JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

Cog-CoT: A Cognitive Chain-of-Thought Framework for Bloom's Taxonomy-Aligned Educational Question Answering

Alfarabi Muzli, Ratih Nur Esti Anggraini *, and Diana Purwitasari

Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia;
e-mail : 6025241061@student.its.ac.id; ratih@its.ac.id; diana@its.ac.id
* Corresponding Author : Ratih Nur Esti Anggraini

Abstract: Large Language Models (LLMs) have shown strong potential in educational question answering, yet they often exhibit cognitive misalignment by generating responses that emphasize linguistic fluency rather than the cognitive depth required by different levels of Bloom's Taxonomy. This limitation arises from three structural gaps: Bloom classifiers are typically disconnected from the generation process, Chain-of-Thought (CoT) reasoning lacks explicit cognitive scaffolding, and Retrieval-Augmented Generation (RAG) verifies factual consistency only at the final output. To address these limitations, this paper proposes Cognitive Chain-of-Thought (Cog-CoT), a four-module framework that integrates Bloom's Taxonomy into the reasoning process of LLMs. The framework consists of (1) a LinearSVC-based Cognitive Classifier (weighted F1 = 0.9915) for Bloom-level prediction, (2) a Hierarchical Cognitive Decomposer that constructs a bottom-up sequence of Bloom-aligned sub-questions, (3) a Cog-CoT Reasoning module employing six level-specific prompt templates with step-wise cosine similarity verification against retrieved context (θ = 0.65, MAX_RETRY = 3), and (4) a Bottom-Up Aggregator that synthesizes verified intermediate responses into a coherent final answer. Experiments using three open-weight LLMs (Gemma-3-4B-IT, LLaMA-3.1-8B-Instruct, and Qwen2.5-7B-Instruct) on an Indonesian Social Studies dataset and external English benchmarks (SQuAD 2.0 and ASQA) demonstrate that Cog-CoT consistently outperforms Zero-Shot prompting, achieving improvements of 4.16%–7.36% in GT Cosine Similarity while also producing superior performance across BLEU-4, ROUGE-L, METEOR, and BERTScore. These results demonstrate the effectiveness of integrating cognitive scaffolding with structured reasoning and retrieval verification for educational question answering.

Keywords: Bloom's Taxonomy; Chain-of-Thought Prompting; Cognitive Scaffolding; Educational Question Answering; Large Language Models; Retrieval-Augmented Generation; Prompt Engineering; Cognitive Reasoning.

Received: July, 2nd 2026
Revised: July, 30th 2026
Accepted: July, 31st 2026
Published: August, 12th 2026

1. Introduction

Bloom's Taxonomy partitions cognitive demand into six levels: Remember (C1), Understand (C2), Apply (C3), Analyze (C4), Evaluate (C5), and Create (C6). These levels span a continuum from recall-oriented lower-order thinking skills (LOTS) to integrative higher-order thinking skills (HOTS), as illustrated in Figure 1 [1]. For decades, this hierarchy has guided educators in designing questions, developing assessment rubrics, and sequencing curricula. Its well-defined cognitive structure makes it particularly attractive for automation. If a machine could approximate the cognitive demand expected of a learner and calibrate its responses to support the corresponding level of understanding, it could serve as an effective pedagogical assistant at scale [2]. Recent studies have further demonstrated that open-weight LLMs can be integrated into real-time educational feedback systems and combined with motivational scaffolding to support personalized learning in higher education [3].

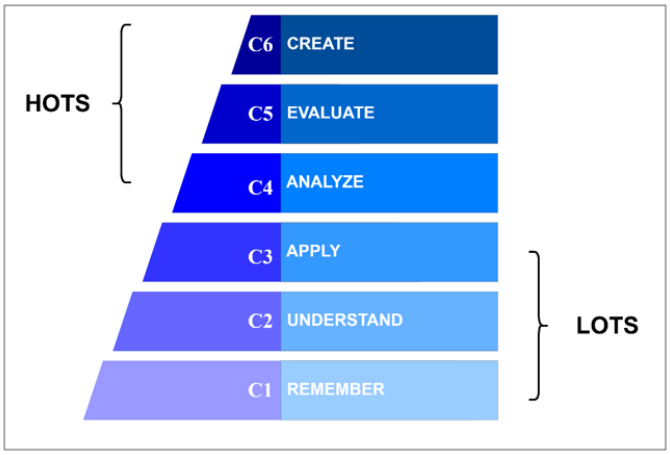

Figure 1. Bloom's Taxonomy hierarchy.

Despite these advances, present-day LLMs consistently fall short of this goal. Regardless of whether a question requires a simple definition (C1) or a multi-perspective critique (C5), these models tend to generate responses with similar structural complexity, typically favoring extended analytical prose. This mismatch between the intended cognitive level of an assessment and the cognitive depth of the generated response, referred to as Cognitive Misalignment, has been empirically observed in both medical [4] and engineering examination datasets [5]. However, existing generative frameworks have not yet addressed this issue in a systematic manner.

Three interrelated technical gaps contribute to this limitation. First, although Bloom-level classifiers now achieve weighted F1 scores exceeding 0.96 [6], [7], they are generally used only for classification. The predicted cognitive level is discarded after evaluation and does not influence the subsequent response generation process. Second, Chain-of-Thought (CoT) prompting [8] encourages multi-step reasoning but does not impose an explicit cognitive progression across reasoning steps. Without structured cognitive scaffolding, the model may shift between different cognitive levels within a single response. Structured CoT variants, such as OSCOT [9], partially address this limitation by prescribing domain-specific reasoning sequences; however, they require manual redesign for different domains and cannot naturally generalize across Bloom's cognitive hierarchy. Third, Retrieval-Augmented Generation (RAG) grounds the final response using retrieved documents [10], but it provides no mechanism to verify intermediate reasoning steps before the final answer is produced. Consequently, reasoning may gradually drift away from the retrieved evidence, while existing single-output verification approaches developed for medical question answering [11] are not designed to preserve cognitive consistency throughout the reasoning process.

To address these limitations, this paper proposes Cognitive Chain-of-Thought (Cog-CoT), a four-module framework that integrates Bloom's Taxonomy directly into the reasoning process of LLMs. A Cognitive Classifier first routes each question to its corresponding Bloom level. A Hierarchical Cognitive Decomposer then constructs a bottom-up sequence of sub-questions from C1 to the target cognitive level, thereby enforcing cognitive completeness throughout the reasoning process. Next, the Cog-CoT Reasoning module applies a level-specific prompt template and verifies each intermediate reasoning step using cosine similarity against the retrieved RAG context before proceeding. Finally, a Bottom-Up Aggregator synthesizes all verified intermediate responses into a coherent answer aligned with the intended cognitive level.

The proposed framework makes four primary contributions. (1) It is the first framework to directly incorporate Bloom-level predictions into the reasoning structure of LLMs rather than treating classification as a standalone task. (2) It introduces six level-specific Cog-CoT prompt templates (C1–C6), each explicitly designed to capture the cognitive operations associated with a particular Bloom level. (3) It proposes a step-wise self-verification mechanism based on cosine similarity with retrieved RAG context, combined with a retry strategy to reduce factual drift before errors accumulate. (4) It provides empirical evidence and a mechanistic explanation of CoT performance degradation, demonstrating that unstructured CoT

reasoning consistently reduces answer quality in RAG-based educational question answering across three architecturally distinct LLMs and two datasets in different languages.

The remainder of this paper is organized as follows. Section 2 reviews the four research areas most closely related to this work. Section 3 presents the proposed Cog-CoT framework. Section 4 describes the experimental setup and discusses the results. Section 5 compares Cog-CoT with representative existing approaches. Finally, Section 6 concludes the paper and outlines its limitations and future research directions.

2. Related Work

2.1. Bloom's Taxonomy in Automated Question Answering

The cognitive hierarchy articulated by Bloom provides a structural vocabulary for distinguishing recall from reasoning in educational assessment [1]. When applied to automated question answering (QA), it prescribes not only the subject matter of an answer but also its epistemic posture: a C1 response enumerates facts, whereas a C5 response marshals evidence to support a judgment [12]. Benchmark studies evaluating GPT-family models on Bloom-structured examination datasets consistently find that these models generate responses with largely uniform cognitive depth regardless of the intended cognitive level of the prompt [4], [5]. This pattern has been attributed to training objectives that prioritize surface fluency over the nuanced cognitive calibration required for structured pedagogical assessment. Several studies have leveraged Bloom's Taxonomy for learning-objective mapping and course design automation [13], [14]; however, none has directly integrated the taxonomy into LLM reasoning control. Cog-CoT treats Bloom's Taxonomy not merely as a post hoc evaluation framework but as an operational scaffold that governs the reasoning process from the moment a question is received.

2.2. Chain-of-Thought Prompting and Its Limitations

Wei et al.'s seminal work on Chain-of-Thought (CoT) prompting [8] demonstrated that eliciting step-by-step reasoning substantially improves LLM performance on arithmetic and commonsense reasoning tasks. Since then, CoT has been adapted for educational scoring [15], causal science [9], and hierarchical QA decomposition [16]. Ensemble-based decoding strategies, such as self-consistency, improve the reliability of CoT reasoning but substantially increase inference costs without addressing cognitive misalignment [17]. Similarly, branching search methods that explore tree-structured reasoning paths improve performance on well-defined reasoning tasks but still lack an explicit cognitive hierarchy [18]. Zhang et al. [16] further extend hierarchical reasoning through the TreeQA framework, which achieves 4–12% improvements over state-of-the-art RAG methods on multi-hop QA benchmarks. However, like previous CoT variants, TreeQA does not perform per-step factual verification against retrieved context. This paper documents, for the first time, that such unchecked reasoning verbosity can completely negate the benefits of CoT in RAG-based question answering. This phenomenon, referred to as CoT performance degradation, is consistently observed across three LLM families and two datasets.

2.3. Retrieval-Augmented Generation for Factual Answer Generation

Retrieval-Augmented Generation (RAG) decouples world knowledge from generative reasoning by retrieving relevant passages at inference time, thereby substantially reducing hallucinations across domain-specific QA tasks [10]. Educational applications have particularly benefited from this paradigm. A recent systematic survey of RAG-based learning systems reported consistent improvements in factual accuracy for both short-answer and long-form educational questions [19]. For Indonesian-language open-domain QA, combining RAG with ReAct prompting reduces unsupported assertions and improves contextual coherence [20]. More recently, Adaptive-RAG strategies [21], which dynamically adjust retrieval depth according to query complexity, have demonstrated promising results, although they have not been integrated with cognitive scaffolding. Despite these advances, retrieval only constrains the final output; intermediate reasoning steps are not explicitly required to remain grounded in the retrieved evidence. Chen et al. [11] proposed output-level cosine similarity verification for Chinese medical relation extraction, but their approach cannot detect factual drift

occurring during intermediate reasoning. Cog-CoT extends this verification principle by validating every intermediate reasoning step.

2.4. Cognitive Classification as Preprocessing

Machine-learning approaches to Bloom-level classification have evolved steadily over time. Early keyword-frequency methods were gradually replaced by TF-IDF + SVM pipelines [6], followed by LSTM models with pretrained embeddings [22], and more recently by fine-tuned Transformer classifiers [7] and generative-AI ensemble approaches [23], achieving weighted F1 scores ranging from 0.94 to 0.96. Bloom-level classification for Indonesian-language educational content has also received increasing attention, with recent IndoBERT-based approaches substantially outperforming earlier monolingual baselines on national curriculum examination questions. Recent generative-AI ensemble methods [23] have further improved classification accuracy. Nevertheless, throughout this line of research, classification accuracy has remained the final objective, and no published framework has utilized the predicted Bloom level to guide downstream generative reasoning. Chindukuri and Sivanesan [6] explicitly identify this integration as an important open research problem. Cog-CoT addresses this gap by using the classifier's output as a pedagogical routing signal to guide sub-question decomposition, prompt template selection, and response aggregation.

