Cognitive-Level Adaptive Generation via Capability-Aware Retrieval and Style Adaptation

Qingsong Wang[1], **Tao Wu**[1], **Wang Lin**[1], **Yueying Feng**[1]
Gongsheng Yuan[1], **Chang Yao**[1] , **Jingyuan Chen**[1*]
[1] Zhejiang University
wqsong@zju.edu.cn
jingyuanchen@zju.edu.cn

Abstract

Large Language Models (LLMs) have demonstrated strong performance in open-ended generation tasks. However, they often struggle to adapt content to users with differing cognitive capacities, leading to a phenomenon we term cognitive misalignment. This issue arises in two forms: knowledge-level misalignment, where content is too complex or too simplistic relative to user understanding, and presentation style misalignment, where the structure or tone hinders effective comprehension. To address these challenges, we propose the Cognitive-Level Alignment Framework (CLAF), a general-purpose generation framework that aligns both knowledge complexity and presentation style with user cognition. CLAF integrates a capability-aware retrieval module based on a hierarchical knowledge graph and a style optimization module guided by Bloom's taxonomy and preference learning. Additionally, a knowledge-controllable generation component ensures consistency and relevance throughout the output. To support training and evaluation, we construct Scale, a cognitively annotated dataset containing responses at multiple comprehension levels per query. Empirical results show that CLAF enhances the adaptability and informativeness of LLM outputs across a range of user profiles, offering a robust solution to cognitive-level alignment in real-world applications. Code and dataset are available at: https://github.com/APTX574/lg

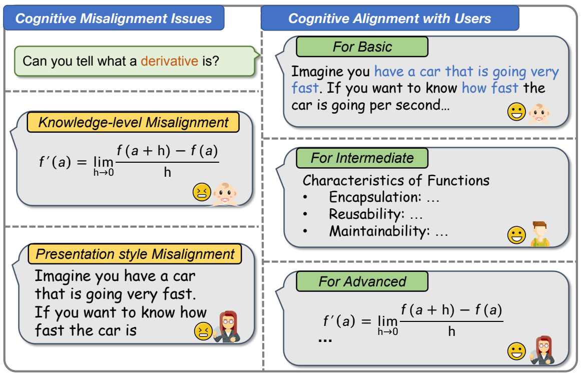

Cognitive Misalignment Issues
Can you tell what a derivative is?
Knowledge-level Misalignment
$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$
Presentation style Misalignment
Imagine you have a car that is going very fast. If you want to know how fast the car is
Cognitive Alignment with Users
For Basic
Imagine you have a car that is going very fast. If you want to know how fast the car is going per second...
For Intermediate
Characteristics of Functions • Encapsulation: ... • Reusability: ... • Maintainability: ...
For Advanced
$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$ ...

Figure 1: **Comparison of Cognitive Misalignment and Alignment in LLMs.** The left side illustrates cognitive misalignment, making the content difficult to understand or boring. In contrast, the right side demonstrates correct cognitive alignment, where everyone receives a suitable response.

1 Introduction

Large Language Models (LLMs) have shown remarkable capabilities in the field of education (Lin et al., 2025; Wang et al., 2025; Huang et al., 2025). Building upon these strengths, LLMs also enable personalized education by adapting their responses to individual learners' needs and communication styles across diverse educational and professional scenarios (Han et al., 2025; Wu et al., 2025a; Kwon et al., 2024; Feng et al., 2024b). Central to the effectiveness of these models is their capacity to adapt responses to users' varying levels of cognitive ability—ensuring that generated content is not only accurate but also aligned with the users' capacity for comprehension (Poole-Dayan et al., 2024; Wu et al., 2025b). However, current LLMs frequently fail to achieve this alignment, resulting in a phenomenon we term **cognitive misalignment**, which impairs the instructional efficacy of model outputs (Liu et al., 2024a; Rooein et al., 2023).

Cognitive misalignment manifests in two primary forms, as illustrated in Figure 1. The first is *knowledge-level misalignment* (He-Yueya et al., 2024), where the complexity of the content exceeds or underestimates users' cognitive capacity. For instance, a novice user may receive explanations embedded with technical jargon, while an expert user may be presented with overly simplistic content, leading to disengagement or frustration. The second is *presentation style misalignment* (Sonkar et al., 2024), which arises when the communica-

[*] Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 11054–11069
November 4-9, 2025 ©2025 Association for Computational Linguistics

tive approach fails to align with users' instructional needs. Similar to how educators tailor pedagogical strategies to users, LLMs should ideally adapt their rhetorical style, explanatory granularity, and instructional scaffolding accordingly. However, many models default to rigid stylistic patterns, resulting in suboptimal educational interaction for users across varied cognitive levels.

While recent efforts in personalized text generation have made notable progress (Liu et al., 2024c; Singh et al., 2024), they generally fall short of addressing these two dimensions of cognitive misalignment. Most approaches focus on user interests or interaction history, with limited consideration of users' cognitive ability. As a result, such approaches often capture **what** users are curious about, but not **how** that information should be **structured for effective comprehension**. Similarly, presentation style adaptation efforts typically depend on extensive personalization data, which is unavailable in many real-world contexts where users are represented by coarse-grained profiles (*e.g.*, "middle school student" or "domain expert") (Liu et al., 2024b). These abstractions provide insufficient granularity for pedagogically appropriate adaptation, resulting in uniform outputs that inadequately serve diverse cognitive needs.

To address these limitations, we propose the **C**ognitive-**L**evel **A**lignment **F**ramework (CLAF), a novel architecture designed to jointly align both content complexity and instructional style with users' cognitive level. Grounded in principles from educational psychology, particularly Vygotsky's Zone of Proximal Development (ZPD) (Nogueira, 2001) and Bloom's Taxonomy of Educational Objectives (Huitt, 2011), CLAF integrates cognitive theory with LLM capabilities to systematically address both dimensions of cognitive misalignment. Figure 4 illustrates the overall architecture of CLAF.

To mitigate knowledge-level misalignment, CLAF employs a **capability-aware retrieval** module inspired by ZPD. This module constructs a hierarchical knowledge graph organized by cognitive complexity, allowing for the retrieval of content that is optimally challenging yet comprehensible for the user's developmental stage. To address presentation style misalignment, we introduce an **adaptive language style optimization** module informed by Bloom's taxonomy and reinforced via human preference optimization. This module adjusts the explanatory tone, rhetorical structure, and pedagogical strategies based on the user's cognitive capabilities (*e.g.*, remembering, understanding, and applying), thereby supporting the the generation of content that is both personalized and instructionally coherent. Moreover, to ensure alignment between retrieved instructional content and generated text, CLAF incorporates a **knowledge controllable generation** mechanism that constrains latent representations during decoding, thereby preserving coherence and pedagogical relevance.

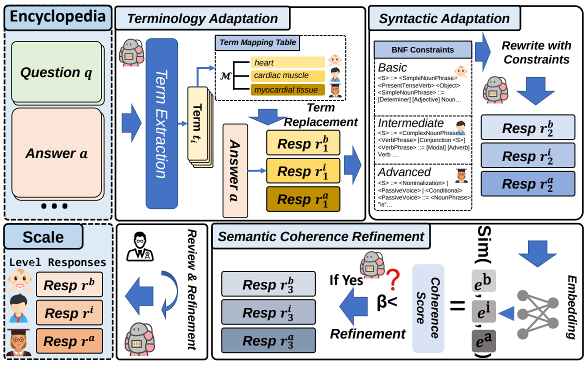


Figure 2: **Construction Pipeline of Scale.** Scale is built from encyclopedia question-answer pairs, forming a dataset where each question is associated with three responses, each customized for a different knowledge level.

To support empirical validation, we construct a novel dataset, **Scale**, comprising responses at three distinct cognitive levels per question—designed to reflect Bloom's taxonomy and instructional design principles. This dataset functions as both a training signal and an evaluation benchmark for cognitive-level adaptive generation.

Our contributions are summarized as follows:

- We identify and formally define **cognitive misalignment** as a core limitation in current LLM-based systems, framed through cognitive development theory.
- We propose **CLAF**, a cognitively grounded generation framework that jointly optimizes content retrieval, linguistic adaptation, and instructional fidelity.
- We construct **Scale**, a cognitively annotated dataset that enables systematic training and evaluation of cognitive-level alignment in LLMs.

