constraints (Martin et al., 2019), applying linguistically motivated syntactic rules (Maddela et al., 2020), and integrating complex span extraction and lay language generation (Chen et al., 2018; Kwon et al., 2022; Jiang and Xu, 2024; Yao et al., 2023) into the simplification process. In contrast to traditional text simplification, which primarily focuses on controlling the extent of paraphrasing, our approaches are designed to produce succinct and informative output for various tasks in both seen and unseen settings, while maintaining different fine-grained levels of desired readability.

Finally, our work follows the instruction tuning technique (Zhang et al., 2023a) to help LLMs learn to follow readability-controlled instructions. Traditional supervised fine-tuning (SFT) techniques often struggle with fine-grained readability control, as they depend on manual annotation or synthetic data generation for enriching readability data, followed by model fine-tuning. This method requires considerable financial and time resources, with repeated tasks for each domain-specific application. Alternatively, recent advances in instruction learning offer a more generalized approach, as highlighted in several studies (Wei et al., 2021; Wang et al., 2022; Honovich et al., 2022; Zhang et al., 2023b; Tran et al., 2023). Instruction learning operates on the premise that the model already possesses the necessary knowledge and skills to perform the target task but requires instructional data to learn how to follow the instructions effectively. By using a FLAN-style Instruction Fine-Tuning method (Wei et al., 2021), we gathered task-specific instructions for ReadCtrl and conducted fine-tuning. Our evaluations, both automatic and human, on seen and unseen tasks, confirm ReadCtrl's effectiveness, simplifying the adaptation process and broadening application scope with minimal data needs.

6 Conclusion

The ReadCtrl approach enhances the adaptability of LLMs by dynamically adjusting content complex-

ity to bridge the gap between writers and readers with diverse expertise. By outperforming mainstream models like GPT-4 in evaluations, Mistral-ReadCtrl demonstrates its capability to generate nuanced, high-quality outputs, highlighting its potential in enabling personalized writing and reading experiences.

7 Limitations

In this paper, we propose a new instruction-learning approach to enhance the controllability of readability levels. While this adjustment is not specific to any particular language, we conducted all of our experiments and analysis exclusively on English-language summarization datasets. Additionally, due to the resource limitation, our analysis was limited to Text Simplification (ASSET and Wiki-iSmall datasets), Paraphrase Generation (PAWS and MRPC datasets), and Semantic Entailment Generation (SNLI and MultiNLI datasets), reflecting their prevalent application in NLG studies. Consequently, this paper does not extend to exploring style variations in non-English and other relevant tasks and datasets, such as some mentioned text-to-text generation datasets in the tutorial at ACL 2024 (Dou et al., 2023). Thus, the scope of this study is confined, and the results may not be universally applicable across different linguistic and stylistic contexts. For readability evaluation, studies have shown that readability formulas may not be ideal for medical text (Zheng and Yu, 2017) because short texts (e.g., abbreviations and fragmented texts rather than complete sentences) are common in EHR notes. In future work, we may explore machine-learning-based approaches (Zheng et al., 2018) for readability in subdomains. Finally, due to resource constraints, we were unable to have actual grade 2, 5, 8, and 11 students provide pairwise preference feedback during our human evaluation. In the future, we plan to collect human evaluation feedback from more appropriate target groups to enhance the reliability of our results further.

8 Ethics Statement

While Mistral-ReadCtrl has demonstrated a high level of readability controllability on several NLG datasets dataset, this does not imply their use as general controllable interactive models (like some chatbot systems). These models should be thoroughly evaluated before being used in different settings to ensure reliability.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Sweta Agrawal and Marine Carpuat. 2019. Controlling text complexity in neural machine translation. *arXiv preprint arXiv:1911.00835*.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. Asset: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. *arXiv preprint arXiv:2005.00481*.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Yixin Cao, Ruihao Shui, Liangming Pan, Min-Yen Kan, Zhiyuan Liu, and Tat-Seng Chua. 2020. Expertise style transfer: A new task towards better communication between experts and laymen. *arXiv preprint arXiv:2005.00701*.