2.5. Positioning of Cog-CoT Among Related Frameworks

Table 1 summarizes how Cog-CoT differs from the five most closely related frameworks. Standard Chain-of-Thought (CoT) introduces step-by-step reasoning but provides neither explicit cognitive ordering nor a factual grounding mechanism, allowing reasoning chains to drift from the retrieved evidence. Retrieval-Augmented Generation (RAG)-based QA systems ground the final response using retrieved documents but do not verify intermediate reasoning steps. TreeQA [16] introduces hierarchical logic-tree decomposition and achieves strong performance on multi-hop question answering; however, its decomposition is driven by logical dependencies rather than pedagogically grounded cognitive structures, and it does not perform per-step verification. ReAct [20] interleaves reasoning with retrieval but does not organize reasoning according to an explicit cognitive hierarchy. Adaptive-RAG [21] dynamically adjusts retrieval depth based on query complexity but lacks both cognitive scaffolding and intermediate verification. Cog-CoT is the first framework to integrate all three missing components within a unified pipeline: (1) a Bloom-level classifier whose output directly guides the reasoning process, (2) a hierarchically ordered chain of sub-questions that enforces cognitive progression from C1 to the target Bloom level, and (3) step-wise cosine similarity verification against RAG context with a retry mechanism that prevents factual drift before it propagates to the final response.

Table 1. Comparison of Cog-CoT with Related Frameworks

Framework	Bloom Routing	Cognitive Chain	Per-Step Verification
Standard CoT	✗	✗	✗
RAG-based QA	✗	✗	✗
TreeQA [16]	✗	Partial	✗
ReAct [20]	✗	✗	Partial
Adaptive-RAG [21]	✗	✗	✗
Cog-CoT (Proposed)	✓	✓	✓

3. Proposed Method

Fig. 2 illustrates the complete Cog-CoT pipeline. Each module receives its input directly from the preceding module, and no human intervention is required between processing stages.

3.1. Dataset

Four datasets were utilized in this study: two primary datasets assembled from Indonesian Social Studies (IPS) textbooks spanning Grades VII–XII (Kemdikdasmen) and two

external English benchmark datasets. IPS was selected because its content naturally spans all six Bloom levels, ranging from recalling historical dates (C1) to proposing civic solutions (C6). In addition, its morphologically rich Indonesian language provides a challenging setting for cosine similarity-based verification.

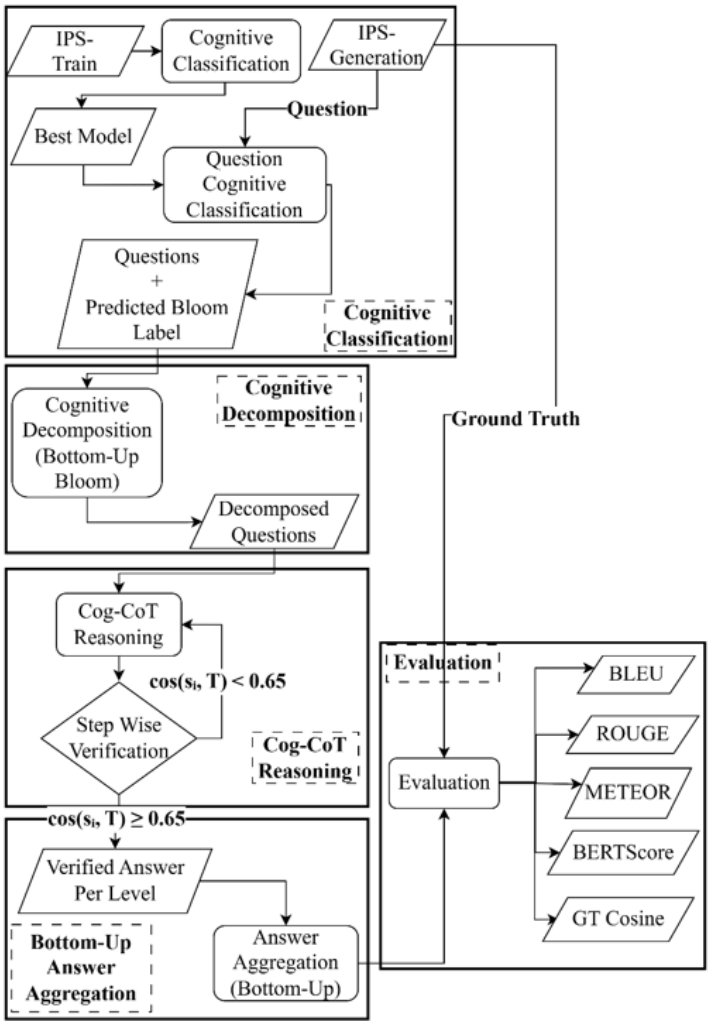

Figure 2. Cog-CoT Flowchart.

IPS-Train (5,190 samples; 865 samples per Bloom level) consists of question–context pairs explicitly annotated with their corresponding Bloom's Taxonomy level (C1–C6). The context provides the reading passage associated with each question, and these labeled pairs are used exclusively to train the Cognitive Classifier. IPS-Generation (120 samples; 20 samples per Bloom level) contains question–context–ground-truth triples, with reference answers written by practicing IPS teachers. This dataset serves both as the RAG knowledge base and as the benchmark for end-to-end pipeline evaluation. To evaluate cross-lingual and cross-domain generalization, two external English benchmark datasets were constructed from SQuAD 2.0 [24] (60 samples; mean ground-truth length of 6.5 words) and ASQA [25] (60 samples; mean ground-truth length of 63 words), intentionally combining short factual answers with longer explanatory responses. Table 2 summarizes the statistics of all four datasets.

The Bloom-level annotations for IPS-Train and the teacher-authored ground-truth answers for IPS-Generation were produced by three annotators, all of whom are practicing IPS (Social Studies) teachers. Two annotators have more than ten years of classroom teaching experience, while one has fewer than five years of experience. The annotation task was divided equally among the three annotators, with each annotator responsible for 40 distinct questions, ensuring complete coverage of all 120 evaluation items by subject-matter experts. Both IPS-Train and IPS-Generation will be made available upon reasonable request to the corresponding author.

Table 2. Dataset Statistics

Dataset	Samples	Bloom Distribution	Avg. GT Length
IPS-Train	5,190	865 per level (C1–C6)	—
IPS-Generation	120	20 per level (C1–C6)	161
SQuAD 2.0	60	Majority C1/C3	6.5
ASQA	60	Majority C1/C3	63.3

3.2. Module 1: Cognitive Classification

The Cognitive Classifier maps each raw question, Q, to its corresponding Bloom level. This predicted level subsequently determines the number of generated sub-questions, the prompt template to be applied, and the depth of the aggregation chain. Text preprocessing converts all characters to lowercase, removes punctuation and numerical digits, and normalizes whitespace. Five classifiers were trained using a stratified 80/20 split of IPS-Train and evaluated using the weighted F1 score. The routing function is defined in Eq. (1).

$$\text{Level} = \text{Classifier}\big(\text{preprocess}(Q)\big) \tag{1}$$

where $Level \in \{C_1, C_2, C_3, C_4, C_5, C_6\}$.

To determine the optimal routing function, five candidate classifiers—LinearSVC, Logistic Regression, SGD, Random Forest, and Naïve Bayes—were trained and evaluated using the weighted F1 metric. Table 3 presents an example illustrating how a sample IPS question is preprocessed and routed through Module 1.

Table 3. Example of Cognitive Classification Routing for a C5 Question

Stage	Value
Raw Input Question	Which has a greater impact on the daily economic activities of Indonesian people: astronomical position (time differences) or climate conditions (two seasons)? Provide your justification.
Preprocessed Question	which has a greater impact on daily economic activities of indonesian people astronomical position time differences or climate conditions two seasons provide your justification
Predicted Level	$C5$ – Evaluate
Routing: n (depth)	5 sub-questions ($C1 \rightarrow C2 \rightarrow C3 \rightarrow C4 \rightarrow C5$)

3.3. Module 2: Hierarchical Cognitive Decomposition

Given Q and target level C_n, the Decomposer generates an ordered set of n sub-questions such that sub-question q_i targets Bloom level c_i and depends semantically on the answer to q_{i-1}, as formalized in Eq. (2).

$$D(Q, C_n) = \{(q_i, c_i) \mid i = 1, 2, \ldots, n\}, n \leq 6 \tag{2}$$

This bottom-up chaining property guarantees cognitive completeness: every level from C_1 to C_n is addressed before reaching the target level, and no level is skipped. The LLM generates sub-questions at temperature $T = 0.0$ with paired correct/incorrect few-shot examples. If the LLM output fails schema validation, a fallback substitutes simplified reformulations of Q at each missing level. Algorithm 1 formalizes this procedure.

The fallback procedure simplify (Q, level=c_i) is triggered when the LLM output fails schema validation, that is, when the number of parsed sub-questions does not match the number of required Bloom levels. In the implementation, the fallback constructs a replacement sub-question for each missing level ci by prepending a level-specific cognitive stem to the original question Q, producing the string "Sub-question c_i: Q" for each absent level. This preserves the original question's topic while signaling the intended Bloom operation, ensuring the pipeline always receives a structurally complete chain of n sub-questions. This strategy deliberately accepts a lower-quality decomposition at failed levels rather than aborting the pipeline, consistent with the principle that the bottom-up chain must be complete even when not semantically optimal.

Algorithm 1. Hierarchical Cognitive Decomposition

INPUT: Q, classified_level
OUTPUT: Sub-question chain$\{(q_i, c_i) \mid i = 1, 2, \ldots, n\}$
prompt ← build_decomposition_prompt (Q, classified_level, few_shot_examples)
raw_output ← LLM (prompt, T=0.0)
question ← parse_subquestions (raw_output)
IF $|\text{questions}| < n$ THEN
    FOR i = $|\text{questions}|$ + 1 TO n DO
        questions[i] ← simplify (Q, level=c_i) # FallBack
    END FOR
RETURN {(questions[i], c_i) | i = 1…n}

The complete decomposition output for the C5 sample question introduced in Table 3 follows this procedure. The chain progresses strictly bottom-up; each sub-question at level c_i assumes the answer at c_{i-1} is already established.

3.4. Module 3: Cog-CoT Reasoning with Step-Wise Verification

For each sub-question (q_i, c_i), the reasoning module applies the level-specific template shown in Table 4, invokes the LLM to produce a step-by-step response, and then verifies each intermediate reasoning step s_j against RAG-retrieved context T using cosine similarity calculated by Eq. (3).

$$\cos(s_j, T) = \frac{\text{emb}(s_j) \cdot \text{emb}(T)}{(\text{emb}\|s_j\| \times \|\text{emb}(T)\|)} \tag{3}$$

A step is assigned PASS if $\cos(s_j, T) \geq \theta = 0.65$ and FAIL otherwise. The threshold $\theta = 0.65$ sits at the boundary between moderate and strong semantic overlap, calibrated to tolerate Indonesian's affixation-induced lexical variation without admitting steps that have genuinely departed from the retrieved context [26]. The final answer step is exempted from verification since it is necessarily generative. If any step fails, the LLM is re-invoked up to $MAX_{RETRY} = 3$. This structured grounding mechanism also provides a degree of robustness against adversarial prompt manipulations that exploit LLMs' inability to reliably separate instructions from retrieved data [27]. Algorithm 2 specifies the full reasoning and verification loop.

Table 4. Level-specific reasoning steps.