2 Related Work

2.1 Personalized Large Language Model

The capabilities of LLMs can be leveraged for personalized teaching. (Park et al., 2024; Neshaei et al., 2024; Tang et al., 2024) utilize students' historical conversations and personal information to model students. (Hu and Wang, 2024) integrates knowledge graphs and prompt engineering into LLMs. (Deng et al., 2023; Chen et al., 2023; Li et al., 2024) select the next learning goal for students based on analysis by LLMs. All these works use students' historical information to model them, aiming to achieve personalized education.

While personalized education excels at modeling individual students, it struggles with group. They continuously analyzes individual learning trajectories, such as knowledge retention patterns and thinking path deviations. In contrast, modeling group cognitive level focuses on group cross-sectional data, categorizing students into homogeneous groups based on predefined proficiency indicators like learning stages. It then designs standardized teaching programs by identifying common features at each cognitive level.

2.2 Controlled text generation

Significant advancements have been made in controllable text generation methods now. DisCup (Zhang and Song, 2022) enhances control by introducing attribute discriminators during training and optimizing control cues through anti-likelihood training. RMT (Zhang et al., 2023) adopts residual learning and cross-attention mechanisms to achieve text generation control and seamlessly integrates with existing LLMs to enable continuous control. REI (Zheng et al., 2023) uses instructions inspired by regular expressions to control text generation through linguistic constraints. In addition to the above methods that require training, ICV (Liu et al., 2023) learns control-related vectors through contextual example text, effectively enhancing controllable text generation (CTG). MacLaSa (Ding et al., 2023) uses variational autoencoders (VAE) to map text to a compact latent space and applies ordinary differential equation (ODE) sampling methods to control multiple attributes.

3 Dataset Curation

Effective research on cognitively aligned text generation needs datasets with two key features: (1) clearly defined cognitive levels, and (2) responses that share the same meaning but vary in depth of explanation and language complexity. Existing datasets on personalization (Liu et al., 2024d; Zheng et al., 2020; Shen et al., 2024) often focus on user traits or history, but they lack structured cognitive levels and controlled variation in how the answers are written. To address this, we introduce the **Scale**, designed to support controlled generation across different cognitive levels. Each item in Scale includes a question and three aligned answers: one each at the *basic*, *intermediate*, and *advanced* levels. These levels are based on cognitive development theory (see Appendix A), and differ in how ideas are explained while keeping the core meaning intact. To check the quality of the answers, we use a multi-step human evaluation. Further information about human evaluation can be found in Appendix B.

3.1 Metadata Collection and Domain Scope

We build the core QA pairs using reliable and informative encyclopedic sources, covering topics like science, nature, biology, and cosmology. These areas are chosen for their broad appeal and their suitability for explaining topics at multiple levels of detail. The questions include types like definitions, explanations, and cause-effect reasoning, offering a range of reasoning modes. To test generalization, we also create a separate test-only set based on Chinese classical poetry, taken from national college entrance exam materials. These expert-written questions help evaluate performance across both domain and language. This part is only used in testing and not seen during training.

3.2 Data Construction Pipeline

As shown in Figure 2, we build the "Scale" in three steps to adapt it to users at different levels:

Step 1: Terminology Adaptation. Since word choice is key to abstraction, we extract important terms from each original answer. Using LLMs, we generate versions of these terms for each level (*basic*, *intermediate*, *advanced*). Experts then review and confirm these mappings, which are used to create different wordings for each answer: $\mathcal{R}_1 = \{{r_1}^b, r_1^i, r_1^a\}$.

Step 2: Syntactic Adaptation. Beyond words, sentence structure also affects comprehension. We define templates based on Backus-Naur Form (BNF) (McCracken and Reilly, 2003) and teaching guidelines. Basic answers use short, direct sentences. Intermediate ones use compound clauses and general ideas. Advanced answers include abstract sentence patterns and more layered grammar. We apply these rules using prompts to generate syntactically distinct versions: $\mathcal{R}_2 = \{r_2^b, r_2^i, r_2^a\}$.We also match the responses with typical learning pat-

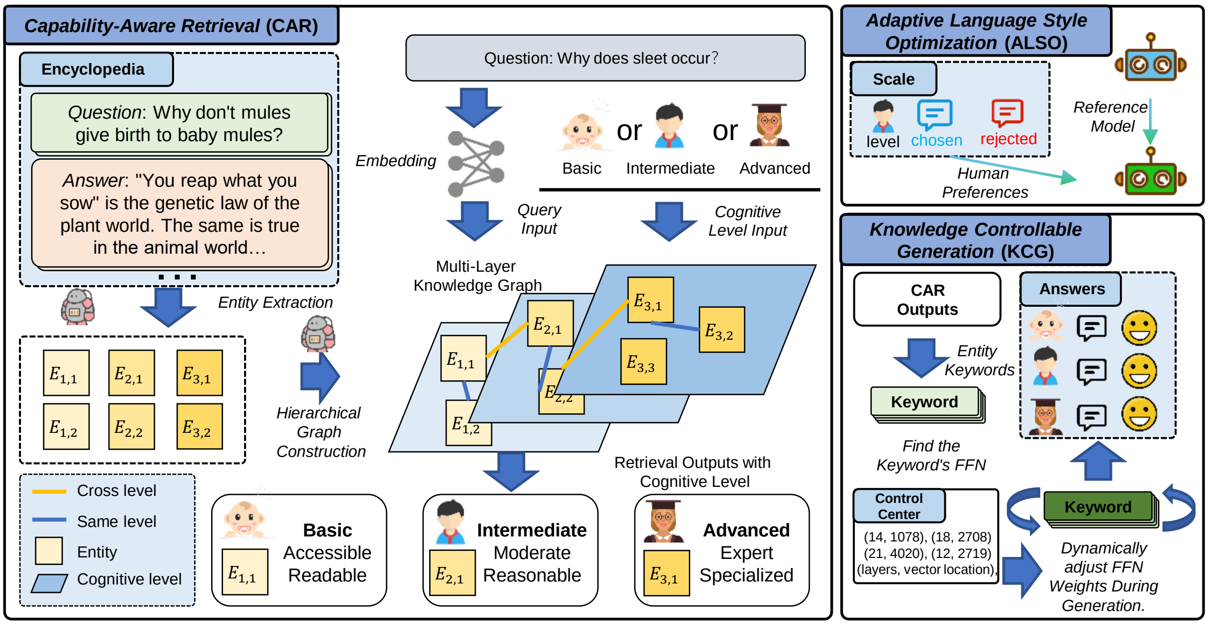


Figure 3: **Overview of the CLAF Framework.** The framework consists of three modules: Capability-Aware Retrieval, Adaptive Language Style Optimization, and Knowledge Controllable Generation.

terns: basic answers use familiar examples and surface facts; intermediate ones add general explanations and causes; advanced ones include inferential thinking and disciplinary concepts.

Step 3: Semantic Coherence Verification. To ensure all answers still mean the same thing, we use vector-based similarity checks. If a rewritten answer is too different in meaning, it is revised. This yields the final triple-set: $\mathcal{R}_3 = \{r_3^b, r_3^i, r_3^a\}$, preserving meaning while varying complexity.

3.3 Dataset Summary

The final Scale contains 593 question-and-answer entries, each with three levels of response. In addition to enabling controlled text generation, Scale supports consistent evaluation of language models' ability to change abstraction and tone while keeping meaning stable. Our modular pipeline allows future extensions to new topics and levels. The extra test set in classical Chinese literature provides a challenge for models in cross-lingual and content-rich understanding tasks.

4 Methodology

This paper presents an innovative framework by aligning generated content with distinct cognitive levels. The framework adapts the scope of knowledge, language styles, and teaching strategies in response to the user's cognitive boundaries. The proposed Cognitive Level Alignment Framework (**CLAF**) consists of three components: 1) Capability-Aware Retrieval, which delivers relevant knowledge tailored to various cognitive levels by retrieving content situated within the user's proximal development zone, as inspired by ZPD; 2) Adaptive Language Style Optimization (ALSO), which allows the model to employ language styles appropriate for different users by adapting tone and pedagogical strategy based on Bloom's taxonomy; and 3) KCG, which dynamically adjusts the scope of knowledge and ensures the output remains faithful to the retrieved content. The overview of the framework is illustrated in Figure 3.