Jin Chen, Zheng Liu, Xunpeng Huang, Chenwang Wu, Qi Liu, Gangwei Jiang, Yuanhao Pu, Yuxuan Lei, Xiaolong Chen, Xingmei Wang, Defu Lian, and Enhong Chen. 2023. When large language models meet personalization: Perspectives of challenges and opportunities. *ArXiv*, abs/2307.16376.

Jinying Chen, Emily Druhl, Balaji Polepalli Ramesh, Thomas K Houston, Cynthia A Brandt, Donna M Zulman, Varsha G Vimalananda, Samir Malkani, and Hong Yu. 2018. A natural language processing system that links medical terms in electronic health record notes to lay definitions: system development using physician reviews. *Journal of medical Internet research*, 20(1):e26.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Ashwin Devaraj, Byron C Wallace, Iain J Marshall, and Junyi Jessy Li. 2021. Paragraph-level simplification of medical texts. In *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2021, page 4972. NIH Public Access.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third international workshop on paraphrasing (IWP2005)*.

Yao Dou, Philippe Laban, Claire Gardent, and Wei Xu. 2023. Automatic and human-ai interactive text generation. *arXiv preprint arXiv:2310.03878*.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. *arXiv preprint arXiv:2210.09932*.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2023. Domain-driven and discourse-guided scientific summarisation. In *European Conference on Information Retrieval*, pages 361–376. Springer.

Robert Gunning. 1952. The technique of clear writing. *(No Title)*.

Yue Guo, Tal August, Gondy Leroy, Trevor Cohen, and Lucy Lu Wang. 2023. Appls: A meta-evaluation testbed for plain language summarization. *arXiv preprint arXiv:2305.14341*.

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021. Automated lay language summarization of biomedical scientific reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 160–168.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2022. Unnatural instructions: Tuning language models with (almost) no human labor. *arXiv preprint arXiv:2212.09689*.

Xu Huang, Jianxun Lian, Yuxuan Lei, Jing Yao, Defu Lian, and Xing Xie. 2023. Recommender ai agent: Integrating large language models for interactive recommendations. *arXiv preprint arXiv:2308.16505*.

Joseph Marvin Imperial. 2022. Uniform complexity for text generation. *arXiv preprint arXiv:2204.05185*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Chao Jiang and Wei Xu. 2024. Medreadme: A systematic study for fine-grained sentence readability in medical domain. *arXiv preprint arXiv:2405.02144*.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. 2024. The benefits, risks and bounds of personalizing the alignment of large language models to individuals. *Nature Machine Intelligence*, pages 1–10.

Dhruv Kumar, Lili Mou, Lukasz Golab, and Olga Vechtomova. 2020. Iterative edit-based unsupervised sentence simplification. *arXiv preprint arXiv:2006.09639*.

Sunjae Kwon, Zonghai Yao, Harmon S Jordan, David A Levy, Brian Corner, and Hong Yu. 2022. Medjex: A medical jargon extraction model with wiki's hyperlink span and contextualized masked language model score. *arXiv preprint arXiv:2210.05875*.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Tian Lan, Wenwei Zhang, Chen Xu, Heyan Huang, Dahua Lin, Kai Chen, and Xian-ling Mao. 2024. Criticbench: Evaluating large language models as critic. *arXiv preprint arXiv:2402.13764*.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2022. Readability controllable biomedical document summarization. *arXiv preprint arXiv:2210.04705*.

Mounica Maddela, Fernando Alva-Manchego, and Wei Xu. 2020. Controllable text simplification with explicit paraphrasing. *arXiv preprint arXiv:2010.11004*.

Jonathan Mallinson, Aliaksei Severyn, Eric Malmi, and Guillermo Garrido. 2020. Felix: Flexible text editing through tagging and insertion. *arXiv preprint arXiv:2003.10687*.

Kelly Marchisio, Jialiang Guo, Cheng-I Lai, and Philipp Koehn. 2019. Controlling the reading level of machine translation output. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 193–203.