Level	Cognitive Operation	Reasoning Steps
C1	Fact recall	Scan → Extract → State
C2	Interpretation	Identify → Describe → Explain → Summarise
C3	Procedure application	Identify Rule → Map Case → Execute → Conclude
C4	Deconstruction	Deconstruct → Examine → Relate → Synthesise
C5	Judgment	Set Criteria → Gather Evidence → Judge → Conclude
C6	Synthesis	Identify Elements → Formulate Plan → Generate → Review

Algorithm 2. Cog-CoT Reasoning

INPUT: Sub-question q_i, Level c_i, RAG Context T, template
OUTPUT: best_verified_answer a_i^*
retry_attempt ← {} # set of retry attempts
FOR k = 1 TO MAX_{RETRY} DO
    prompt ← template (q_i, T) # level-specific prompt + context
    response ← LLM (prompt, T=0.0)
    steps ← extract_reasoning_steps(response)
    scores ← []
    FOR EACH step s_j IN steps (excluding final answer step) DO
        score ← cos(emb(s_j, emb(T))
        IF score ≥ θ THEN mark s_j as PASS ELSE mark s_j as FAIL

Algorithm 2 cont.

```
        scores.append(score)
    END FOR
    mean_score ← mean(scores)
    retry_attempt ← retry_attempt ∪ {(response, mean_score)}
    IF ALL steps PASS THEN BREAK
END FOR
a_i^* ← response with highest mean_score in retry_attempt
RETURN a_i^*
```

Table 5 traces the step-wise verification process for sub-question $q_1(C1)$ from the running example, showing per-step cosine scores against the RAG-retrieved context passage.

Table 5. Example: Step-wise verification for sub-question q_1.

Step	Reasoning Step Context (truncated)	Cosine Score	Status
Step 1 – Scan	The context discusses Indonesia's geographical position spanning three time zones (WIB, WITA, WIT) due to its astronomical location between 6°N–11°S and 95°E–141°E, and a tropical climate with two seasons: the rainy season (October–March) and the dry season (April–September)...	0.821	PASS
Step 2 – Extract	Indonesia's astronomical position is defined by its latitude and longitude coordinates, resulting in three time zones. At the same time, its tropical climate produces two seasons that directly shape agricultural cycles, traditional industries, and livelihood patterns across the archipelago...	0.793	PASS
Step 3 – State	Indonesia's astronomical position spans three time zones, which affect daily scheduling. At the same time, its two-season tropical climate determines when crops are planted and harvested, directly influencing the income of millions of farmers and the output of season-dependent industries...	0.748	PASS
Final Answer	(exempt from verification – generative synthesis) Mean cosine (verified steps): 0.787 → Attempt accepted (no retry needed)	—	EXEMPT

3.5. Module 4: Bottom-Up Answer Aggregation

Once all n verified sub-answers $\{a_1^*, \ldots, a_n^*\}$ have been obtained, the Aggregator synthesizes them into a single response A^* that answers the original question Q at the target cognitive level C_n, as defined in Eq. (4).

$$A^* = \text{LLM}_{\text{agg}}(\{a_i^* | i = 1, \ldots, n\}, Q, C_n), n \leq 6 \tag{4}$$

The aggregation prompt instructs the LLM to generate a single coherent paragraph that (1) follows a bottom-up cognitive progression ($C1$ facts first and C_n reasoning last), (2) directly answers the original question Q without exceeding the cognitive scope of C_n, and (3) avoids introducing knowledge that is absent from the verified sub-answers. The aggregation temperature is fixed at 0.0 to maximize determinism and preserve consistency with the verified reasoning chain. For $C1$ questions, a_i^* is returned directly, bypassing the aggregation step. Algorithm 3 summarizes the aggregation procedure.

Algorithm 3. Bottom-Up Answer Aggregation

```
INPUT: verified sub answers {a_1^*,…, a_n^*}, original question Q, level C_n
OUTPUT: Final coherent answer A^*
IF n = 1 THEN
    RETURN a_1^* # direct return from C1
prompt ← build_aggregation_promtp (Q, C_n, {a_i^* | i = 1,…,n})
# instruct: bottom-up structure, direct answer, no scope excess
A^* ← LLM (prompt, T=0.0)
RETURN A^*
```

3.6. Experimental Setup

All experiments were conducted using the Hugging Face Transformers library with bfloat16 precision. Three open-weight LLMs—Gemma-3-4B-IT, LLaMA-3.1-8B-Instruct, and Qwen2.5-7B-Instruct—were evaluated under three experimental conditions:

1. Zero-Shot: Direct question answering without retrieved context or Chain-of-Thought prompting.
2. Standard CoT: Question answering with the instruction "Think step by step" but without Bloom-based cognitive scaffolding or Retrieval-Augmented Generation.
3. Cog-CoT: The complete proposed pipeline.

All three conditions used identical input data, making the prompting strategy the only independent experimental variable.

All five Bloom classifiers were implemented using scikit-learn. The TF-IDF vectorizer employed bigram features with ngram_range = (1,2). LinearSVC, Logistic Regression, and SGD used class_weight = "balanced" and max_iter = 2000, while LinearSVC additionally used dual = True. Random Forest used n_estimators = 100, and all classifiers were trained with random_state = 42. The dataset was partitioned using an 80:20 stratified train-test split with random_state = 42. Retrieval employed a TF-IDF vectorizer with the default scikit-learn configuration, fitted on the unique context passages in the evaluation dataset. During inference, the top-k=3 most similar passages were retrieved using cosine similarity between the input query and the indexed context vectors. Retrieval was purely lexical and did not employ a neural retriever. The retrieved passages were concatenated and supplied as context to the reasoning module. Both the step-wise verification process and the final semantic evaluation employed the paraphrase-multilingual-MiniLM-L12-v2 sentence encoder, which produces 384-dimensional sentence embeddings and supports both Indonesian and English.

All LLMs were loaded using Hugging Face Transformers with device_map = "auto" and bfloat16 precision. Gemma-3-4B-IT specifically required bfloat16 inference. Both the decomposition and reasoning modules used temperature = 0.0, while the aggregation module also used temperature = 0.0. A repetition_penalty = 1.1 was applied to prevent repetitive generations, particularly for Gemma. The maximum number of generated tokens was set to 2048. Retrieved context passages were passed directly without chunking, and tokenizer input length was capped at max_length = 4096 with truncation enabled.

Performance was evaluated using Ground Truth (GT) Cosine Similarity, which measures the semantic similarity between the generated answer and the teacher-authored reference using the cosine similarity function defined in Eq. (3), together with BLEU-4, ROUGE-L, METEOR, and BERTScore. To assess the robustness of the proposed verification mechanism, a sensitivity analysis was performed by varying the verification threshold $\theta \in \{0.55,0.60,0.65,0.70\}$ using a stratified subset of the IPS-Generation dataset.

4. Results and Discussion

4.1. Cognitive Classification Results

Table 6 reports classifier performance on the IPS-Train test set across all five candidate models. LinearSVC achieved a weighted F1 of 0.9915, the highest among all candidates, with only 9 misclassifications across 1,038 test samples. Critically, all nine errors fell between adjacent Bloom levels (C2↔C3 or C4↔C5), confirming that the classifier never confuses LOTS with HOTS.

Table 6. Comparative performance of cognitive classifiers.

Model	Accuracy	Precision	Recall	Weighted F1
Naïve Bayes	87.93	0.8851	0.8793	0.8805
Logistic Regression	98.33	0.9834	0.9833	0.9833
SGD	98.70	0.9871	0.9870	0.9870
Random Forest	96.10	0.9612	0.9610	0.9610
LinearSVC	**99.15**	**0.9916**	**0.9915**	**0.9915**

4.2. End-to-End Generative Performance

To evaluate the overall generative performance of the proposed framework, the final aggregated answers were compared with the teacher-authored reference answers using five complementary evaluation metrics. Table 7 reports the complete experimental results for all

three prompting strategies (Zero-Shot, Standard CoT, and Cog-CoT) across the three evaluated LLMs.

Cog-CoT consistently achieves the highest GT Cosine Similarity for all evaluated models, reaching 0.8066 for Gemma-3-4B-IT, 0.8083 for LLaMA-3.1-8B-Instruct, and 0.8124 for Qwen2.5-7B-Instruct. Compared with their respective Zero-Shot baselines, these correspond to relative improvements of 6.90%, 4.16%, and 7.36%, respectively. Similar trends are observed for BERTScore, with improvements of 6.16%, 3.83%, and 6.80%, indicating that Cog-CoT consistently improves semantic alignment with the teacher-authored references.

The substantial gap between GT Cosine Similarity (approximately 0.81) and BLEU-4 (approximately 0.08) should not be interpreted as poor generation quality. Rather, it reflects the characteristics of the evaluation corpus. The IPS-Generation references are teacher-authored explanatory answers with an average length of 161 words, making exact n-gram overlap inherently difficult even when the generated responses are semantically correct. Consequently, embedding-based metrics, such as GT Cosine Similarity and BERTScore, provide a more reliable assessment of semantic quality than lexical overlap metrics (BLEU-4, ROUGE-L, and METEOR) for long-form educational question answering.

Table 7. End-to-End Evaluation Results

Model	Strategy	GT Cosine Similarity	BLEU-4	ROUGE-L	METEOR	BERTScore
Gemma-3-4B-IT	Zero-Shot	0.7546	0.0285	0.1564	0.2227	0.6915
	Standard CoT	0.6969	0.0362	0.1586	0.2497	0.6884
	Cog-CoT	**0.8066**	**0.0637**	**0.2276**	**0.2380**	**0.7341**
LLaMA-3.1-8B-Instruct	Zero-Shot	0.7760	0.0552	0.2155	0.2144	0.7105
	Standard CoT	0.7151	0.0456	0.1838	0.2182	0.6903
	Cog-CoT	**0.8083**	**0.0781**	**0.2427**	**0.2413**	**0.7377**
Qwen2.5-7B-Instruct	Zero-Shot	0.7567	0.0396	0.1874	0.2237	0.7002
	Standard CoT	0.6972	0.0421	0.1777	0.2384	0.6868
	Cog-CoT	**0.8124**	**0.0934**	**0.2682**	**0.2432**	**0.7478**

4.2.1. Comparison with Extended Baselines

Table 8 extends the main comparison with two additional structured reasoning baselines. ReAct achieves an overall GT Cosine Similarity of 0.7408, falling below both Zero-Shot (0.7546) and Cog-CoT (0.8066). The interleaved reasoning-and-retrieval loop of ReAct is effective for isolated factual lookups but imposes no cognitive structure on the reasoning chain; each action retrieves context independently without building on a hierarchically grounded prior. The sharpest performance drop occurs at C3-Menerapkan (GT Cosine: 0.6362), a level where multi-step application demands structured scaffolding that a flat action sequence cannot provide. Tree-of-Thought (ToT) achieves an overall GT Cosine Similarity of 0.8170, making it the strongest baseline and the closest competitor to Cog-CoT (0.8066). ToT's branching path exploration allows recovery from locally poor reasoning choices, yielding strong results at C1 (0.8841) and C5 (0.8587). However, ToT neither incorporates Bloom-level routing nor applies per-step RAG verification; its branching heuristic is driven by a general scoring function rather than pedagogical cognitive structure. This means the reasoning chain cannot guarantee coverage of prerequisite lower-order knowledge before addressing the target.

The METEOR scores further differentiate the strategies: ReAct (0.1394) falls below Zero-Shot (0.2227) because its shorter, action-oriented outputs are poorly aligned with the explanatory prose of the teacher-authored IPS-Generation ground truth. ToT (0.2812) is competitive with Standard CoT (0.2497) on METEOR, but both remain below Cog-CoT, whose aggregation step synthesises multi-level sub-task outputs into coherent explanatory answers structurally closer to the reference format. In summary, ToT is a strong general-purpose reasoning baseline, but Cog-CoT outperforms it on four of five metrics. The advantage is attributable to two structural properties absent from both ReAct and ToT: (1) Bloom-level cognitive routing that guarantees prerequisite knowledge is covered at every chain step, and (2) per-step RAG verification with retry logic that prevents factual drift from propagating to the final answer.