4.1 Capability-Aware Retrieval

The first step toward cognitive alignment is ensuring the knowledge matches the user's cognitive level. Inspired by (Jin et al., 2025; Feng et al., 2024a; Wang et al., 2023) ,CAR achieves it by building a hierarchical knowledge graph derived from educational materials, where each node represents an atomic concept labeled with a cognitive tier $l \in \{0, 1, 2\}$, corresponding to basic, intermediate, and advanced levels, roughly aligned with Bloom's taxonomy (*e.g.*, Remember/Understand, Apply/Analyze, Evaluate/Create). Relations among nodes encode prerequisite chains, logical dependencies, and topic proximity.

This structure enables CLAF to perform Bloom-informed, ZPD-aware retrieval. For a user at level c, CAR traverses the graph to extract a subgraph $K_c^{(k)}$, constrained to nodes with $l \leq c$, and a depth d that increases with c . As such, beginners are

Model	Flesch Kin		Gunning Fog		SMOG		Match Level			
	Bas.↓	Adv. ↑	Bas.↓	Adv. ↑	Bas.↓	Adv.↑	Bas. ↑	Int. ↑	Adv. ↑	Avg. ↑
Closed-source LLMs										
GPT-4o	6.97	14.19	8.31	16.13	8.38	15.94	79.85	85.30	84.74	83.30
GPT-4o-FS	6.55	14.97	7.89	17.10	7.85	16.19	89.72	85.69	90.52	88.65
Gemini-1.5	6.77	13.10	8.07	14.19	8.54	14.48	83.57	74.50	82.73	80.27
Gemini-1.5-FS	6.17	13.21	7.49	14.10	8.08	14.47	80.07	84.36	82.70	82.38
Claude-3.5	7.36	15.66	8.54	16.53	9.21	16.43	73.75	85.48	73.86	77.70
Claude-3.5-FS	7.15	15.88	8.39	16.85	8.69	16.60	78.33	84.90	77.15	80.13
Qwen-Plus	6.60	13.79	7.87	14.56	8.06	15.14	67.91	89.07	70.40	75.79
Qwen-Plus-FS	6.47	13.80	7.91	14.98	8.34	15.21	77.17	87.85	79.45	81.49
Qwen-2.5-3B-Instruct										
Few-Shot	7.17	12.99	8.44	14.45	8.13	14.24 65.79	87.54	68.47	73.93	
SFT	6.91	13.29	8.26	14.69	8.32	14.38	78.33	82.80	79.37	80.17
CLAF(ours)	6.69	12.80	8.10	14.06	8.17	14.43	76.43	85.94	81.15	81.17
Qwen-2.5-7B-Instruct										
Few-Shot	6.80	13.01	8.08	13.85	8.74	14.71	76.01	86.93	75.07	79.34
SFT	6.37	13.64	7.72	14.58	8.06	**15.23**	79.00	81.15	77.55	79.23
CLAF(ours)	**5.81**	13.47	**7.16**	14.50	**8.02**	15.04	78.01	87.63	81.63	82.42
Llama-3.1-8B-Instruct										
Few-Shot	7.17	13.32	8.51	13.70	9.09	14.84	26.35	**92.80**	28.90	49.35
SFT	6.60	13.92	8.30	16.21	8.37	12.90	85.53	78.15	78.78	80.82
CLAF(ours)	6.25	**13.78**	8.19	**16.38**	8.14	14.22	**90.75**	86.30	**90.87**	**89.31**

Table 1: **Experimental Comparison of CLAF Against Other Baseline Models.** The results validate the effectiveness of the proposed framework.

exposed to foundational content, while advanced users receive broader and deeper knowledge. The retrieved concepts serve as inputs for downstream modules. Full retrieval procedures are detailed in Algorithm C.2 and Appendix C.3.

4.2 Adaptive Language Style Optimization

To further refine the alignment of content with the user's cognitive level, we introduce the ALSO module. This module leverages Direct Preference Optimization (DPO) (Rafailov et al., 2023) to tailor the language style and complexity according to the user's stage. By using Scale, ALSO adapts the style of large language models (LLMs) to match cognitive requirements, dynamically adjusting aspects such as term difficulty, sentence structure, and pedagogical approach.

Unlike static prompt-based strategies, our approach continuously adapts to the user's needs. The DPO framework fine-tunes the model by maximizing the expected reward of the output style while minimizing its divergence from a reference model. The optimization is expressed as:

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(y|x)} \left[r_\phi(x, y)\right] - \beta D_{\mathrm{KL}} \left(\pi_\theta(y|x) \,\|\, \pi_{\mathrm{ref}}(y|x)\right) \qquad (1)$$

where $\pi_\theta(y|x)$ represents the model's output distribution, $r_\phi(x, y)$ is the reward function, β controls the trade-off between reward and divergence, and D_{KL} is the Kullback-Leibler divergence between the model and the reference model.

Cognitive-level Adaptation The module tailors responses to user' capabilities through three-tier adaptation inspired by Bloom's taxonomy of cognitive objectives: For *basic-level* users, it simplifies concepts using fundamental terminology, analogies, and clear explanations aimed at fostering lower-order cognitive processes such as remembering and understanding. The output distribution $\pi_\theta(y|x)$ is optimized via reward modeling to prioritize accessibility. *Intermediate-level* users receive balanced explanations that integrate foundational knowledge with logical reasoning and contextual examples, supporting mid-level cognitive goals such as applying and analyzing. *Advanced-level* users obtain domain-specific terminology and deductive reasoning aligned with expert-level cognition, aligning with higher-order objectives such as evaluating and creating. The preference mechanism follows:

$$P(y_w \succ y_l|x) = \sigma(r(x, y_w) - r(x, y_l)), \qquad (2)$$

where $P(y_w \succ y_l|x)$ denotes the probability of output y_w being preferred over y_l, σ is the logistic function, and $r(\cdot)$ represents the reward model that evaluates cognitive alignment.

4.3 Knowledge Controllable Generation

To enhance the consistency between LLMs output and CAR's retrieval content, we incorporate a Knowledge Controllable Generation(KCG) module that enables precise control over the output. This approach, based on prior work (Feng et al., 2024c; Hu et al., 2021; Feng et al., 2024d), allows for adaptive management of output's content by adjusting the weights of vectors in the Feedforward Network (FFN) layers. The module constructs a control center for each token in the model's vocabulary, influencing the generation of domain-specific content by modifying the FFN vector weights.

The generation process consists of four stages: initialization, monitoring, adaptation, and filtering. In the initialization stage, relevant keywords from the CAR module are collected, and their corresponding FFN vectors are identified. The monitoring stage evaluates the relevance of each token generated, dynamically adjusting weights to optimize domain-specific alignment. During the adaptation stage, the weights of the control centers are modified to guide the model towards generating content that aligns with the desired knowledge scope. The weight adjustment is given by:

$$\omega_{a_i}^{t+1} = \lambda \cdot \sigma\left(-(\mu_\omega - \hat{\mu}_{a_i}^t) \cdot l_t\right), \qquad (3)$$

where σ is the sigmoid function, μ_ω is a predefined threshold, and $\hat{\mu}_{a_i}^t$ represents the cumulative alignment. This dynamic weight adjustment prevents over-specialization by resetting weights when alignment exceeds thresholds. Finally, in the filtering stage, thresholds are applied to ensure the quality and relevance of the generated content.

The KCG enables precise control over the generation process, enhancing the relevance of output content to knowledge retrieved by the CAR.

5 Experiments

5.1 Experimental Setups

Baselines. We compare CLAF with the open-source models LLaMA 3.1-8B-Instruct (Touvron et al., 2023) and Qwen-2.5-7B-Instruct (Bai et al., 2023),Qwen-2.5-3B-Instruct, as well as the closed-source models ChatGPT-4o (Achiam et al., 2023), Gemini 1.5 (Team et al., 2023), Qwen-Plus (Bai et al., 2023), and Claude 3.5 (Anthropic, 2024).

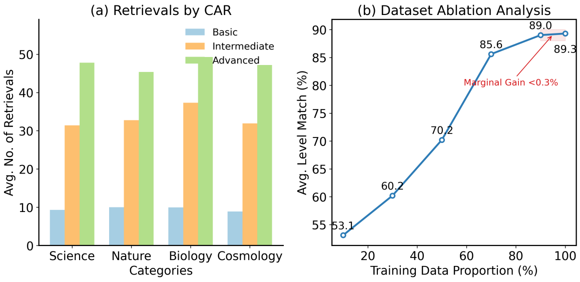


Figure 4: (a) Number of retrievals in the CAR across different knowledge levels and question types. (b) Results of the Scale-size experiment, indicating that the current dataset volume is sufficient.