Louis Martin, Angela Fan, Éric De La Clergerie, Antoine Bordes, and Benoît Sagot. 2020. Muss: Multilingual unsupervised sentence simplification by mining paraphrases. *arXiv preprint arXiv:2005.00352*.

Louis Martin, Benoît Sagot, Eric de la Clergerie, and Antoine Bordes. 2019. Controllable sentence simplification. *arXiv preprint arXiv:1910.02677*.

Priscilla Moraes, Kathleen F McCoy, and Sandra Carberry. 2016. Enabling text readability awareness during the micro planning phase of nlg applications. In *Proceedings of the 9th International Natural Language Generation conference*, pages 121–131.

Daiki Nishihara, Tomoyuki Kajiwara, and Yuki Arase. 2019. Controllable text simplification with lexical constraint loss. In *Proceedings of the 57th annual meeting of the association for computational linguistics: Student research workshop*, pages 260–266.

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.

Dongqi Pu and Vera Demberg. 2023. Chatgpt vs human-authored text: Insights into controllable text summarization and sentence style transfer. *arXiv preprint arXiv:2306.07799*.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. Infobench: Evaluating instruction following ability in large language models. *arXiv preprint arXiv:2401.03601*.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. *arXiv preprint arXiv:1803.06535*.

Leonardo FR Ribeiro, Mohit Bansal, and Markus Dreyer. 2023. Generating summaries with controllable readability levels. *arXiv preprint arXiv:2310.10623*.

Chris Richardson, Yao Zhang, Kellen Gillespie, Sudipta Kar, Arshdeep Singh, Zeynab Raeesy, Omar Zia Khan, and Abhinav Sethy. 2023. Integrating summarization and retrieval for enhanced personalization via large language models. *arXiv preprint arXiv:2310.20081*.

RJ Senter and Edgar A Smith. 1967. Automated readability index. Technical report, Technical report, DTIC document.

Murray Shanahan, Kyle McDonell, and Laria Reynolds. 2023. Role play with large language models. *Nature*, pages 1–6.

Jiao Sun, Yufei Tian, Wangchunshu Zhou, Nan Xu, Qian Hu, Rahul Gupta, John Frederick Wieting, Nanyun Peng, and Xuezhe Ma. 2023. Evaluating large language models on controlled generation tasks. *arXiv preprint arXiv:2310.14542*.

Hieu Tran, Zhichao Yang, Zonghai Yao, and Hong Yu. 2023. Bioinstruct: Instruction tuning of large language models for biomedical natural language processing. *arXiv preprint arXiv:2310.19975*.

Sowmya Vajjala. 2021. Trends, limitations and open challenges in automatic readability assessment research. *arXiv preprint arXiv:2105.00973*.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

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

Zonghai Yao, Nandyala Siddharth Kantu, Guanghao Wei, Hieu Tran, Zhangqi Duan, Sunjae Kwon, Zhichao Yang, Hong Yu, et al. 2023. Readme: Bridging medical jargon and lay understanding for patient education through data-centric nlp. *arXiv preprint arXiv:2312.15561*.

Zonghai Yao and Hong Yu. 2021. Improving formality style transfer with context-aware rule injection. *arXiv preprint arXiv:2106.00210*.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. 2023. Evaluating large language models at evaluating instruction following. *arXiv preprint arXiv:2310.07641*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023a. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Xinlu Zhang, Chenxin Tian, Xianjun Yang, Lichang Chen, Zekun Li, and Linda Ruth Petzold. 2023b. Alpacare: Instruction-tuned large language models for medical application. *arXiv preprint arXiv:2310.14558*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*.

Jiaping Zheng and Hong Yu. 2017. Readability formulas and user perceptions of electronic health records difficulty: a corpus study. *Journal of medical Internet research*, 19(3):e59.

Jiaping Zheng, Hong Yu, et al. 2018. Assessing the readability of medical documents: a ranking approach. *JMIR medical informatics*, 6(1):e8611.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. *arXiv preprint arXiv:2210.07197*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361.