Table 8. Extended baseline comparisons.

Strategy	GT Cosine Similarity	BLEU-4	ROUGE-L	METEOR	BERTScore
Zero-Shot	0.7546	0.0285	0.1564	0.2227	0.6915
Standard CoT	0.6969	0.0362	0.1586	0.2497	0.6884
ReAct	0.7408	0.0282	0.1802	0.1394	0.7042
Tree-of-Thought	**0.8170**	**0.0857**	**0.2631**	**0.2812**	0.7315
Cog-CoT	0.8066	0.0637	0.2276	0.2380	**0.7341**

4.3. Qualitative Example: Bottom-Up Aggregated Answer

To ground the quantitative findings in a concrete case, Table 9 presents the aggregated answers produced by all three strategies for the C5 sample question introduced in Section 3.2–3.5, "Which has a greater impact on daily economic activities of Indonesian people: astronomical position (time differences) or climate conditions (two seasons)?", alongside the teacher-authored ground truth.

Table 9. Qualitative comparison of aggregated answers for the C5 sample question across all three strategies.

Strategy	Aggregated Answer
Ground Truth	Climate conditions with two seasons have a greater impact on the daily economic activities of Indonesian people than the astronomical position. First, the two-season climate (rainy and dry) directly and massively influences the agricultural sector, which is the livelihood of the majority of Indonesians. Planting times, crop types, and harvest yields are highly dependent on the seasons, so climate conditions determine the income of millions of farmers. Crop failure due to drought or flooding can cause enormous economic losses. Second, climate influences traditional industries such as salt, roof tiles, and crackers that depend on sunlight for production. Third, the rainy season increases the risk of hydrometeorological disasters that can halt economic activity and damage infrastructure. In contrast, astronomical position-based time differences (WIB, WITA, WIT) have a more limited impact. Time differences affect the coordination of national activities such as television broadcasts or remote meetings, but their impact is administrative and can be adapted through schedule adjustments. Time differences do not threaten production or cause material losses the way seasonal changes can.
Zero-Shot	Climate conditions (two seasons) have a greater impact on the daily economic activities of Indonesian people. Climate influences the agricultural sector, fisheries, and tourism industry through the effects of the rainy and dry seasons on production volumes, crop types, and prices of basic goods. Although astronomical position creates time-zone differences that affect international schedules, its impact on everyday economic life is structural and relatively minor compared with the direct, large-scale influence that seasonal changes exert on livelihoods and market supply. (**GT Cosine Similarity: 0.710**)
Standard CoT	Step 1: The question asks us to compare the impact of astronomical position versus climate conditions on daily economic activities. Step 2: The astronomical position creates three time zones (WIB, WITA, WIT), affecting work schedules, school hours, and business operations nationwide. Climate conditions produce two seasons: the rainy season (October–March), which drives high-water crops, and the dry season (April–September), suited to drought-tolerant plants, both of which shape planting calendars and harvests. Step 3: Considering both factors, climate conditions exert a broader and more direct influence on economic life because they determine agricultural output, affect traditional industries dependent on sunlight, and create disaster risks from flooding or drought. Time-zone differences, by contrast, are manageable through scheduling adjustments and do not threaten material production. Therefore, climate conditions have the greatest economic impact. (**GT Cosine Similarity: 0.665**)

The comparison illustrates how cognitive scaffolding shapes the structure and depth of responses beyond what metric scores alone convey. The Zero-Shot answer provides a reasonable surface-level response but lacks the systematic bottom-up reasoning structure. The Standard CoT answer adds intermediate steps but drifts from the RAG-retrieved context as verbosity increases, producing a response that is longer yet less semantically aligned to the ground truth (GT Cosine Similarity: 0.665 vs. Zero-Shot: 0.710). The Cog-CoT answer, by contrast, follows the bottom-up aggregation instruction: it opens with C1 factual grounding (astronomical position and time zones), builds through C2 interpretation (economic scheduling effects), C3 application (seasonal agricultural impact), and C4 analysis (comparative impacts), before arriving at the C5 evaluative judgment that climate conditions exert the greater economic influence. This progression mirrors the structure of the teacher-authored ground truth and yields the highest GT Cosine Similarity (0.810) among all three strategies.

4.4. Per-Level Bloom Analysis

Figure 3 traces GT Cosine Similarity by Bloom level across all strategies and models, and Table 10 presents the Cog-CoT results in numerical and heatmap form. The level-wise profile in Table 11 reveals three sub-patterns worth examining separately. The general descent from C1 (avg. 0.8710) through C3 (0.7998) is expected: longer sub-question chains accumulate small errors at each step, and C3 Apply tasks require correct procedural identification in addition to factual recall. The C4 recovery (0.8043 > 0.7998) is attributable to the Deconstruct → Examine → Relate → Synthesize template, whose explicit relational structure aligns closely with how IPS ground truths explain cause-effect and comparative relationships. C5 Evaluate ranks second overall (0.8312) because the Indonesian IPS corpus tends to embed explicit evaluative criteria directly in the context passage, making the Set Criteria → Gather Evidence → Judge template unusually effective at capturing ground-truth structure.

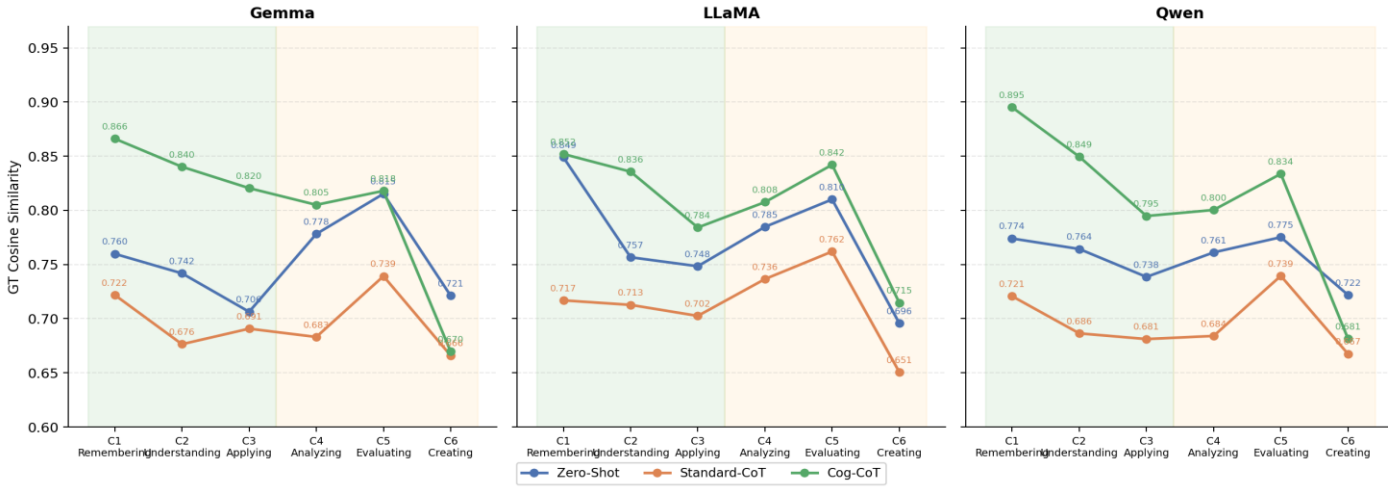

Figure 3. GT Cosine Similarity per Bloom level per strategy for each model

Table 10. GT Cosine Similarity per Bloom under Cog-CoT, averaged across three models.

Level	Gemma	LLaMA	Qwen	Average
C1	0.8661	0.8519	0.8951	0.8710
C2	0.8402	0.8356	0.8495	0.8418
C3	0.8205	0.7841	0.7947	0.7998
C4	0.8050	0.8075	0.8003	0.8043
C5	0.8180	0.8420	0.8337	0.8312
C6	0.6695	0.7146	0.6814	0.6885

For Gemma and Qwen, Cog-CoT GT Cosine Similarity at C6 falls below their respective Zero-Shot scores (Gemma: 0.6695; Qwen: 0.6814). Three compounding factors explain this. First, the verification threshold $\theta = 0.65$ is calibrated against retrieved context vocabulary; C6 Create tasks, however, require the model to synthesise novel formulations that may be

semantically sound yet lexically distant from the retrieved passage, causing steps to receive FAIL verdicts even when the content is pedagogically valid. Second, the six-level decomposition chain is the longest in the pipeline, so cumulative drift is magnified. Third, single-reference evaluation cannot credit a creative synthesis that departs from the teacher's particular wording while remaining semantically sound. Table 11 (C6 Verification Penalization Examples) illustrates these dynamics across three representative Qwen samples drawn from the IPS-Generation evaluation, showing how creative output is penalized despite semantic coherence.

Table 11. C6 Verification Penalization Examples.

Question	GT Cosine: Zero Shot	GT Cosine: Cog-CoT	Avg. Step Cosine	Steps FAIL/Total	Cause
Design a 3-day travel itinerary across Jakarta (WIB), Bali (WITA), and Raja Ampat (WIT), accounting for time-zone differences in flight schedules and communication.	0.8343	0.6286	0.291	5 / 5	Generated itinerary uses creative scheduling terminology not present in the retrieved context, causing all step cosines to fall well below θ=0.65 despite factual correctness.
Design a global campaign slogan to raise awareness about the limited availability of habitable land on Earth.	0.7506	0.5517	0.124	1 / 1	A single-step synthesis produces a creative slogan phrase with zero lexical overlap with the context passage, yielding cosine = 0.124 and triggering all MAX_{RETRY} attempts without improvement.
Design a scenario where a person can transition from "Not in the Labor Force" status to "Employed" status over six months.	0.9232	0.7903	0.321	1 / 1	The highest Zero-Shot score in the C6 set. Cog-CoT introduces a structured narrative with economic terminology diverging from the specific phrasing of the context, penalizing an otherwise strong response.

As shown in Table 11, all three cases share a common pattern: the Zero-Shot response is competitive or superior precisely because it produces a shorter, more context-proximate answer, while the Cog-CoT pipeline generates a richer, multi-step synthesis that is semantically valid but lexically divergent from the retrieved passage. The average step cosine scores (0.124-0.321) are uniformly below θ = 0.65, meaning all verification steps FAIL and the pipeline falls back to retaining the highest-scoring attempt rather than converging on a verified answer. This confirms that the verification mechanism, effective at catching factual drift in C1-C5, systematically over-penalises legitimate creative synthesis at C6.

4.5. Self-Verification Analysis

Figure 4 and Table 12 detail the verification outcomes across all models.

Table 12. Step-level self-verification statistics

Level	Total Steps	PASS	FAIL	Avg. Pass Cosine
Gemma-3-4B-IT	1,409	96	1,313	0.740
LLaMA-3.1-8B-Instruct	185	29	156	0.774
Qwen2.5-7B-Instruct	210	51	159	0.742

The 6.8–24.3% PASS rates are low in absolute terms but should not be interpreted as evidence of poor reasoning quality. Steps that exceed the verification threshold achieve average cosine scores of 0.740–0.774, confirming that the filter performs selective, high-precision verification rather than being trivially satisfied. The low overall PASS rate largely reflects the characteristics of Indonesian morphology: prefixation and suffixation produce surface forms that diverge considerably from the dictionary roots present in the retrieved context, thereby reducing cosine similarity even when the underlying meaning remains intact. Furthermore, the fallback mechanism in Algorithm 3 guarantees that a FAIL verdict never discards an answer after MAX_{RETRY} attempts. Instead, the pipeline retains the highest-scoring attempt and proceeds with aggregation regardless of whether the threshold is met. Consequently, FAIL

functions primarily as a quality-ranking signal across retry attempts rather than as a strict accept–reject criterion for the final output.