Model	Flesch Kin		Level Match		
	Bas. ↓	Adv. ↑	Bas. ↑	Int. ↑	Adv. ↑
CLAF	**8.19**	**16.38**	**90.75**	**86.30**	**90.87**
- w/o KCG	8.24	16.36	89.99	85.75	90.22
- w/o CAR	8.51	13.70	84.78	78.35	78.83
- w/o ALSO	9.79	14.06	57.17	76.67	59.11

Table 2: **Ablation Study of the CLAF Framework.** Results demonstrating the effectiveness of each component model within the CLAF framework.

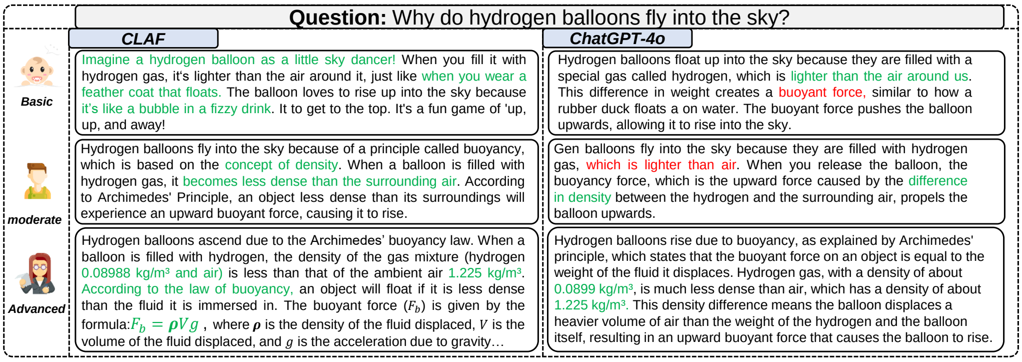
Metrics. We assess text readability and complexity using Flesch-Kincaid Grade Level(FK) (Solnyshkina et al., 2017), Gunning Fog Index (Gunning, 1969), and SMOG Index (Mc Laughlin, 1969). Cognitive hierarchical alignment is evaluated using GPT-o1. See prompts in Appendix E.

5.2 Results

We evaluated various models, including our proposed one, across different cognitive levels, with results in Table 1. The models' outputs were assessed using the Flesch-Kincaid Grade Level, Gunning Fog Index, SMOG, and Level Match to measure precision and appropriateness. "FS" refers to few-shot prompts, and "Supervised Fine-Tuning (SFT)" involves fine-tuning with CLAF.

The Flesch-Kincaid, Gunning Fog, and SMOG indices assess sentence complexity, with higher scores indicating more difficulty. Basic-level users benefit from lower indices for better comprehension, while Advanced-level users benefit from higher indices. Intermediate-level users require a balance between the two for optimal learning.

Overall Performance. The results in Table 1 show that CLAF significantly enhances cognitive alignment in text generation. By integrating ALSO with CAR, CLAF improves readability and hierarchical matching rates. It reduces the Flesch-Kincaid score by 5.3% for basic-level outputs on Llama-3.1-



Question: Why do hydrogen balloons fly into the sky?

	CLAF	ChatGPT-4o
Basic	Imagine a hydrogen balloon as a little sky dancer! When you fill it with hydrogen gas, it‘s lighter than the air around it, just like when you wear a feather coat that floats. The balloon loves to rise up into the sky because it’s like a bubble in a fizzy drink. It to get to the top. It's a fun game of 'up, up, and away!	Hydrogen balloons float up into the sky because they are filled with a special gas called hydrogen, which is lighter than the air around us. This difference in weight creates a buoyant force, similar to how a rubber duck floats a on water. The buoyant force pushes the balloon upwards, allowing it to rise into the sky.
moderate	Hydrogen balloons fly into the sky because of a principle called buoyancy, which is based on the concept of density. When a balloon is filled with hydrogen gas, it becomes less dense than the surrounding air. According to Archimedes' Principle, an object less dense than its surroundings will experience an upward buoyant force, causing it to rise.	Gen balloons fly into the sky because they are filled with hydrogen gas, which is lighter than air. When you release the balloon, the buoyancy force, which is the upward force caused by the difference in density between the hydrogen and the surrounding air, propels the balloon upwards.
Advanced	Hydrogen balloons ascend due to the Archimedes’ buoyancy law. When a balloon is filled with hydrogen, the density of the gas mixture (hydrogen 0.08988 kg/m³ and air) is less than that of the ambient air 1.225 kg/m³. According to the law of buoyancy, an object will float if it is less dense than the fluid it is immersed in. The buoyant force (F_b) is given by the formula:$F_b = \boldsymbol{\rho} V g$, where $\boldsymbol{\rho}$ is the density of the fluid displaced, V is the volume of the fluid displaced, and g is the acceleration due to gravity…	Hydrogen balloons rise due to buoyancy, as explained by Archimedes' principle, which states that the buoyant force on an object is equal to the weight of the fluid it displaces. Hydrogen gas, with a density of about 0.0899 kg/m³, is much less dense than air, which has a density of about 1.225 kg/m³. This density difference means the balloon displaces a heavier volume of air than the weight of the hydrogen and the balloon itself, resulting in an upward buoyant force that causes the balloon to rise.

Figure 5: **Case Study.** The results indicate that our CLAF achieves better cognitive level alignment.

Model	Level Match		
	Bas. ↑	Int. ↑	Adv. ↑
Closed-source LLMs			
GPT-4o-FS	50.01	87.81	55.96
Gemini-1.5-FS	51.05	86.61	57.30
Claude-3.5-FS	51.27	83.63	61.19
Qwen-Plus-FS	**56.78**	**88.01**	**61.85**
Open-source LLMs			
QwQ-32B-Preview-FS	35.00	71.51	58.94
Llama-3.1-8B	22.05	**84.32**	38.26
Llama-3.1-8B-SFT	55.46	79.04	57.71
CLAF(ours)	**60.55**	80.07	**64.97**

Table 3: **Experimental Results on the Chinese Classical Poetry Appreciation Dataset.** Results demonstrating the strong generalization ability of CLAF across different domains and languages.

8B and increases the Gunning Fog score by 1.05 points for advanced-level outputs, indicating effective complexity management. The SMOG scores (Bas.=8.02, Int.=12.81, Adv.=15.04) on Qwen-7B surpass Gemini-1.5-FS, validating KCG's role in academic depth modulation. CLAF enables Qwen-7B to achieve an 81.63 advanced matching rate, outperforming Qwen-Plus (70.40). For open-source models, it achieves an 89.31% average matching rate on Llama-3.1-8B, surpassing Few-Shot and SFT baselines by 40.96 and 8.49 percentage points, respectively. The CAR mechanism boosts advanced generation to a 90.87% matching rate, outperforming state-of-the-art closed models.

Effectiveness of the Scale. The Scale demonstrates effectiveness in three ways: (1) In closed-source models, few-shot prompting with Scale improves performance, with GPT-4o-FS achieving a 90.52% advanced-level match (+5.78% over zero-shot); (2) For open-source models, full-parameter SFT training with Scale significantly enhances capabilities, with Llama-8B-SFT reaching an 80.82% average match rate (+63.8% over the few-shot baseline) and improving basic-level performance from 26.35% to 85.53%; (3) Scale identifies catastrophic failure patterns in unadapted models, such as Llama-FewShot's low basic (26.35%) and advanced (28.90%) matching rates, and highlights the bias toward intermediate content (92.80% Int. match), effectively diagnosing LLM bias through contrastive evaluation.

CLAF	FK-Bas.↓	FK-Adv.↑	LM-Bas.↑	LM-Int.↑	LM-Adv.↑
0-25%	**8.34**	**14.32**	**87.69**	**84.31**	**86.76**
w/o KCG 0-25%	8.45	13.82	84.34	84.39	80.14
25-50%	**8.29**	**15.97**	**88.41**	**85.12**	**88.09**
w/o KCG 25-50%	8.33	14.88	85.43	82.46	85.35

Table 4: **Ablation Study of the KCG Under Poor Retrieval Conditions**. The results show that KCG brings more significant performance gains when the quality of retrieved results is limited.