A   Readability Metrics

Developed by Kincaid et al. (1975), the **Flesch-Kincaid Grade Level(FKGL)** score is a metric that assigns higher scores to texts that are easier to read. It is calculated using the formula:

$$\text{FKGL} = 206.835 - 1.015 \left(\frac{\text{totalWords}}{\text{totalSentences}} \right) - 84.6 \left(\frac{\text{totalSyllables}}{\text{totalWords}} \right)$$

The **Gunning Fog Index (GFI)**, proposed by Gunning (1952), quantifies the level of formal education required to comprehend a text upon first reading. It is computed as:

$$GFI = 0.4 \left(\frac{\text{totalWords}}{\text{totalSentences}} + 100 \frac{\text{longWords}}{\text{totalWords}} \right)$$

where *longWords* are defined as words containing more than seven characters. Higher values indicate lower readability.

The **Automated Readability Index (ARI)**, developed by Senter and Smith (1967), correlates to the U.S. school grade level needed to understand the text. It uses the formula:

$$ARI = 4.71 \left(\frac{\text{totalCharacters}}{\text{totalWords}} \right) + 0.5 \left(\frac{\text{totalWords}}{\text{totalSentences}} \right) - 21.43$$

Developed by Coleman and Liau (1975), the **Coleman-Liau Index (CLI)** focuses on characters rather than syllables to assess text readability. The formula for CLI is:

$$CLI = 0.0588L - 0.296S - 15.8$$

where L is the average number of letters per 100 words, and S is the average number of sentences per 100 words. This metric provides an estimate of the grade level required to understand the text.

B   More details about human evaluation

We provide additional details on our human evaluation setup. Human preference annotators are 5 students who have completed a bachelor's degree or above from an American university and are fluent in English. We add some tasks with known answers (i.e., cases where the most/least readable and good/bad quality text should be clear), enabling us to estimate the accuracy of annotators who work on these. Annotators with low accuracy on tasks with known answers are automatically removed from our worker pool. Only the annotators who passed these final tests were accepted to work on the human preference in this paper. We gave annotators fair compensation (20$/hrs).
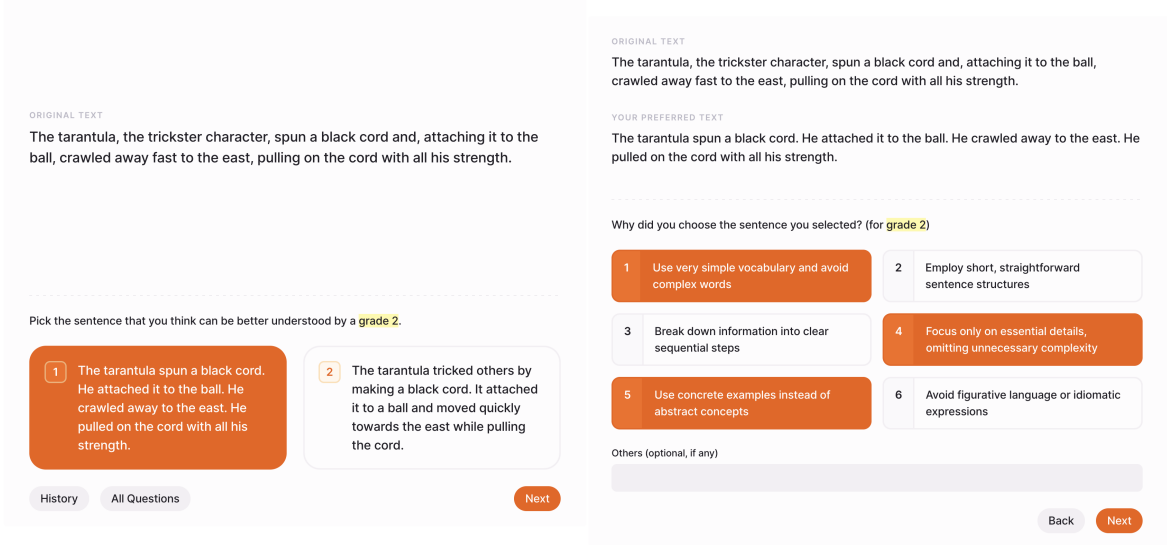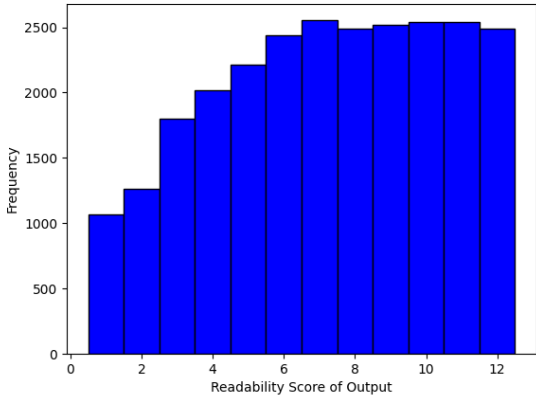
Figure 4: Screenshot of the human evaluation.