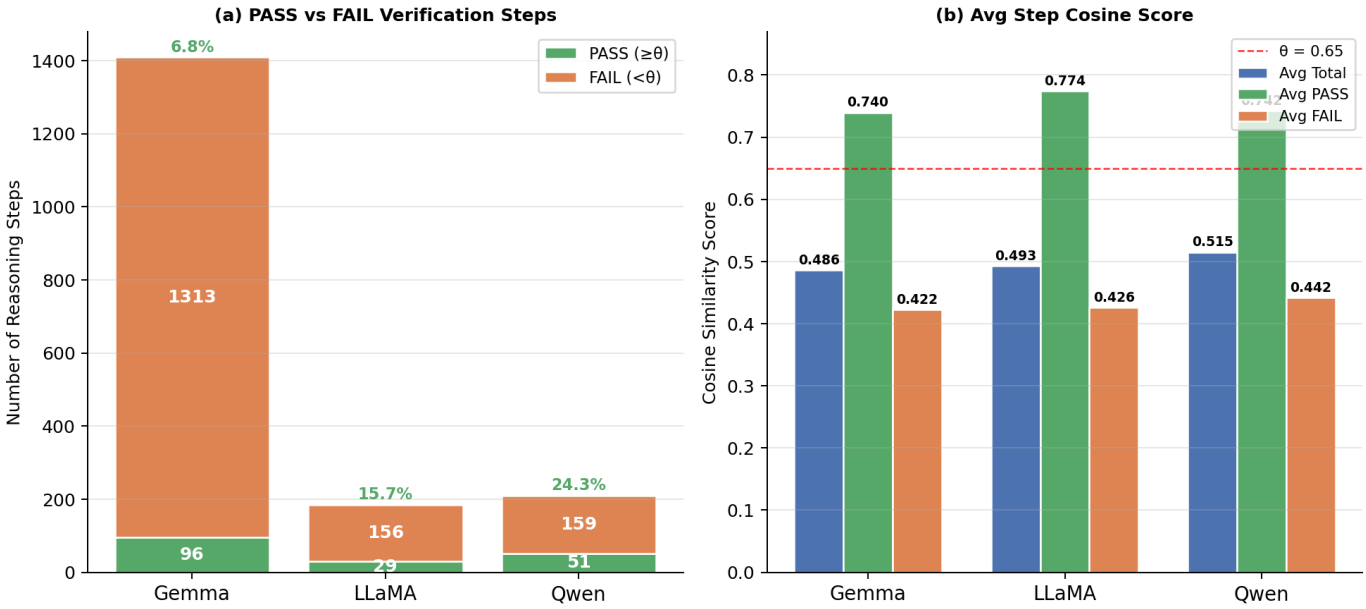

Figure 4. Self-verification per model: (**a**) PASS vs. FAIL step counts with percentage labels; (**b**) average cosine scores for PASS and FAIL steps.

A key design question is whether the step-wise verification mechanism functions as a genuine validation gate or merely as a quality-ranking signal that selects the best among imperfect retry attempts. The evidence supports the latter characterization in the majority of cases. Table 13 presents three representative FAIL cases illustrating the primary causes of verification failure. Case A (Near Miss) and Case B (Very Low Score) demonstrate that FAIL verdicts frequently arise from lexical mismatches between semantically sound reasoning steps and the TF-IDF-indexed context rather than from factual errors. In these cases, the mechanism functions primarily as a ranking signal across retry attempts. In contrast, Case C (Genuine Failure) shows a scenario in which the FAIL verdict correctly identifies a problematic, ungrounded sub-question introduced by the Decomposer, where the inability to produce a passing attempt leads to genuine output degradation. A similar pattern is observed at C6, as illustrated in Table 11, where all five reasoning steps FAIL for the itinerary example not because the generated schedule is factually incorrect, but because creative scheduling terminology diverges from the vocabulary of the retrieved context. The low overall PASS rates therefore reflect the combined effects of Indonesian morphological variation, which suppresses cosine similarity, and a smaller proportion of genuine factual errors.

Table 13. Three representative FAIL cases.

Case	Sample Question	Step Cosine (FAIL)	Root Cause	Aggregated Final GT Cosine Similarity
A – Near Miss	Which continent is the largest and most densely populated in the world?	0.583–0.590	Lexical paraphrase of context, not a factual error	0.697 (vs. Zero-Shot 0.755, CoT 0.697)
B – Very Low Score	State the founding year of the Budi Utomo organization.	0.186	Short fact diluted against long multi-topic context	0.818 (the highest)
C – Genuine Failure	Explain why La Niña affects farmers' planting-season decisions	0.499–0.589	Decomposer introduced an ungrounded sub-question	0.487 (below Zero-Shot 0.706 and CoT 0.699)

4.6. Generalizability on External Benchmark

Figure 5 places IPS-Generation and SQuAD+ASQA results side by side; Table 14 provides the full numerical breakdown.

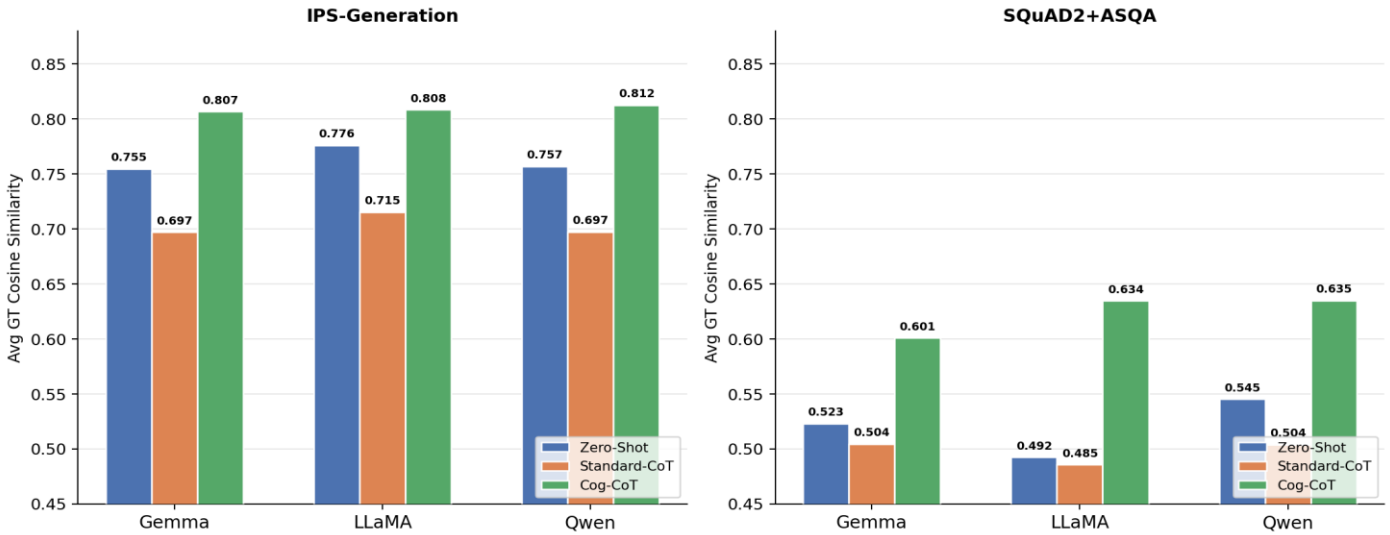

Figure 5. GT Cosine Similarity comparison: (**a**) IPS-Generation; and (**b**) SquAD 2.0 + ASQA.

Table 14. Evaluation results on the external benchmark.

Model	Strategy	GT Cosine Similarity	BLEU-4	ROUGE-L	METEOR	BERTScore
Gemma-3-4B-IT	Zero-Shot	0.5229	0.0095	0.1050	0.0835	0.6515
	Standard CoT	0.5044	0.0109	0.0968	0.1071	0.6136
	Cog-CoT	**0.6007**	**0.0295**	**0.1655**	**0.2143**	**0.6865**
LLaMA-3.1-8B-Instruct	Zero-Shot	0.4923	0.0099	0.1111	0.0784	0.6572
	Standard CoT	0.4855	0.0123	0.0909	0.1168	0.6139
	Cog-CoT	**0.6344**	**0.0518**	**0.2138**	**0.2705**	**0.7134**
Qwen2.5-7B-Instruct	Zero-Shot	0.5451	0.0152	0.1221	0.1244	0.6544
	Standard CoT	0.5038	0.0142	0.0940	0.1265	0.6179
	Cog-CoT	**0.6347**	**0.0510**	**0.2200**	**0.2714**	**0.7101**

Three observations merit discussion. First, the performance ranking of Cog-CoT > Zero-Shot > Standard CoT remains consistent across both datasets and all three models, establishing a reliable behavioral pattern within this study. Second, CoT performance degradation reappears in English data at magnitudes comparable to those in the Indonesian corpus, ruling out any language-specific explanation. Third, Cog-CoT's relative gains over Zero-Shot are substantially larger on this benchmark than on IPS-Generation, an unexpected divergence explained by baseline quality: when Zero-Shot performance drops because the classifier was trained on Indonesian text and issues suboptimal routing labels for English queries, the cognitive scaffold compensates by imposing step structure that the model would otherwise lack entirely. The absolute GT Cosine Similarity scores are lower on this benchmark precisely because classifier accuracy degrades on out-of-domain English input.

The cross-lingual performance gap is primarily attributable to the Cognitive Classifier being trained exclusively on Indonesian IPS data. Several multilingual approaches could close this gap in future iterations. Encoder models such as mBERT and XLM-R have demonstrated strong zero-shot cross-lingual transfer on text classification tasks by jointly training across 100+ languages and sharing a unified token representation space. For Bloom-level classification specifically, a fine-tuned XLM-R or multilingual DeBERTa checkpoint trained on a combined Indonesian-English question bank would likely resolve the routing degradation observed on out-of-domain English queries. Alternatively, a language-detection preprocessing step could route English queries to a separately trained English Bloom classifier.

4.7. Statistical Significance of Performance

To confirm that Cog-CoT's performance improvements are not attributable to random variation, Wilcoxon signed-rank tests were conducted on per-sample scores for all five-

evaluation metrics on the IPS-Generation dataset, comparing Cog-CoT against both Zero-Shot and Standard CoT baselines across all three LLMs. The Wilcoxon signed-rank test was selected as a non-parametric paired test that does not assume normality, appropriate given the bounded distribution of cosine-based scores. Results are presented in Table 15.

Cog-CoT's superiority is statistically significant ($p < 0.05$) in 25 of 30 comparisons (83.3%). The five non-significant comparisons occur exclusively on the METEOR metric, specifically for Gemma vs. Zero-Shot ($p = 0.823$), Gemma vs. Standard CoT ($p = 1.000$), LLaMA vs. Standard CoT ($p = 0.083$), Qwen vs. Zero-Shot ($p = 0.610$), and Qwen vs. Standard CoT ($p = 0.977$). In these five cases, Cog-CoT's average METEOR score is marginally lower than the respective baseline (e.g., Gemma Cog-CoT METEOR: 0.2380 vs. Standard CoT: 0.2497), explaining the non-significance. This is attributable to a known limitation of METEOR: it relies on WordNet synonym matching and English morphological rules and is therefore less sensitive to Indonesian lexical variation. The four remaining metrics (GT Cosine Similarity, BLEU-4, ROUGE-L, BERTScore) are all statistically significant across all three models and both baseline comparisons, confirming that the METEOR non-significance reflects metric insensitivity to Indonesian rather than a failure of the pipeline.

Table 15. Wilcoxon Signed-Rank Test results.