Impact of Model Scaling. Model scaling experiments show framework adaptability: Qwen-7B improves the average matching rate by 1.25 points (82.42 vs 81.17) over Qwen-3B, with SMOG scores rising from 14.43 to 15.04, indicating larger models better utilize KCG signals. Notably, our method surpasses most closed-source models in advanced matching using only 10-25% of the parameters of commercial models (Llama-3.1-8B vs Claude-3.5), demonstrating effectiveness under limited computation. These results validate the tripartite mechanism: 1) ALSO creates granular linguistic representations via DPO; 2) CAR dynamically constrains knowledge boundaries during generation; 3) KCG ensures output content relevance to the question. The framework's multi-objective optimization enables new applications in educational content generation and personalized information delivery with hierarchical adaptation.

Result on Poetry Appreciation. As shown in Table 3, our method demonstrates effectiveness on Chinese classical poetry appreciation. When built upon the Llama-3.1-8B-Instruct base model, our framework achieves 60.55% basic-level matching (+38.5 points over vanilla Llama-3.1-8B) and 64.97% advanced-level accuracy, surpassing all closed-source models in basic-level adaptation (vs Qwen-Plus-FS=56.78) while maintaining competitive intermediate-level performance (80.07 vs 88.01). Notably, the advanced-level result (64.97%) approaches closed-source models' upper bound (Claude-3.5-FS=61.19, Qwen-Plus-FS=61.85), proving our CLAF effectively handles cultural domain-specific complexity despite the base model's limited Chinese poetry training data.

5.3 Ablation Study

Ablation Study in CLAF. We conducted an ablation study on FK scores and Level Match accuracy. As shown in Table 2, removing any component significantly reduces performance. Excluding CAR results in a 16.4% drop in advanced complexity metrics, emphasizing its importance in maintaining professional depth. Disabling ALSO leads to a 37% reduction in Basic level match accuracy, highlighting its crucial role in cognitive alignment.

The KCG module strengthens our framework by guiding generation through keywords from retrieved content. Its impact is most notable when retrieval quality is low. As shown in Table 4, KCG significantly boosts performance in the 0–25% retrieval quality range (*e.g.*, LM-Bas. +3.35, LM-Adv. +6.62) and shows consistent gains in the 25–50% range (*e.g.*, FK-Adv. +1.09, LM-Bas. +2.98). These results highlight KCG's compensatory role, helping maintain generation quality by extracting key concepts from suboptimal retrievals, thereby enhancing the CLAF's robustness across varying retrieval conditions.

The Figure 4 (a) shows the number of knowledge retrieved by CAR at different levels, demonstrating the wider range of knowledge that our CAR can provide as the user's cognition improves.

Ablation Study in Scale. The scaling experiment in Figure 4 (b) reveals two phases. As training data increases from 10% to 70% (52→364 samples), accuracy jumps 32.5 points (53.1%→85.62%), reaching 95.3% of full-data performance. Beyond 70%, gains taper off (70%→90%: +3.4; 90%→100%: +0.28), indicating performance saturation. This non-linear trend shows our method effectively captures core stylistic features with limited data, while the rest mainly refines edge cases. The plateau after 90% confirms our 593-sample dataset achieves near-optimal utility via the CAR and KCG.

Model	Flu.	Align.	Guid.	Bas.	Adv.
CLAF(our)	4.41	**3.89**	**4.58**	**53.47%**	**60.34%**
ChatGPT-4o	**4.52**	3.51	4.23	32.15%	21.52%
SFT	4.23	3.47	4.31	14.38%	18.14%

Table 5: **Human Evaluation Results.** Human evaluation indicate that outputs from CLAF are more preferred compared to other open-source models, demonstrating both the model quality and the effectiveness of Scale.

5.4 Human Evaluation

To complement automatic evaluation metrics, we conducted comprehensive human evaluations involving expert assessments and user preference:

- **Expert Evaluation**: Three graduate-level education specialists assessed 100 tri-level responses using a 5-point Likert scale. Evaluation criteria included fluency (Flu.), cognitive-level alignment (Align.), and pedagogical effectiveness (Guid.).
- **User Preference Evaluation**: We recruited three elementary school students (Bas.) and three graduate students (Adv.) in biochemistry to represent novice and advanced users. Participants compared answers generated by CLAF, ChatGPT-4o, and a supervised fine-tuned (SFT) model, and selected their preferred responses.

The results are shown in Table 5, and the results show that CLAF has better results than other models in manual evaluation.

5.5 Case Study

Figure 5 showcases CLAF's hierarchical advantages across three comparative levels. At the basic level, our CLAF module enables vivid metaphors and anthropomorphic language (*e.g.*, "sky dancers"), outperforming GPT-4o's more technical phrasing (*e.g.*, "buoyancy"). At the intermediate level, both models explain density differences, but ours adds clarity with a structured three-step logic: "Archimedes" law → density comparison → force chain." At the advanced level, both give accurate answers, but our CAR module distinguishes itself with precise mathematical expressions (*e.g.*, $F_b = \rho V g$, $H_2 = 0.08988$ kg/m^3), offering a solid foundation for academic research.

6 Conclusions

We address the challenge of cognitive-level misalignment in LLM-based generation by introducing Scale, a dataset with tri-level cognitively aligned answers. Building on Scale, we propose CLAF, a modular framework that adapts content and style to users' cognitive capacity. Experiments show that

CLAF significantly improves cognitive alignment and controllability. This work lays the groundwork for cognitively adaptive generation across education and other user-facing domains.

Acknowledge: This research was supported by grants from the “Pioneer” and “Leading Goose” R&D Program of Zhejiang (2025C02022), National Natural Science Foundation of China (No.62307032) and the Key Research and Development Program of Zhejiang Province (No.2024C03270).

Limitations

In this section, we discuss the limitations of our work as below:

- Currently, CLAF categorizes users into three levels. While this approach provides a general framework, a more refined categorization might lead to better adaptive responses and more accurate modeling of student learning needs. We will leave this as future work.
- Our work focuses on cognitive alignment and acknowledges the limits of a three-level categorization; however, it does not account for motivational, affective, or individual user differences, which may affect its practical applicability in real-world settings. Future extensions will explore incorporating these factors into the framework.

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*.

AI Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Eason Chen, Ray Huang, Han-Shin Chen, Yuen-Hsien Tseng, and Liang-Yi Li. 2023. Gptutor: a chatgpt-powered programming tool for code explanation. In *International Conference on Artificial Intelligence in Education*, pages 321–327. Springer.

Yang Deng, Zifeng Ren, An Zhang, Wenqiang Lei, and Tat-Seng Chua. 2023. Towards goal-oriented intelligent tutoring systems in online education. *arXiv preprint arXiv:2312.10053*.

Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2023. Maclasa: Multi-aspect controllable text generation via efficient sampling from compact latent space. *arXiv preprint arXiv:2305.12785*.

Yueying Feng, WenKang Han, Tao Jin, Zhou Zhao, Fei Wu, Chang Yao, Jingyuan Chen, et al. 2024a. $\hat{E}^3$: Exploring embodied emotion through a large-scale egocentric video dataset. *Advances in Neural Information Processing Systems*, 37:118182–118197.

Yueying Feng, Fan Ma, Wang Lin, Chang Yao, Jingyuan Chen, and Yi Yang. 2024b. Fedpam: Federated personalized augmentation model for text-to-image retrieval. In *Proceedings of the 2024 International Conference on Multimedia Retrieval*, pages 1185–1189.

Zijian Feng, Hanzhang Zhou, Kezhi Mao, and Zixiao Zhu. 2024c. FreeCtrl: Constructing control centers with feedforward layers for learning-free controllable text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7627–7640, Bangkok, Thailand. Association for Computational Linguistics.

Zijian Feng, Hanzhang Zhou, Zixiao Zhu, Junlang Qian, and Kezhi Mao. 2024d. Unveiling and manipulating prompt influence in large language models. *arXiv preprint arXiv:2405.11891*.

Robert Gunning. 1969. The fog index after twenty years. *Journal of Business Communication*, 6(2):3–13.