Figure 5: Distribution of examples readability scores from instruction tuning datasets

Parameter	Value
Computing Infrastructure	40GB NVIDIA A100 GPU
Optimizer	Adam
Optimizer Params	$\beta = (0.9, 0.999), \epsilon = 10^{-8}$
Learning rate	3×10^{-4}
Learning Rate Decay	Linear
Weight Decay	0
Warmup Steps	200
Batch size	128
Epoch	5

Table 3: Hyperparameter settings for Mistral 7B Read-Ctrl.

C More details about LLM evaluation

To reduce the heavy human evaluation and make the evaluation easier to reproduce, we use a similar setting of our human preference evaluation for AI evaluation. Comparison-based feedback evaluation assesses the accuracy of LLM in deciding preferences between two responses. However, it is widely acknowledged that current LLMs exhibit significant **positional bias** (Lan et al., 2024; Wang et al., 2023; Zheng et al., 2024; Zeng et al., 2023), i.e., LLMs tend to prefer responses based on their specific position in the prompt. We implement a rigorous verification process to mitigate the effects of positional bias to evaluate the real capability. Specifically, given responses R_a and R_b to be compared, we obtain the comparison based on two orders, noted as $F_a^c = F_c(R_a, R_b)$ and

$F_b^c = F_c(R_b, R_a)$. The objective scores are computed by:

$$s = \frac{1}{N} \sum_{i=1}^{N} 1(L(F_{a,i}^c, F_{b,i}^c))$$

where $L(F_a^c, F_b^c)$ is true if and only if $F_a^c \neq F_b^c$ and F_a^c, F_b^c align with ground-truth preference label. N is the number of test samples. The prompts we used for LLM-as-a-judge (claude-3-opus-20240229 and gpt-3.5-turbo-0125) evaluation can be found in Table 4.

D Hyper-parameter Settings

The experiments were executed using the version 4.37.1 of the transformers library released by Hugging Face. In Table 3, we report the hyperparameters used to train the models on our combined dataset. We use the Adam optimizer and employ

a linearly decreasing learning rate schedule with warm-up step is 200. In this section, we detail our experimental setup, the datasets employed, and the evaluation strategy adopted for assessing the performance of our instruction-tuned LLMs in various BioNLP tasks. Furthermore, all experiments were conducted using two Nvidia A100 GPUs, each with 40 GB of memory. The CPU used was an Intel Xeon Gold 6230 processor, and the system was equipped with 192 GB of RAM.

E   Experiments with GPT3.5, GPT4, Claude-3

All of our experiments were conducted on the version of GPT3.5, GPT4 and Claude 3 between 25 March 2023 and 13 April 2024 by using the OpenAI's API.10 We set temperature = 1, top_p=1, frequency penalty = 0, and presence penalty = 0.

F   ReadCtrl instruction following evaluation setting

We have plotted Figures 1, 6, 7, 8, 9, 10, and 11 by calculating the readability scores or reading levels of the outputs generated in response to prompts that request specific reading levels ranging from 1 to 12. These calculations were performed across all test sets of the six datasets mentioned in the Experiment section. Additionally, we calculated the standard deviation of the readability scores across these test sets to assess the consistency of the output's readability.