Model	Strategy	p vs (Zero-Shot)	Sig.	p (vs. Standard CoT)	Sig.	Note
Gemma-3-4B-IT	GT Cosine Similarity	4.44e-06	✓	5.58e-14	✓	
	BLEU-4	3.95e-12	✓	1.48e-05	✓	
	ROUGE-L	5.69e-14	✓	1.24e-13	✓	
	METEOR	**0.8232**	✗	**1.0000**	✗	*
	BERTScore	4.95e-15	✓	5.11e-15	✓	
LLaMA-3.1-8B-Instruct	GT Cosine Similarity	5.14e-04	✓	3.03e-14	✓	
	BLEU-4	3.46e-04	✓	5.32e-07	✓	
	ROUGE-L	2.82e-05	✓	3.08e-10	✓	
	METEOR	**0.0184**	✓	**0.0831**	✗	*
	BERTScore	1.59e-10	✓	2.88e-17	✓	
Qwen2.5-7B-Instruct	GT Cosine Similarity	7.50e-06	✓	2.57e-15	✓	
	BLEU-4	3.65e-09	✓	2.13e-06	✓	
	ROUGE-L	1.30e-11	✓	6.45e-11	✓	
	METEOR	**0.6103**	✗	**0.9769**	✗	*
	BERTScore	7.42e-15	✓	1.90e-17	✓	

4.8. Impact of Bloom Classification Errors on Pipeline Output

4.8.1. Classification Accuracy on the Evaluation Set

The Cognitive Classifier achieved 99.1% accuracy on the IPS-Generation evaluation set. Per-level accuracy was 100% for C2 through C6 and 95.0% for C1-Mengingat (19/20), as shown in Table 16. The single misclassification involved a C1 (Remember) question predicted as C3 (Apply): Which continent is the largest and most densely populated in the world?. The classifier assigned Apply because the comparative superlative cues resemble application-level surface patterns, despite the answer requiring only direct factual recall.

4.8.2. Downstream Impact on Generation Quality

The single misclassified sample provides a controlled natural experiment for assessing how a routing error propagates to generation quality. Table 17 presents per-strategy GT Cosine Similarity, ROUGE-L, and BERTScore for this sample alongside the dataset-wide correctly classified average.

Table 16. Bloom classifier accuracy per level.

Level	Correct	Total	Accuracy (%)	Misclassified
C1	19	20	95.0	C3 (Apply)
C2	20	20	100.0	-
C3	20	20	100.0	-
C4	20	20	100.0	-
C5	20	20	100.0	-
C6	20	20	100.0	-
Total	120	120	99.1	-

Table 17. Generation quality on the misclassified sample vs. correctly classified average

Strategy	GT Cosine (Misclassified)	GT Cosine (Correct Avg.)	ROUGE-L (Misclassified)	BERTScore (Misclassified)
Zero-Shot	0.7554	0.7546	0.2069	0.6562
Standard CoT	0.6399	0.6969	0.1556	0.6408
Cog-CoT	**0.8264**	**0.8123**	**0.2953**	**0.7588**

Despite receiving a C3 (Apply) routing instead of the correct C1 (Remember), Cog-CoT still achieves the highest GT Cosine Similarity (0.8264) among all three strategies, outperforming Zero-Shot (0.7554) and Standard CoT (0.6399). This occurs because even when incorrectly routing to C3, the pipeline constructs a bottom-up chain beginning at C1 (Remember) and C2 (Understand) before reaching C3 (Apply). The earlier sub-tasks ground the chain in factual recall before the chain reaches the incorrectly assigned target level. The structural completeness of the cognitive chain therefore provides inherent robustness against adjacent-level misclassification that neither Zero-Shot nor Standard CoT can replicate. Notably, the misclassified sample's Cog-CoT score (0.8264) is marginally above the correctly classified average (0.8123), suggesting that the richer C3 chain produced a more complete and contextually grounded answer than a minimal C1 chain would have. Deeper cognitive scaffolding, even when applied to a lower-order question, can enhance answer comprehensiveness without introducing factual drift.

The bottom-up chain architecture provides inherent robustness against upward misclassifications (a question routed to a higher Bloom level than its true level), because lower levels are always instantiated as sub-tasks before the incorrectly assigned target. Conversely, a downward misclassification (e.g., C4 routed as C2) would truncate the chain and may produce an insufficiently deep response. The 99.1% classifier accuracy is also partially a function of the evaluation set being drawn from the same domain and grade range as the training corpus; out-of-domain generalisation may yield more misclassifications, making this robustness property more consequential and further motivating the cross-domain extension.

4.9. Threshold Sensitivity Analysis

The verification threshold θ controls the minimum cosine similarity a reasoning step must achieve against the retrieved context to receive a PASS verdict. A lower θ is more permissive while a higher θ is more stringent, triggering more retries but potentially enforcing stricter factual grounding. To evaluate sensitivity to this hyperparameter, the full Cog-CoT pipeline was run across four threshold values ($\theta \in \{0.55, 0.60, 0.65, 0.70\}$) on a stratified subset of IPS-Generation. Results are presented in Tables 18 and 19.

Table 18. Overall metrics and verification across threshold values.

θ	GT Cosine Similarity	BLEU-4	ROUGE-L	METEOR	BERT Score	Pass Rate	Avg. Retries
0.55	0.8154	0.0662	0.2374	0.2445	0.7356	**0.354**	5.60
0.60	0.8052	0.0634	0.2324	0.2358	0.7323	0.237	5.87
0.65	**0.8224**	**0.0714**	**0.2375**	**0.2470**	**0.7389**	0.179	**6.40**
0.70	0.8090	0.0674	0.2327	0.2441	0.7383	0.123	6.73

Table 18 reveals that theta=0.65 achieves the highest overall GT Cosine Similarity (0.8224), BLEU-4 (0.0714), METEOR (0.2470), and BERTScore (0.7389) among the four candidates, confirming it as the optimal threshold on this subset. The relationship between theta and generation quality is non-monotonic: GT Cosine rises from theta=0.55 (0.8154) to a peak at theta=0.65 (0.8224) before declining at theta=0.70 (0.8090). This pattern reflects two competing forces. Lowering theta allows more steps to pass without retry, reducing the quality-ranking benefit of the verification signal. Raising theta beyond 0.65 makes the threshold stricter than the typical embedding distance between correct steps and Indonesian retrieved context, causing retry exhaustion on steps that are factually grounded but lexically paraphrased. The pass rate decreases monotonically from 0.354 at theta=0.55 to 0.123 at theta=0.70, and average retries increase from 5.60 to 6.73. This confirms the expected inverse relationship between threshold strictness and verification success rate. The substantially higher retry count at theta=0.70 relative to theta=0.65 (6.73 vs. 6.40) indicates diminishing returns in verification stringency: the additional retries are not producing better steps meaningfully but are increasing computational overhead.

Table 19. Per-level GT cosine similarity across threshold values.

Level	Theta=0.55	Theta=0.60	Theta=0.65	Theta=0.70	Sensitivity
C1	0.8767	0.8765	**0.8908**	0.8730	Low (range: 0.018)
C2	0.8062	0.8055	**0.8289**	0.8098	Low (range: 0.023)
C3	**0.8749**	0.8078	0.8718	0.8259	High (range: 0.067)
C4	0.8500	**0.8563**	0.8420	0.8489	Low (range: 0.014)
C5	0.8376	0.8370	**0.8399**	0.8380	Low (range: 0.003)
C6	0.6469	0.6483	**0.6612**	0.6582	Moderate (range: 0.014)

Table 19 shows that per-level sensitivity varies substantially. C1, C2, C4, and C5 exhibit low sensitivity (range <0.03), suggesting that recall, comprehension, analysis, and evaluation steps are sufficiently grounded in the retrieved context to perform consistently regardless of the threshold. C3 exhibits high sensitivity (range 0.067, from 0.8078 at theta=0.60 to 0.8749 at theta=0.55), while C6 shows moderate sensitivity (range 0.014). The C3 pattern is noteworthy: the highest C3 performance occurs at theta=0.55, the most permissive threshold, suggesting that C3 application steps are frequently penalised by stricter thresholds even when they correctly apply retrieved knowledge to a new context. Taken together, these results provide two design guidelines. First, theta=0.65 is a well-calibrated default for the IPS-Generation domain, achieving the best overall quality while maintaining a manageable retry load. Second, the C3 and C6 sensitivity patterns further support the case for a level-adaptive threshold; a lower threshold for application-level (C3) and creative (C6) tasks would improve performance at those levels without sacrificing accuracy at the lower-order levels where fixed-threshold sensitivity is already low.

4.10. CoT Performance Degradation

The most unexpected finding concerns Standard CoT. Across all three models, unscaffolded CoT instruction degrades GT Cosine Similarity below Zero-Shot: −7.65% on Gemma, −7.86% on LLaMA, and −7.84% on Qwen. Section 4.6 shows that identical degradation occurs on the English benchmark, ruling out a domain-specific or language-specific artifact. The radar visualization in Fig. 6 makes the degradation conspicuous: on every model, the Standard CoT polygon shrinks within the Zero-Shot polygon. RAG-retrieved text is finite and specific, whereas parametric knowledge is diffuse; the longer an unguided reasoning chain extends, the further it drifts from the ground-truth context. This drift then propagates to the final answer, reducing its cosine similarity to the teacher-authored reference.

From a deployment standpoint, this finding carries a direct warning: integrating a generic CoT instruction into a RAG-based educational QA system is not a neutral or conservative choice. This study demonstrates that this change tends to degrade overall answer quality. The failure case analysis in Section 4.5 provides additional supporting evidence. Case C, where the Decomposer introduced an ungrounded sub-question, mirrors the mechanism hypothesised for Standard CoT degradation: once the reasoning chain departs from retrieved context at any step, subsequent steps build on ungrounded content, compounding the drift. The critical

difference is that in Cog-CoT, per-step verification catches and limits this propagation; in Standard CoT, there is no such mechanism, allowing drift to accumulate unchecked before producing the final answer. This explains why Standard CoT degradation is consistent across all three LLMs. Furthermore, the fact that Standard CoT produces longer responses than Zero-Shot, evidenced by marginally higher METEOR scores in some conditions despite lower GT Cosine Similarity, is consistent with the verbosity-amplified context drift mechanism: the model generates more text, but that additional text departs further from the teacher-authored reference. We characterise this as a verbosity-amplified context drift phenomenon and note that it represents a previously undocumented failure mode of unscaffolded CoT in retrieval-grounded settings.

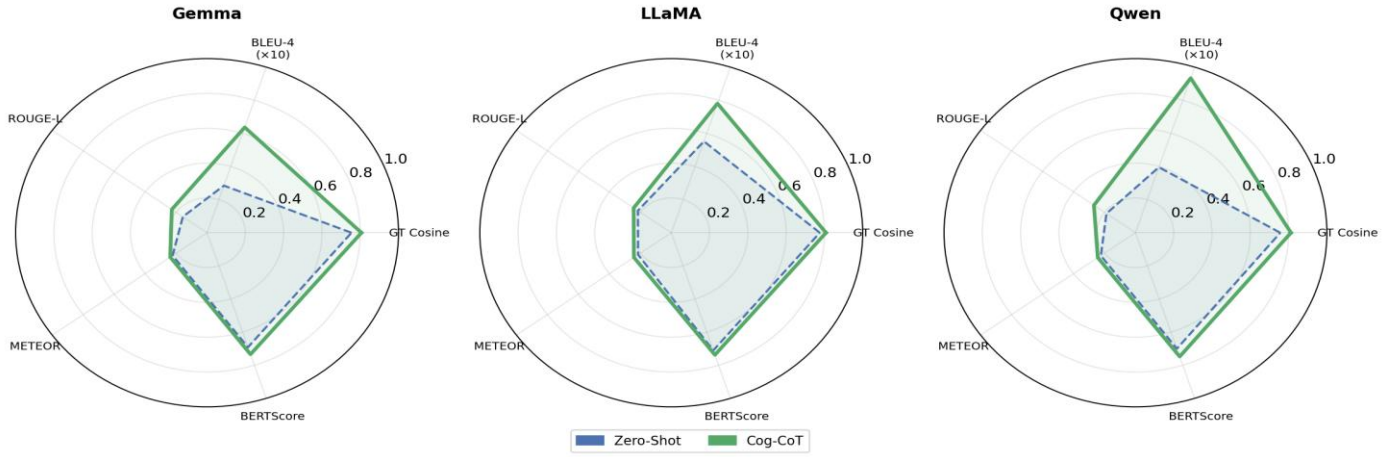

Figure 6. Comparison of Zero-Shot vs. Cog-CoT across all five metrics per model.