Wenkang Han, Wang Lin, Liya Hu, Zhenlong Dai, Yiyun Zhou, Mengze Li, Zemin Liu, Chang Yao, and Jingyuan Chen. 2025. Contrastive cross-course knowledge tracing via concept graph guided knowledge transfer. *arXiv preprint arXiv:2505.13489*.

Joy He-Yueya, Wanjing Anya Ma, Kanishk Gandhi, Benjamin W Domingue, Emma Brunskill, and Noah D Goodman. 2024. Psychometric alignment: Capturing human knowledge distributions via language models. *arXiv preprint arXiv:2407.15645*.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2021. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. *arXiv preprint arXiv:2108.02035*.

Silan Hu and Xiaoning Wang. 2024. Foke: A personalized and explainable education framework integrating foundation models, knowledge graphs, and prompt engineering. In *China National Conference on Big Data and Social Computing*, pages 399–411. Springer.

Zihan Huang, Tao Wu, Wang Lin, Shengyu Zhang, Jingyuan Chen, and Fei Wu. 2025. Autogeo: Automating geometric image dataset creation for enhanced geometry understanding. *IEEE Transactions on Multimedia*.

William Huitt. 2011. Bloom et al.'s taxonomy of the cognitive domain. *Educational psychology interactive*, 22:1–4.

Tao Jin, Wang Lin, Hao Jiang, Jian Wang, Xiao Jin, Zhimeng Zhang, Jingyuan Chen, Zhou Zhao, and Zhongfei Zhang. 2025. Recognize-and-tell: Generating video captions with textual cue in scene. *Expert Systems with Applications*, page 127831.

Soonwoo Kwon, Sojung Kim, Minju Park, Seunghyun Lee, and Kyuseok Kim. 2024. BIPED: Pedagogically informed tutoring system for ESL education. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3389–3414, Bangkok, Thailand. Association for Computational Linguistics.

Qingyao Li, Wei Xia, Kounianhua Du, Qiji Zhang, Weinan Zhang, Ruiming Tang, and Yong Yu. 2024. Learning structure and knowledge aware representation with large language models for concept recommendation. *arXiv preprint arXiv:2405.12442*.

Wang Lin, QingSong Wang, Yueying Feng, Shulei Wang, Tao Jin, Zhou Zhao, Fei Wu, Chang Yao, and Jingyuan Chen. 2025. Non-natural image understanding with advancing frequency-based vision encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29756–29766.

Biao Liu, Ning Xu, and Xin Geng. 2024a. Progressively label enhancement for large language model alignment. *arXiv preprint arXiv:2408.02599*.

Jing Liu, Lele Sun, Weizhi Nie, Yuting Su, Yongdong Zhang, and Anan Liu. 2024b. Inter-and intra-domain potential user preferences for cross-domain recommendation. *IEEE Transactions on Multimedia*.

Jiongnan Liu, Yutao Zhu, Shuting Wang, Xiaochi Wei, Erxue Min, Yu Lu, Shuaiqiang Wang, Dawei Yin, and Zhicheng Dou. 2024c. Llms+ persona-plug= personalized llms. *arXiv preprint arXiv:2409.11901*.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. 2023. In-context vectors: Making in context learning more effective and controllable through latent space steering. *arXiv preprint arXiv:2311.06668*.

Zhengyuan Liu, Stella Xin Yin, Geyu Lin, and Nancy F Chen. 2024d. Personality-aware student simulation for conversational intelligent tutoring systems. *arXiv preprint arXiv:2404.06762*.

G Harry Mc Laughlin. 1969. Smog grading-a new readability formula. *Journal of reading*, 12(8):639–646.

Daniel D McCracken and Edwin D Reilly. 2003. Backus-naur form (bnf). In *Encyclopedia of Computer Science*, pages 129–131.

Seyed Parsa Neshaei, Richard Lee Davis, Adam Hazimeh, Bojan Lazarevski, Pierre Dillenbourg, and Tanja Käser. 2024. Towards modeling learner performance with large language models. *arXiv preprint arXiv:2403.14661*.

Carlos Fino Nogueira. 2001. Vygotsky e a zona de desenvolvimento proximal (zdp): três implicações pedagógicas. *Revista Portuguesa de educação*, 14(2):0.

Minju Park, Sojung Kim, Seunghyun Lee, Soonwoo Kwon, and Kyuseok Kim. 2024. Empowering personalized learning through a conversation-based tutoring system with student modeling. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–10.

Elinor Poole-Dayan, Deb Roy, and Jad Kabbara. 2024. Llm targeted underperformance disproportionately impacts vulnerable users. *arXiv preprint arXiv:2406.17737*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Donya Rooein, Amanda Cercas Curry, and Dirk Hovy. 2023. Know your audience: Do llms adapt to different age and education levels? *arXiv preprint arXiv:2312.02065*.

Xiaoteng Shen, Rui Zhang, Xiaoyan Zhao, Jieming Zhu, and Xi Xiao. 2024. Pmg: Personalized multimodal generation with large language models. In *Proceedings of the ACM on Web Conference 2024*, pages 3833–3843.

Harmanpreet Singh, Nikhil Verma, Yixiao Wang, Manasa Bharadwaj, Homa Fashandi, Kevin Ferreira, and Chul Lee. 2024. Personal large language model agents: A case study on tailored travel planning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 486–514.

Marina Solnyshkina, Radif Zamaletdinov, Ludmila Gorodetskaya, and Azat Gabitov. 2017. Evaluating text complexity and flesch-kincaid grade level. *Journal of social studies education research*, 8(3):238–248.

Shashank Sonkar, Kangqi Ni, Sapana Chaudhary, and Richard G Baraniuk. 2024. Pedagogical alignment of large language models. *arXiv preprint arXiv:2402.05000*.

Yihong Tang, Bo Wang, Dongming Zhao, Xiaojia Jin, Jijun Zhang, Ruifang He, and Yuexian Hou. 2024. Morpheus: Modeling role from personalized dialogue history by exploring and utilizing latent space. *arXiv preprint arXiv:2407.02345*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Shulei Wang, Shuai Yang, Wang Lin, Zirun Guo, Sihang Cai, Hai Huang, Ye Wang, Jingyuan Chen, and Tao Jin. 2025. Omni-chart-600k: A comprehensive dataset of chart types for chart understanding. *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4051–4069.

Ye Wang, Wang Lin, Shengyu Zhang, Tao Jin, Linjun Li, Xize Cheng, and Zhou Zhao. 2023. Weakly-supervised spoken video grounding via semantic interaction learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10914–10932.

Tao Wu, Jingyuan Chen, Wang Lin, Mengze Li, Yumeng Zhu, Ang Li, Kun Kuang, and Fei Wu. 2025a. Embracing imperfection: Simulating students with diverse cognitive levels using llm-based agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 9887–9908. Association for Computational Linguistics.

Tao Wu, Jingyuan Chen, Wang Lin, Jian Zhan, Mengze Li, Kun Kuang, and Fei Wu. 2025b. Personalized distractor generation via mcts-guided reasoning reconstruction. *arXiv preprint arXiv:2508.11184*.

Hanqing Zhang, Sun Si, Haiming Wu, and Dawei Song. 2023. Controllable text generation with residual memory transformer. *arXiv preprint arXiv:2309.16231*.

Hanqing Zhang and Dawei Song. 2022. Discup: Discriminator cooperative unlikelihood prompt-tuning for controllable text generation. *arXiv preprint arXiv:2210.09551*.

Xin Zheng, Hongyu Lin, Xianpei Han, and Le Sun. 2023. Toward unified controllable text generation via regular expression instruction. *arXiv preprint arXiv:2309.10447*.

Yinhe Zheng, Rongsheng Zhang, Minlie Huang, and Xiaoxi Mao. 2020. A pre-training based personalized dialogue generation model with persona-sparse data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9693–9700.

A Detail of Cognitive Level

In our study, we construct a dataset by carefully selecting a diverse set of questions q from each source, along with their corresponding answers a. To capture cognitive adaptability, we generate three distinct responses for each answer, each tailored to a specific cognitive level: **basic**, **intermediate**, and **advanced**. These levels correspond to key learning stages, ensuring that the responses are both educationally relevant and cognitively engaging:

- **Basic level**: This level targets early childhood to elementary school users. It focuses on providing simplified explanations and structured guidance to support foundational understanding and cognitive growth.
- **Intermediate level**: Geared towards middle and high school students, this level introduces more complex concepts and encourages moderate reasoning. It aims to bridge the gap between basic comprehension and advanced analytical skills, fostering critical thinking and problem-solving abilities.
- **Advanced level**: Designed for undergraduate students and beyond, this level explores complex, abstract concepts that require strong analytical skills. It challenges users to engage with sophisticated ideas, promoting deep understanding and intellectual development.