G   Examples of output generated by Mistral-ReadCtrl and GPT4 during ReadCtrl instruction following evaluation

Tables 4 present distinct levels of output generated by the Mistral-ReadCtrl and GPT4 and their readability scores given by Flesch-Kincaid Grade Level (FKG), Gunning fog index (GFI), and Coleman-Liau index (CLI) metrics.

We will delve into the observed discrepancies between the Readability Gap and the performance curves in our evaluation, as demonstrated by our results for the PAWS and MultiNLI datasets. The Readability Gap, calculated as the average difference between the actual readability score of the output and the requested readability score across all samples, shows intriguing variations in behavior across different datasets.

For the PAWS dataset, although the Readability Gap appears almost perfect in Table 1, the corresponding curve does not exhibit as favorable performance. This anomaly can be attributed to the output readability distribution of PAWS, which is somewhat concentrated within a specific range (typically between 4-8). While this concentration allows for excellent performance within this median range, it leads to a less generalized performance across the full spectrum of readability levels (from 1-12). Therefore, even a small Readability Gap in numerical terms may not accurately reflect an evenly distributed ability to target all requested readability levels.

Conversely, the MultiNLI dataset exhibits a larger Readability Gap in Table 1, yet the performance curve approaches perfection. This suggests that while the average gap is larger, the outputs are more uniformly distributed across the entire range of readability levels, allowing for closer adherence to the target levels across a broader spectrum. This indicates a more generalized and adaptable performance despite the numerically larger gap.

This analysis underscores the importance of considering both the Readability Gap and the distribution of output readability scores when assessing model performance. A low Readability Gap might suggest excellent average performance but could conceal poor adaptability across a range of readability levels. Conversely, a higher Readability Gap might indicate a more uniform distribution of performance across all levels, suggesting a different kind of effectiveness.

Further investigation into these patterns for all six datasets employed in our study reveals similar trends.
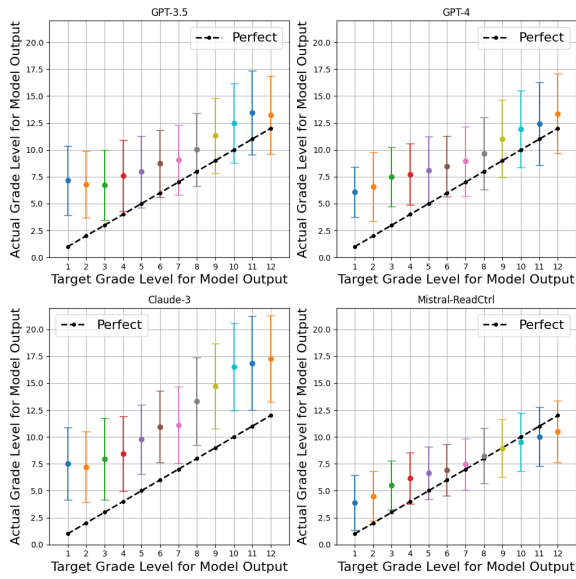
Figure 6: ReadCtrl instruction following ability evaluation on ASSET.
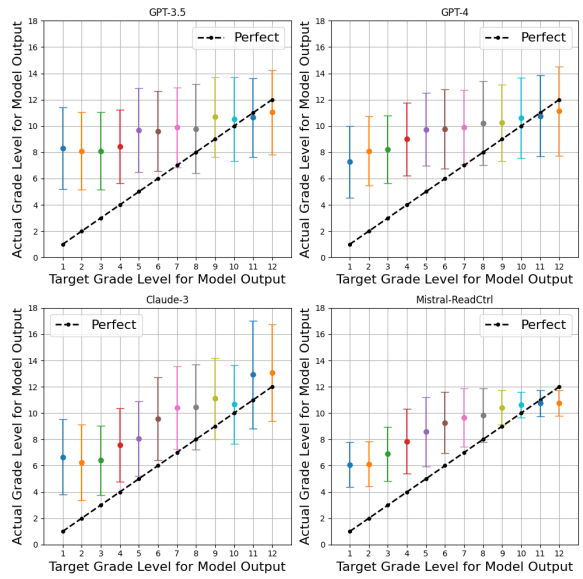