4.11. Ablation Study

4.11.1 Component Removal Analysis

Table 20 reports the contribution of each pipeline component when removed in isolation**.** NoRAG produces the largest single-component loss across all five metrics. This outcome is not merely expected but mechanistically necessary: without retrieved context, the verification module has no reference against which to compute cosine scores, so every step scores below θ, every retry attempt exhausts, and the pipeline's quality guarantee collapses. The functional dependence between RAG and verification is thus quantified rather than assumed.

NoDecomp is the second-largest contributor to GT Cosine Similarity (−1.77%), confirming that hierarchical sub-question scaffolding adds value beyond what a direct top-level answer can achieve, even when RAG and verification are retained. NoVerify and NoRetry sit close together (GT Cosine Similarity 0.8037 and 0.7970, respectively), suggesting that the retry mechanism contributes more than the filtering action of verification per se: it is the act of regenerating under a failed signal, rather than the discard itself, that raises final quality. NoAggLLM drops GT Cosine Similarity to 0.7656, demonstrating that naïve concatenation of sub-answers, without LLM-mediated synthesis, fails to produce the coherent paragraph structure that embedding-based metrics reward**.**

Table 20. Ablation study results averaged across three LLMs.

Variant	GT Cosine Similarity	BLEU-4	ROUGE-L	METEOR	BERTScore
Cog-CoT (Full)	**0.8123**	**0.0855**	**0.2448**	**0.2375**	**0.7357**
NoDecomp	0.7979	0.0476	0.2211	0.2075	0.7308
NoRAG	0.7462	0.0438	0.1953	0.2007	0.7158
NoVerify	0.8037	0.0620	0.2310	0.2328	0.7340
NoRetry	0.7970	0.0632	0.2302	0.2339	0.7335
NoAggLLM	0.7656	0.0600	0.2252	0.2254	0.7324

Section 4.12.2 extends this analysis with a granular component isolation study, building from a bare baseline up to the full system to quantify the independent contribution of each module.

4.11.2 Granular Component Isolation

To complement the component removal analysis, Table 21 presents a granular component isolation study evaluating six configurations on a stratified IPS-Generation subset, progressively adding pipeline modules from a bare baseline to the complete system. Per-level GT Cosine Similarity patterns are discussed in the analysis below.

Table 21. Component isolation results.

Configuration	Modules Active	GT Cosine	BLEU-4	ROUGE-L	METEOR	BERTScore
Bare	No Bloom routing, no RAG, no verification	0.8007	0.0439	0.2171	0.1896	0.7267
Classifier Only	Bloom routing only; no RAG, no verification	0.7491	0.0423	0.2064	0.1946	0.7185
RAG Only	RAG grounding only; no Bloom, no verification	0.8351	0.0582	0.2434	0.2130	0.7357
Classifier + RAG	Bloom routing + RAG; no verification	0.8078	0.0538	0.2330	0.2047	0.7279
Verification Only	RAG + per-step verification; no Bloom routing	**0.8408**	0.0599	0.2446	**0.2211**	0.7318
Full Cog-CoT	Complete system: Bloom + RAG + verification + retry	0.8002	**0.0681**	**0.2457**	0.2188	**0.7361**

Table 21 reveals that the Bare configuration, generating answers without any Bloom routing, RAG grounding, or per-step verification, achieves a GT Cosine Similarity of 0.8007, establishing a strong unstructured baseline. The Classifier Only configuration, which adds Bloom-level routing without RAG context or verification, drops to 0.7491 (−0.0516 vs. Bare), the lowest among all six configurations. This finding confirms that Bloom-level routing in isolation, without retrieved context to anchor the sub-task answers, actively constrains generation: the structural decomposition prescribes cognitive levels but without factual grounding, producing hallucinated or topic-drifting sub-task outputs that degrade the final aggregation. The RAG Only configuration achieves 0.8351, establishing that context grounding is the single most impactful module in the pipeline, consistent with the component removal ablation in Table 20. Adding the Bloom Classifier to RAG (Classifier + RAG, 0.8078) yields a slight decrease relative to RAG Only (−0.0273), again confirming that the cognitive chain imposes overhead that requires per-step verification to be beneficial. The Verification Only configuration (RAG + per-step verification, without Bloom routing) achieves the highest overall GT Cosine (0.8408), demonstrating that per-step grounding verification is highly effective even without cognitive scaffolding and further establishing retrieval-augmented verification as the dominant performance driver.

Together, the component isolation results complement the component removal findings in Table 20 and yield three converging design insights. First, RAG grounding is the foundational module; its presence or absence dominates all other configuration choices. Second, the Bloom Classifier requires both RAG and verification to deliver its benefit: in isolation it is the weakest configuration, but in the full system it provides the cognitive structure that drives the C3 and C4 gains observed in the main results. Third, Verification Only is a strong alternative for applications where cognitive scaffolding is not required. The Full system's advantage is concentrated at the mid-to-high Bloom levels where pedagogical structure matters most.

4.12. Discussion

The classifier addresses the first gap identified in the introduction: confining all nine misclassifications to adjacent Bloom levels means the predicted label is reliable enough to serve as a routing decision, rather than something to be validated and discarded after the fact, as prior Bloom classifiers have typically been used. Its margin over the next-best model is

modest in absolute terms. Yet this margin matters disproportionately for pipeline integrity because a single misclassification can misdirect the template and decomposition depth for every downstream module. The second gap, the absence of cognitive ordering within conventional CoT prompting, is addressed by the consistency of the strategy ranking itself: Cog-CoT outperforms Zero-Shot, and Zero-Shot outperforms Standard CoT, across all nine model-dataset combinations without exception. That this ranking holds across three architecturally distinct models developed by three independent organizations (Google, Meta, and Alibaba) suggests the improvement stems from the scaffold rather than from an incidental alignment between a model and the prompt wording. The ablation study closes the third gap by identifying where this property originates: removing RAG or verification yields the two largest drops in GT Cosine Similarity, supporting the claim that per-step grounding, not decomposition alone, prevents the answer from drifting.

The degradation observed under Standard CoT is arguably the more consequential finding here, since it is unexpected given that CoT prompting is routinely treated as a safe default. Instructing the model to reason step by step, without any cognitive scaffold, not only fails to help in this RAG setting; it also produces a measurable 7–8% decline on the Indonesian data and a comparable margin on the English benchmark. Because this pattern held across three model families built through three distinct instruction-tuning procedures, it is difficult to attribute the effect to any single model's training process. The explanation proposed earlier, that retrieval-context abandonment increases under verbosity pressure as an open-ended reasoning chain draws increasingly on parametric memory rather than the retrieved passage, is consistent with this pattern but was not measured directly at the token level; confirming the mechanism more precisely is left to future work. What the result does establish, independent of mechanism, is where Cog-CoT's value actually sits: its advantage over Standard CoT (+12% to +15% GT Cosine Similarity) is roughly twice its advantage over Zero-Shot (+4% to +7%), indicating that the scaffold's primary function is to prevent unscaffolded reasoning from damaging output quality, more than it is to outperform the model's default behavior.

The level-by-level breakdown qualifies the headline claim that Cog-CoT improves on Zero-Shot, since the improvement is not uniform across Bloom levels. C4 and C5 perform best under Cog-CoT, which is unsurprising given that the Deconstruct→Examine→Relate→Synthesize sequence used at C4 and the Set Criteria→Gather Evidence→Judge sequence used at C5 both resemble the structure IPS teachers already use when writing comparative or evaluative answers. C6 behaves differently; the fixed cosine threshold penalizes answers that express content the retrieved context already supports using different wording, working against the model rather than for it. Because BERTScore is more tolerant of lexical variation and reverses this verdict when applied to the same C6 outputs, the underlying issue appears to lie more with evaluation sensitivity than with generation failure; the C6 problem is, at least in part, an artifact of measuring correctness against a single teacher's phrasing rather than a genuine shortcoming in what the model generates.

The verification statistics require careful interpretation, since an unqualified reading of Table 12 might suggest the pipeline fails most of the time it runs. PASS rates of 6.8–24.3% appear low until the case studies in Table 13 are considered. Case A scores 0.583 not because the underlying reasoning is incorrect but because it paraphrases the retrieved context rather than echoing it; Case B scores 0.186 because a one-sentence factual answer is being compared against a long, multi-topic passage, a setting in which cosine similarity is known to underestimate semantic relatedness. Only Case C, where the decomposer introduces a sub-question with no grounding in the source material, represents a genuine failure, and even there the error originates in the decomposition module rather than in verification itself. The retry-and-retain logic in Algorithm 2 limits how much this distinction affects the final answer, since a FAIL verdict never blocks output; it only determines which of up to three candidate responses is retained. A high FAIL count is therefore consistent with a pipeline that is functioning correctly under a verification threshold stricter than ground-truth correctness actually requires, rather than evidence that the threshold itself is failing. Whether that threshold is appropriately calibrated for Indonesian remains an open question: the low absolute PASS rates trace back to Indonesian affixation compressing cosine scores even when meaning is preserved, which points toward morphologically aware encoders such as IndoBERT or IndoSBERT as a more suitable verification backbone in the future.

The external benchmark results distinguish two considerations that are often treated as one: whether the architecture itself works, and whether the specific classifier trained for this

paper generalizes outside its training distribution. Both the strategy ranking and the degradation effect are reproduced on SQUAD 2.0 and ASQA, indicating that the architecture does not depend on Indonesian morphology or the IPS domain specifically. What does not reproduce is absolute performance; GT Cosine Similarity falls from roughly 0.81 on IPS-Generation to roughly 0.60–0.63 on the English data. Since the classifier was trained exclusively on Indonesian IPS questions, it has a limited basis for routing English questions correctly, and an incorrect label alters the decomposition depth and prompt template applied throughout the rest of the pipeline. This pattern is better understood as a limitation of the specific classifier used in these experiments than as evidence against the underlying design, which is also why a multilingual Bloom-level classifier, rather than any modification to the reasoning or verification modules, is identified as the most impactful direction for future work.

5. Comparison

Table 22 positions Cog-CoT relative to five representative prior approaches along the four dimensions most relevant to the research problem: cognitive scaffolding, per-step factual verification, cross-architecture evaluation, and cross-domain portability.

Table 22. Positions of Cog-CoT relative to five prior approaches.

Approach	Cognitive Scaffold	Per Step Verification	Cross-Arch.	Cross-Domain
Zero-Shot (LLM)	None	None	Yes	Yes
Standard CoT	None (ad-hoc)	None	Yes	Yes
TreeQA [16]	Linguistic dep.	None	Limited	Limited
RAG + CoT [11]	None	Output only	No	No (medical)
RAG + ReAct [20]	None	None	No	Indonesian only
Cog-CoT (proposed)	Bloom C1–C6	Per-step θ=0.65	Yes (3 LLMs)	Yes (4 datasets)

Cog-CoT is the only entry in Table 23 that satisfies all four criteria simultaneously. Standard CoT satisfies the cross-architecture and cross-domain criteria but lacks both cognitive scaffolding and per-step verification; the CoT performance degradation findings in Section 4.11 show that this absence is not a minor oversight but a property that actively inverts the technique's advertised benefit in RAG-based educational QA. TreeQA [16] introduces structured decomposition but grounds it in linguistic dependency rather than cognitive hierarchy, which prevents it from enforcing the requirement that every Bloom level receive explicit treatment before the target level is addressed. Chen et al. and Tampubolon et al. demonstrate that domain-specific grounding through RAG (with output-level verification) or ReAct can reduce hallucination. Still, neither system attempts to regulate cognitive depth. Cog-CoT's architecture bridges all these gaps, and its GT Cosine Similarity gains of +4.16% to +7.36% over Zero-Shot constitute the first architecture-agnostic evidence that Bloom's Taxonomy can function as a computational reasoning scaffold yielding consistent pedagogical improvements.