B Dataset Curation Detail

B.1 Dataset Matadata

Category	Quantity
science	153
nature	140
biology	192
cosmology	35
poetry	73
total	593

B.2 Detail about Terminology Adaptation

This appendix documents the prompt design and methodological details for terminology processing in our tiered knowledge adaptation framework.

B.2.1 Terminology Extraction

We ultimately obtained 1985 pairs belonging to the mapping using the following prompts.

Identify domain-specific terms requiring adaptation. **Prompt Template:**

```
You are a linguistics expert analyzing
↪  educational content. Carefully extract all
↪  key technical terms from the following
↪  text that require complexity adjustment
↪  for different learner levels. Consider:

1. Specialized vocabulary beyond daily usage
2. Abstract conceptual terminology
3. Domain-specific jargon
4. Terms with complexity variations across
↪  cognitive levels
Return a JSON list without commentary:
{
  "terms": ["term1", "term2", ...]
}
Text: {insert_content}
```

B.2.2 Cognitive-Level Mapping Prompt

Generate tiered synonyms aligned with educational stages:

Prompt Template:

```
As an expert lexicographer with pedagogical
↪  training, generate three cognitive-level
↪  appropriate synonyms for the term "{term}"
↪  using these guidelines:

Basic ({target_age}):
- Simple concrete language
- Maximum 2 syllables preferred
- Use everyday analogues
Example: "Photosynthesis" → "Plant
↪  food-making"
Intermediate ({target_age}):
- Introduce conceptual components
- Allow 3-4 syllables
- Maintain precision while improving
↪  accessibility
Example: "Mitochondria" → "Cell energy
↪  factories"
Advanced ({target_age}):
- Technical precision prioritized
- Permit specialized jargon
- Match academic literature usage
Example: "Catalyst" → "Chemical reaction
↪  mediator"

Provide JSON output:
{
  "term": "{term}",
  "cognitive_mapping": {
    "basic": "...",
    "intermediate": "...",
    "advanced": "..."
  }
}
```

B.3 Detail about Syntactic Adaptation

B.3.1 BNF Constraints for Cognitive Levels

We formalize syntactic complexity control through BNF grammars:

Basic Level Grammar

```
<S> ::= <SimpleNounPhrase> <PresentTenseVerb>
↪  <Object>
<SimpleNounPhrase> ::= [Determiner]
↪  [Adjective] Noun
<Object> ::= Noun | "that" <SimpleClause>
<SimpleClause> ::= <SimpleNounPhrase> Verb
```

Features: Only simple present tense Maximum 1 subordinate clause Prohibited structures: passives, modals, gerunds

Intermediate Level Grammar

```
<S> ::= <ComplexNounPhrase> <VerbPhrase>
↪  [Conjunction <S>]
<VerbPhrase> ::= [Modal] [Adverb] Verb
↪  [PrepositionalPhrase]
<ComplexNounPhrase> ::= [Determiner]
↪  [Adjective+] Noun [RelativeClause]
<RelativeClause> ::= "that" <VerbPhrase> |
↪  "which" <VerbPhrase>
```

Features: Allows past/future tenses Permits 2-level clause nesting Limited modals (can/may/will)

Advanced Level Grammar

```
<S> ::= <Nominalization> | <PassiveVoice> |
↪  <Conditional>
<PassiveVoice> ::= <NounPhrase> "is"
↪  VerbPastParticiple [PrepositionalPhrase]
<Conditional> ::= "If" <S> "," ("then" <S> |
↪  <ModalVerb> <S>)
<Nominalization> ::= <GerundPhrase> Verb
↪  <ComplexNounPhrase>
```

Features: Supports all verb forms (gerunds, participles) Allows multi-clause embeddings Permits abstract syntactic constructions

B.3.2 Syntax Adjustment Prompt

Transform text to match target cognitive level's BNF grammar:

Prompt Template:

```
As a linguistic editor, rewrite the following
↪  text strictly adhering to these BNF
↪  constraints for {cognitive_level}:
{insert_relevant_BNF_rules}
Key requirements:
1. Sentence structure must validate against BNF
2. Lexical complexity matches
↪  {cognitive_level} terminology
3. Preserve original semantic content

Input: {text}
Output (JSON):
{
  "original": "...",
  "restructured": "...",
  "validation": {"pass": bool, "issues": []}
}
```

B.3.3 Consistency Revision Pipeline

Embeddings are generated using Sentence-BERT (all-mpnet-base-v2) with cosine similarity measurement.

Prompt Template:

~~~
Input:
```json
{
  "basic": "[simple sentence]",
  "intermediate": "[mid-level sentence]",
  "advanced": "[complex sentence]"
}
Instruction:
"Detect factual conflicts across three
↪  cognitive-level sentences. Revise only
↪  conflicting parts using
↪  strikethrough→correction while preserving
↪  original complexity:
Cross-check scientific accuracy
Modify contradictions only
Maintain sentence structure
Output:
{
  "revisions": {
    "basic": "[revised]",
    "intermediate": "[revised]",
    "advanced": "[revised]"
  },
}
~~~

C Construction of Adaptive Knowledge Graph

C.1 Extraction of Concepts

Extracting surrogate cognitive-level entities and relations from text

Prompt Template:

```
Given a text document that is potentially
↪  relevant to this activity and a list of
↪  entity types, identify all entities of
↪  those types from the text and all
↪  relationships among the identified
↪  entities. Use {language} as output
↪  language.
-Steps-
1. Identify all entities. For each identified
↪  entity, extract the following information:
- entity_name: Name of the entity, use same
↪  language as input text. If English,
↪  capitalized the name.
- entity_type: One of the following types:
↪  [{entity_types}]
- entity_description: Comprehensive
↪  description of the entity's attributes and
↪  activities
- entity_cognitiev_level: One of the following
↪  cognitive levels:
    Basic level: This level targets early
    ↪  childhood to elementary school
    ↪  learners. It focuses on providing
    ↪  simplified explanations and structured
    ↪  guidance to support foundational
    ↪  understanding and cognitive growth.
    Intermediate level: Geared towards middle
    ↪  and high school students, this level
    ↪  introduces more complex concepts and
    ↪  encourages moderate reasoning. It aims
    ↪  to bridge the gap between basic
    ↪  comprehension and advanced analytical
    ↪  skills, fostering critical thinking
    ↪  and problem-solving abilities.
    Advanced level: Designed for undergraduate
    ↪  students and beyond, this level
    ↪  explores complex, abstract concepts
    ↪  that require strong analytical skills.
    ↪  It challenges learners to engage with
    ↪  sophisticated ideas, promoting deep
    ↪  understanding and intellectual
    ↪  development.
  \end{itemize}
Format each entity as ("entity"⟨entity_name⟩
↪  ⟨entity_type⟩ ⟨entity_description⟩
↪  ⟨entity_cognitiev_level⟩)
2. From the entities identified in step 1,
↪  identify all pairs of (source_entity,
↪  target_entity) that are *clearly related*
↪  to each other.
For each pair of related entities, extract the
↪  following information:
- source_entity: name of the source entity, as
↪  identified in step 1
- target_entity: name of the target entity, as
↪  identified in step 1
- relationship_description: explanation as to
↪  why you think the source entity and the
↪  target entity are related to each other
- relationship_strength: a numeric score
↪  indicating strength of the relationship
↪  between the source entity and target entity
- relationship_keywords: one or more
↪  high-level key words that summarize the
↪  overarching nature of the relationship,
↪  focusing on concepts or themes rather than
↪  specific details
- relationship_cognitiev_level: A cognitive
↪  level indicating the cognitiev_level
↪  required to understand the relationship.
↪  This should be calculated by considering
↪  both the complexity of the relationship
↪  itself and the average cognitiev_level of
↪  the source and target entities. Use the
↪  following guideline:
    - Calculate the average cognitiev_level of
    ↪  the source and target entities.
    - Consider the inherent complexity of the
    ↪  relationship.
    - Assign a cognitiev_level level from the
    ↪  three cognitive levels.
Format each relationship as ("relationship"
↪  ⟨source_entity⟩ ⟨target_entity⟩
↪  ⟨relationship_description⟩
↪  ⟨relationship_keywords⟩
↪  ⟨relationship_strength⟩
↪  ⟨relationship_cognitiev_level⟩)
```