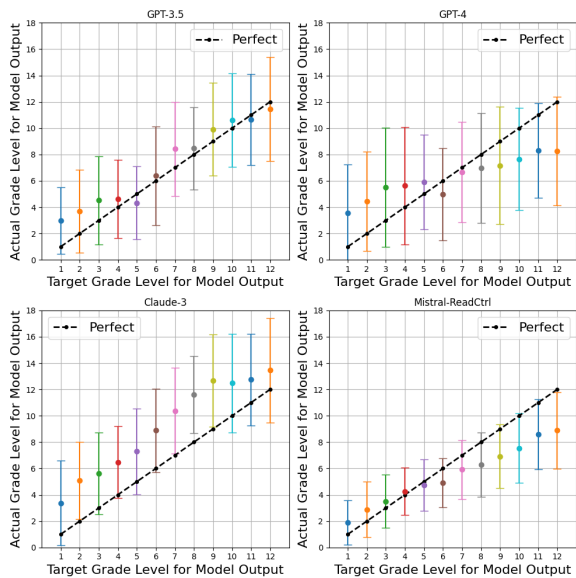Figure 8: ReadCtrl instruction following ability evaluation on PAWS

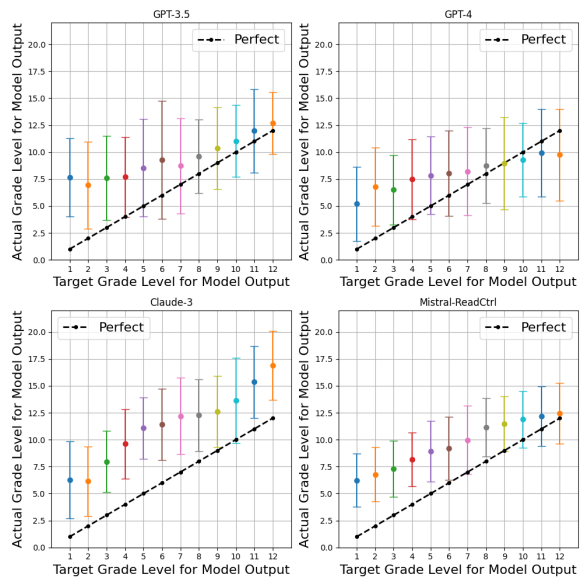Figure 7: ReadCtrl instruction following ability evaluation on SNLI

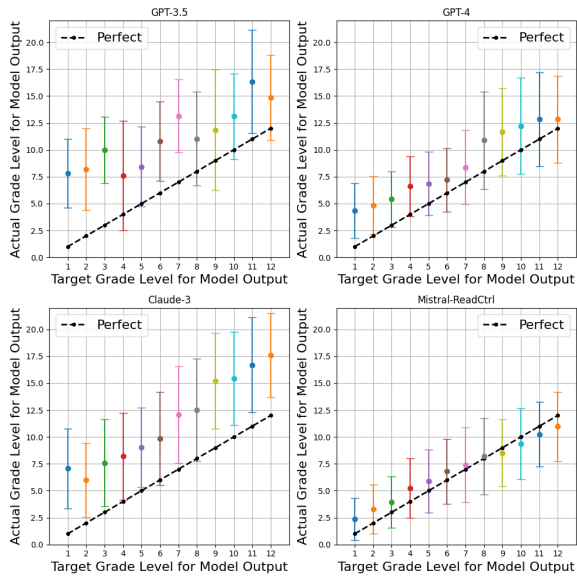Figure 9: ReadCtrl instruction following ability evaluation on WikiSmall