6. Conclusions

This paper introduced Cog-CoT, a four-module pipeline that converts Bloom's Taxonomy from an evaluation rubric into a first-class computational scaffold. Three findings merit emphasis. The LinearSVC Cognitive Classifier effectively routes queries (weighted F1=0.9915), ensuring that any misclassifications are strictly confined to adjacent Bloom levels. While classification serves as a secondary support mechanism in this framework, the absence of LOTS-HOTS confusion provides the highly reliable foundational signal required to initiate the subsequent bottom-up cognitive decomposition without propagating routing errors.

The full Cog-CoT pipeline consistently outperforms Zero-Shot by +4.16% to +7.36% in GT Cosine Similarity and outpaces Standard CoT by +12% to +15% across three model families and two datasets. That the same ordinal ranking, Cog-CoT > Zero-Shot > Standard CoT, holds without exception across nine model-dataset combinations indicates a structural rather than incidental advantage: the scaffold constrains the generative process in a way that aligns with how ground-truth answers are constructed. Standard CoT consistently degrades

relative to Zero-Shot in this RAG-based educational setting, for which this paper offers both empirical documentation and a mechanistic account (retrieval-context abandonment under verbosity pressure). This finding has immediate practical implications: system designers should treat generic CoT as a risk factor rather than a safe default when deploying LLMs in document-grounded educational QA.

Several limitations bound the current findings. First, the Cognitive Classifier is trained exclusively on Indonesian IPS questions; its cross-lingual applicability is limited, as evidenced by the lower absolute GT Cosine Similarity scores on the English benchmark. Future work should evaluate multilingual encoder-based classifiers (e.g., XLM-R, mDeBERTa) trained on multilingual Bloom-annotated corpora to remove this bottleneck. Second, all primary experiments are conducted within the IPS domain. While the consistent pattern across IPS and the English benchmark suggests some domain robustness, the framework has not been evaluated on STEM domains where correct answers differ structurally from narrative social studies content. Third, the verification threshold theta=0.65 was calibrated for Indonesian morphological variation, and the C6 penalization analysis (Table 11) suggests a level-adaptive threshold may better balance verification precision and creative generation at higher Bloom levels. Fourth, evaluation relies exclusively on automatic metrics; expert educator assessment of pedagogical quality and Bloom-level appropriateness remains an important future validation step. Addressing these limitations constitutes a concrete research agenda for extending Cog-CoT beyond its current scope.

Author Contributions: Conceptualization: A.M., R.N.E.A., and D.P.; Methodology: A.M.; Software: A.M.; Validation: A.M., R.N.E.A., and D.P.; Formal analysis: A.M.; Investigation: A.M.; Data curation: A.M.; Writing - original draft: A.M.; Writing - review and editing: R.N.E.A. and D.P.; Supervision: R.N.E.A. and D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received partial funding from the IRN Type C Grant from Institut Teknologi Sepuluh Nopember with contract No 2662/PKS/ITS/2026.

Data Availability Statement: The datasets generated and analyzed during this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors gratefully acknowledges the Department of Informatics, Institut Teknologi Sepuluh Nopember (ITS), Surabaya, for providing the computational resources, partial funding and administrative support that facilitated this research. During the preparation of this work, the authors used Claude (Anthropic) to assist with language editing and formatting checks. All conceptual, methodological, and analytical contributions originated entirely from the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] R. Prakash and R. Litoriya, "Pedagogical Transformation of Bloom Taxonomy's LOTs into HOTs: An Investigation in Context with IT Education," *Wirel. Pers. Commun.*, vol. 122, no. 1, pp. 725–736, Jan. 2022, doi: 10.1007/s11277-021-08921-2.

[2] L. Yan *et al.*, "Practical and ethical challenges of large language models in education: A systematic scoping review," *Br. J. Educ. Technol.*, vol. 55, no. 1, pp. 90–112, Jan. 2024, doi: 10.1111/bjet.13370.

[3] A. E. Evwiekpaefe, D. T. Chinyio, and L. K. Tohomdet, "The Llama–ARCS Adaptive Learning framework: AI–VR Integration System for Real-Time Motivational Feedback in Higher Education," *J. Comput. Theor. Appl.*, vol. 3, no. 3, pp. 260–273, Feb. 2026, doi: 10.62411/jcta.15031.

[4] A. Herrmann-Werner *et al.*, "Assessing ChatGPT's Mastery of Bloom's Taxonomy Using Psychosomatic Medicine Exam Questions: Mixed-Methods Study," *J. Med. Internet Res.*, vol. 26, p. e52113, Jan. 2024, doi: 10.2196/52113.

[5] E. Kasneci *et al.*, "ChatGPT for good? On opportunities and challenges of large language models for education," *Learn. Individ. Differ.*, vol. 103, p. 102274, Apr. 2023, doi: 10.1016/j.lindif.2023.102274.

[6] A. Sangodiah, T. Jee San, Y. Tien Fui, L. Ean Heng, R. K. Ayyasamy, and N. A Jalil, "Identifying Optimal Baseline Variant of Unsupervised Term Weighting in Question Classification Based on Bloom Taxonomy," *MENDEL*, vol. 28, no. 1, pp. 8–22, Jun. 2022, doi: 10.13164/mendel.2022.1.008.

[7] M. Chindukuri and S. Sivanesan, "Transfer learning for Bloom's taxonomy-based question classification," *Neural Comput. Appl.*, vol. 36, no. 31, pp. 19915–19937, Nov. 2024, doi: 10.1007/s00521-024-10241-y.

[8] J. Miao, C. Thongprayoon, S. Suppadungsuk, P. Krisanapan, Y. Radhakrishnan, and W. Cheungpasitporn, "Chain of Thought Utilization in Large Language Models and Application in Nephrology," *Medicina (B. Aires).*, vol. 60, no. 1, p. 148, Jan. 2024, doi: 10.3390/medicina60010148.
[9] M. Lu, F. Gao, X. Tang, and L. Chen, "Analysis and prediction in SCR experiments using GPT-4 with an effective chain-of-thought prompting strategy," *iScience*, vol. 27, no. 4, p. 109451, Apr. 2024, doi: 10.1016/j.isci.2024.109451.
[10] Y. Liu, "Retrieval-Augmented Generation: Methods, Applications and Challenges," *Appl. Comput. Eng.*, vol. 142, no. 1, pp. 99–108, Apr. 2025, doi: 10.54254/2755-2721/2025.KL22312.
[11] Z. Chen *et al.*, "MedScaleRE-PF: a prompt-based framework with retrieval-augmented generation, chain-of-thought, and self-verification for scale-specific relation extraction in Chinese medical literature," *Inf. Process. Manag.*, vol. 62, no. 6, p. 104278, Nov. 2025, doi: 10.1016/j.ipm.2025.104278.
[12] O. Almatrafi and A. Johri, "Leveraging generative AI for course learning outcome categorization using Bloom's taxonomy," *Comput. Educ. Artif. Intell.*, vol. 8, p. 100404, Jun. 2025, doi: 10.1016/j.caeai.2025.100404.
[13] Z. Kolagar, F. Zalkow, and A. Zarcone, "Investigating Methods for Mapping Learning Objectives to Bloom's Revised Taxonomy in Course Descriptions for Higher Education," in *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, 2025, pp. 415–445. doi: 10.18653/v1/2025.bea-1.32.
[14] N. Duong-Trung, X. Wang, and M. Kravčík, "BloomLLM: Large Language Models Based Question Generation Combining Supervised Fine-Tuning and Bloom's Taxonomy," in *Lecture Notes in Computer Science*, 2024, pp. 93–98. doi: 10.1007/978-3-031-72312-4_11.
[15] G.-G. Lee, E. Latif, X. Wu, N. Liu, and X. Zhai, "Applying large language models and chain-of-thought for automatic scoring," *Comput. Educ. Artif. Intell.*, vol. 6, p. 100213, Jun. 2024, doi: 10.1016/j.caeai.2024.100213.
[16] X. Zhang *et al.*, "TreeQA: Enhanced LLM-RAG with logic tree reasoning for reliable and interpretable multi-hop question answering," *Knowledge-Based Syst.*, vol. 330, p. 114526, Nov. 2025, doi: 10.1016/j.knosys.2025.114526.
[17] A. Plaat, A. Wong, S. Verberne, J. Broekens, N. Van Stein, and T. Bäck, "Multi-Step Reasoning with Large Language Models, a Survey," *ACM Comput. Surv.*, vol. 58, no. 6, pp. 1–35, Apr. 2026, doi: 10.1145/3774896.
[18] Z. Chu *et al.*, "Navigate through Enigmatic Labyrinth A Survey of Chain of Thought Reasoning: Advances, Frontiers and Future," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024, pp. 1173–1203. doi: 10.18653/v1/2024.acl-long.65.
[19] Z. Li, Z. Wang, W. Wang, K. Hung, H. Xie, and F. L. Wang, "Retrieval-augmented generation for educational application: A systematic survey," *Comput. Educ. Artif. Intell.*, vol. 8, p. 100417, Jun. 2025, doi: 10.1016/j.caeai.2025.100417.
[20] A. L. Manuel Tampubolon, R. N. E. Anggraini, and S. C. Hidayati, "Retrieval Augmented Generation with Synergizing Reasoning and Acting Prompt Engineering for Indonesian Open-Domain Question Answering," in *2025 International Conference on Data Science and Its Applications (ICoDSA)*, Jul. 2025, pp. 333–338. doi: 10.1109/ICoDSA67155.2025.11157319.
[21] S. Jeong, J. Baek, S. Cho, S. J. Hwang, and J. Park, "Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Question Complexity," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2024, pp. 7036–7050. doi: 10.18653/v1/2024.naacl-long.389.
[22] S. Shaikh, S. M. Daudpotta, and A. S. Imran, "Bloom's Learning Outcomes' Automatic Classification Using LSTM and Pretrained Word Embeddings," *IEEE Access*, vol. 9, pp. 117887–117909, 2021, doi: 10.1109/ACCESS.2021.3106443.
[23] M. O. Gani *et al.*, "Towards enhanced assessment question classification: a study using machine learning, deep learning, and generative AI," *Conn. Sci.*, vol. 37, no. 1, Dec. 2025, doi: 10.1080/09540091.2024.2445249.
[24] P. Rajpurkar, R. Jia, and P. Liang, "Know What You Don't Know: Unanswerable Questions for SQuAD," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 784–789. doi: 10.18653/v1/P18-2124.
[25] I. Stelmakh, Y. Luan, B. Dhingra, and M.-W. Chang, "ASQA: Factoid Questions Meet Long-Form Answers," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 8273–8288. doi: 10.18653/v1/2022.emnlp-main.566.
[26] T. M. Seinen, J. A. Kors, E. M. van Mulligen, and P. R. Rijnbeek, "Using Structured Codes and Free-Text Notes to Measure Information Complementarity in Electronic Health Records: Feasibility and Validation Study," *J. Med. Internet Res.*, vol. 27, p. e66910, Feb. 2025, doi: 10.2196/66910.
[27] C. Prakash, M. Lind, and E. De La Cruz, "Hybrid Real-time Framework for Detecting Adaptive Prompt Injection Attacks in Large Language Models," *J. Comput. Theor. Appl.*, vol. 3, no. 3, pp. 286–301, Jan. 2026, doi: 10.62411/jcta.15254.