```
3. Identify high-level key words that summarize
↪  the main concepts, themes, or topics of
↪  the entire text. These should capture the
↪  overarching ideas present in the document.
Format the content-level key words as
↪  ("content_keywords"<high_level_keywords>)
4. Return output in {language} as a single
↪  list of all the entities and relationships
↪  identified in steps 1 and 2. Use
↪  **{record_delimiter}** as the list
↪  delimiter.
5. When finished, output
↪  {completion_delimiter}
```

C.2 Multi-layer Knowledge Graph Construction Algorithm

Algorithm 1 Adaptive Retrieval Graph Construction

```
Require: Text corpus $D$, maximum level $L$
Ensure: Adaptive Retrieval Graph $G_{total}$
$\mathcal{E}, \mathcal{R} \leftarrow$ Extration($D$)
for each $e_i \in \mathcal{E}$ do
    $l_i \leftarrow$ HierarchyAssigner($e_i$)
end for
for each $(e_i, r, e_j) \in (\mathcal{E}, \mathcal{R})$ do
    if $l_i = l_j$ then
        $G_{total}$.add_edge($e_i, e_j, r$)
    end if
    if $|l_i - l_j| \leq 1$ and $l_i \neq l_j$ then
        $G_{total}$.add_crosslink($e_i, e_j$)
    end if
end for
return $G_{total}$
```

C.3 Hierarchical Knowledge Retrieval

Algorithm 2 Hierarchical Knowledge Retrieval

```
Require: Query $q$, cognitive level $c \in \{0, 1, 2\}$,
    graph $G_{total}$, parameters top-k $k$, depth $d$
Ensure: Knowledge subset $K_c^{(k)}$
Set maximum level based on cognitive level:
$l_{max} \leftarrow c$
$G_c \leftarrow \{e \in G_{total} \mid l_e \leq l_{max}\}$
$\phi_q \leftarrow$ QueryRewriter($q, c$)
Initialize result set $K_c^{(k)} \leftarrow \emptyset$
Perform initial query traversal:
$S_k \leftarrow$ TopK(Neighbor($\phi_q, G_c$), $k$)
Add results to $K_c^{(k)}$: $K_c^{(k)} \leftarrow K_c^{(k)} \cup S_k$
for each $e_i \in S_k$ do
    Retrieve neighbors at depth $d + 1$: $N_i \leftarrow$
        NeighborDepth($e_i, G_c, d + 1$)
    Add neighbors to $K_c^{(k)}$: $K_c^{(k)} \leftarrow K_c^{(k)} \cup N_i$
end for
return $K_c^{(k)}$
```

D Implementation Details.

The experiments were conducted on a cluster with 8×NVIDIA A100 80GB GPUs, utilizing BF16 mixed precision and FlashAttention-2 for computational efficiency. The specific configurations are as follows: (1) Hierarchical Retrieval-Augmented Generation: We extracted a knowledge graph comprising 6,244 entities and 6,364 relations by setting the chunk token size to 600 and the chunk overlap token size to 100; (2) DPO Training: Using Llama3.1-8B-Instruct as the base model, we first performed 1 epoch of Supervised Fine-Tuning (SFT) with a global batch size of 64 (micro batch size of 16) and a learning rate of 5e-6, followed by 1 epoch of DPO with a learning rate of 5e-7 and beta=0.1, maintaining the same batch configuration as in SFT; (3) KCG: We extracted Control Center features from the DPO-trained model and implemented dynamic parameter control.

E Metrics

Prompt Template:

```
You are tasked with evaluating how well a
↪  given response matches the intended
↪  audience level. Consider the following
↪  audience types and their criteria:


Basic level: This level targets early
↪  childhood to elementary school learners.
↪  It focuses on providing simplified
↪  explanations and structured guidance to
↪  support foundational understanding and
↪  cognitive growth.
Intermediate level: Geared towards middle and
↪  high school students, this level
↪  introduces more complex concepts and
↪  encourages moderate reasoning. It aims to
↪  bridge the gap between basic comprehension
↪  and advanced analytical skills, fostering
↪  critical thinking and problem-solving
↪  abilities.
Advanced level: Designed for undergraduate
↪  students and beyond, this level explores
↪  complex, abstract concepts that require
↪  strong analytical skills. It challenges
↪  learners to engage with sophisticated
↪  ideas, promoting deep understanding and
↪  intellectual development.

Audience Type 0 (Basic level):
Is the response fun and engaging?
```

```
Does it use simple knowledge points and avoid
↪  complex vocabulary?
Are analogies and metaphors used
↪  appropriately?
Is any difficult content explained in simple
↪  terms?

Audience Type 1 (Intermediate level):
Does the response provide normal knowledge
↪  points?
Is it based on common sense and easily
↪  understandable?
Audience Type 2 (Advanced level):
Is the response professional and detailed?
Does it use technical language appropriate for
↪  experts in the field?
Question: {question}

Answer for Audience Type 0: {answer_type_0}

Answer for Audience Type 1: {answer_type_1}

Answer for Audience Type 2: {answer_type_2}

Evaluate each response based on how well it
↪  aligns with the specified audience type
↪  and provide a score out of 100 for each.
↪  Output only the scores as numbers.
```

F Dataset Construction Validation

F.1 Terminology Mapping Validation

- **Expert Panel**: Three biochemistry graduate researchers assessed the conceptual accuracy and complexity of mapped terms.
- **Consensus Mechanism**: Terms were refined collaboratively if inconsistencies were flagged.
- **User Testing**: Clarity and accessibility were validated through user feedback.

F.2 Educational Suitability Review

- **Panel**: Five education specialists in curriculum design assessed responses across basic, intermediate, and advanced tiers.
- **Criteria**: Each response was graded for age-appropriateness and conceptual clarity.
- **Revision**: 278 out of 593 samples were iteratively improved (46.8% revision rate).

G Details of Human Assessment

G.1 Expert Evaluation

Three graduate-level education specialists evaluated responses across all three difficulty levels using a 5-point Likert scale, where a higher score indicates better performance. They assessed:

- **Fluency**: Language clarity and logical structure.
- **Cognitive Level Alignment**: Appropriateness for target users.
- **Pedagogical Guidance**: Educational effectiveness.

A total of 100 questions were evaluated.

G.2 Users Evaluation

To assess real-world effectiveness, we recruited:

- **Basic Level** Three elementary school students.
- **Advanced Level** Three graduate students.

Participants compared responses from three systems (CLAF, ChatGPT-4o, and an SFT-tuned model) and selected their preferred answers.

H Detail of Personnel and Computational Cost

H.1 Human Resource Compensation

To support the evaluation and dataset construction processes, we involved several participants from both education and biochemistry domains. All participants were compensated accordingly. For international clarity, compensation amounts are approximated in U.S. dollars.

During the expert evaluation phase, we recruited three graduate-level education specialists. Each specialist was compensated approximately $60 for one and a half days of participation.

In the users evaluation stage, we involved three elementary school students and three graduate students in biochemistry. The graduate students received approximately $30 each, while the elementary students were compensated about $20 per participant, all for one day of participation.

During the dataset construction process, three biochemistry graduate students helped validate the terminology mapping. Each received around $40 for two days of work.

Finally, five graduate students specializing in education were recruited for the final dataset review. Each participant was compensated approximately $40 for two days of evaluation work.

H.2 Computational Efficiency

Despite the layered architecture of our framework, the system remains computationally efficient and scalable.

The construction of the entire domain-specific knowledge graph, consisting of over 6,000 entities and relations, took less than ten minutes and cost approximately $1 using a commercially available API. This demonstrates the feasibility of rapid knowledge graph generation.

CLAF operates with computational requirements similar to those of the LLaMA-3.1-8B model, requiring around 20GB of VRAM. Further optimization through quantization can reduce these requirements, enabling deployment on standard hardware.

The CAR multi-step retrieval module is highly efficient, achieving an average retrieval time of approximately 2 seconds per query, which supports real-time interactive use without noticeable latency for end-users.

These cost and performance metrics confirm that our framework can be practically adopted in large-scale educational settings without significant human or computational overhead.