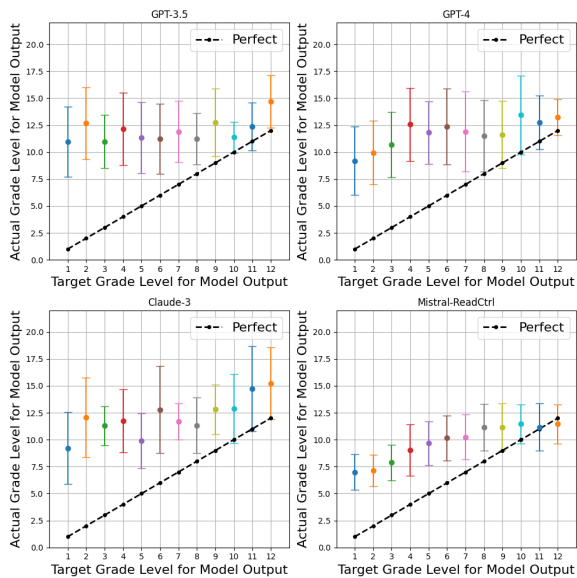Figure 10: ReadCtrl instruction following ability evaluation on MultiNLI

Figure 11: ReadCtrl instruction following ability evaluation on MRPC

H Prompts

Type	Prompt
AI Evaluation	You are evaluating two systems, both of which are trying to convert inputs to specific readability ↪ requirements to produce output suitable for the user. I will show you the input and output of the two systems on grade 2/5/8/11, respectively. Tell me which ↪ system's output you prefer by specify system 1 or system 2 or tie if the quality is the same. ↪ Please explain the reason for your preference. Input: {input} System 1 output: grade 2: {system1_2} grade 5: {system1_5} grade 8: {system1_8} grade 11: {system1_11} System 2 output: grade2: {system2_2} grade 5: {system2_5} grade 8: {system2_8} grade 11: {system2_11} Please use the following json format for your output: 'grade 2 preference': xxxx, 'grade 2 preference reasons': xxxx, 'grade 5 preference': xxxx, 'grade 5 preference reasons': xxxx, 'grade 8 preference': xxxx, 'grade 8 preference reasons': xxxx, 'grade 11 preference': xxxx, 'grade 11 preference reasons': xxxx Please only output your response following the required format, and do not output any other content. Now ↪ tell me your preference and reasons:
Readability Control Generation Prompt	Below is an instruction that describes a task. Write a response that appropriately completes the request. ### Instruction: please output an entailment at a {grade}-reading level. ### Input: {input} ### Response:

All prompts used in our work.

Document	FKG	GFI	CLI
The tarantula, the trickster character, spun a black cord and, attaching it to the ball, crawled away fast to the east, pulling on the cord with all his strength.	9.9	15.74	14.8
Requested level: second-grade reading level(Mistral ReadCtrl) The tarantula spun a black cord. He attached it to the ball. He crawled away to the east. He pulled on the cord with all his strength.	0.0	4.2	0.8
Requested level: second-grade reading level(GPT-4) The tarantula tricked others by making a black cord. It attached it to a ball and moved quickly towards the east while pulling the cord.	4.6	6.6	5.9
Requested level: fifth-grade reading level(Mistral ReadCtrl) The tarantula, the trickster character, spun a black cord. He attached it to the ball and crawled away fast to the east. He pulled on the cord with all his strength.	2.6	6.7	4.2
Requested level: fifth-grade reading level(GPT-4) The tarantula, who is known for playing tricks, made a black cord. He attached it to the ball and quickly crawled east, pulling the cord as hard as he could.	4.4	7.33	6.2
Requested level: eighth-grade reading level(Mistral ReadCtrl) The tarantula, the trickster character, spun a black cord and attached it to the ball. He crawled away fast to the east, pulling on the cord with all his strength.	4.4	8.67	7.1
Requested level: eighth-grade reading level(GPT-4) The tarantula, the sneaky character, made a black cord and tied it to the ball. It quickly moved east, pulling the cord strongly.	4.2	8.08	6.2
Requested level: eleventh-grade reading level(Mistral ReadCtrl) The tarantula, the trickster character, spun a black cord and attached it to the ball before crawling away to the east, pulling on the cord with all his strength.	11.1	14.36	14.8
Requested level: eleventh-grade reading level(GPT-4) The character of the tarantula, known for trickery, spun a black cord. He attached this cord to the ball and rapidly scuttled eastward, tugging at the cord with maximum force.	6.8	10.0	9.0

Table 4: Examples of generated summaries for different readability levels measured using FKG, GFI and CLI metrics.