Improving Human Verification of LLM Reasoning through Interactive Explanation Interfaces

Runtao Zhou
University of Virginia
uar6nw@virginia.edu

Giang Nguyen
Auburn University
nguyengiangbkhn@gmail.com

Nikita Kharya
Independent Researcher
nikitakharya09@gmail.com

Anh Totti Nguyen
Auburn University
anh.ng8@gmail.com

Chirag Agarwal
University of Virginia
chiragagarwal@virginia.edu

Abstract

The reasoning capabilities of Large Language Models (LLMs) have led to their increasing employment in several critical applications, particularly education, where they support problem-solving, tutoring, and personalized study. Chain-of-thought (CoT) reasoning capabilities [1, 2] are well-known to help LLMs decompose a problem into steps and explore the solution spaces more effectively, leading to impressive performance on mathematical and reasoning benchmarks. As the length of CoT tokens per question increases substantially to even thousands of tokens per question [1], it is unknown how users could comprehend LLM reasoning and detect errors or hallucinations. To address this problem and understand how reasoning can improve human-AI interaction, we present three new interactive reasoning interfaces: interactive CoT (iCoT), interactive Program-of-Thought (iPoT), and interactive Graph (iGraph). That is, we ask LLMs themselves to **generate an interactive web interface** wrapped around the original CoT content, which may be presented in text (iCoT), graphs (iGraph) or code (iPoT). This interface allows users to interact with and provide a novel experience in reading and validating the reasoning chains of LLMs. Across a study of 125 participants, interactive interfaces significantly improve user performance. Specifically, iGraph users score the highest error detection rate (85.6%), followed by iPoT (82.5%), iCoT (80.6%), all outperforming standard CoT (73.5%). Interactive interfaces also lead to faster user validation time—iGraph users are faster (57.9 secs per question) than the users of iCoT and iPoT (60 secs) and the standard CoT (64.7 secs). A post-study questionnaire shows that users prefer iGraph, citing its superior ability to enable them to follow the LLM's reasoning. We discuss the implications of these results and provide recommendations for the future design of reasoning models. Try our interactive interfaces on HuggingFace.

1 Introduction

Large language models (LLMs) are now woven into high-stakes, real-world workflows, including education, where they assist with problem solving, tutoring, and personalized study [3–5]. A key driver of their recent success on complex tasks is chain-of-thought (CoT) prompting [6]: eliciting step-by-step intermediate reasoning often yields dramatic accuracy gains on arithmetic, commonsense, and symbolic reasoning benchmarks, with state-of-the-art performance on GSM8K and related math datasets. However, the current format for presenting LLM explanations may be far from optimal [7].

While CoT outputs contain rich information, their long-text presentation introduces significant challenges for human interpretation [9–11]. In practice, especially in modern reasoning LLMs

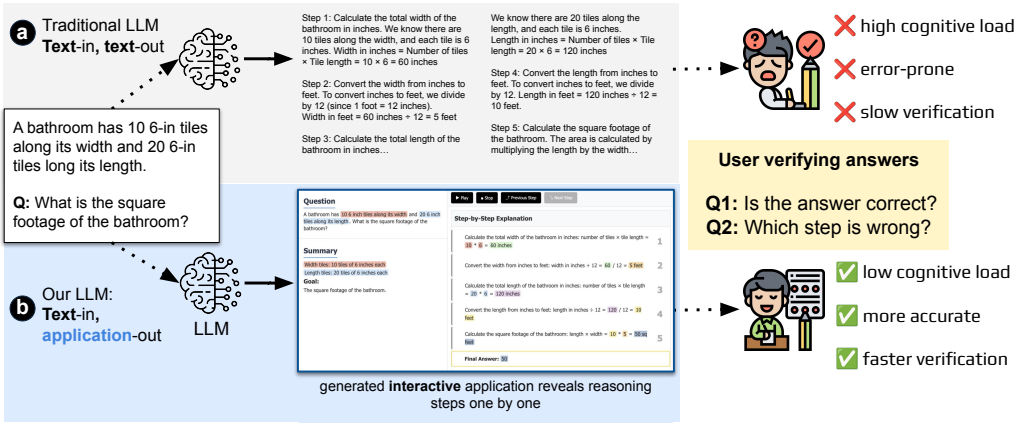

Figure 1: Given a GSM8K question, LLMs typically provide step-by-step reasoning followed by the final answer. However, such output presentation is often static and long, posing a higher cognitive load to users and leading to slower and more erroneous answer verification. In contrast, we prompt LLMs to *generate an interactive HTML/JavaScript application* wrapped around the reasoning. This interface enables users to verify the reasoning more efficiently via tools of (a) navigation buttons (inspired by those in common IDEs) and (b) colored highlights [8].

[1], CoT often appears as a long "*wall of text*" that can be time-consuming, obscure structure, increase cognitive load, and make error detection harder precisely when it matters. For example, when an explanation contains a subtle mistake in one intermediate step, detecting that error often requires carefully parsing multiple sentences and mentally reconstructing the reasoning flow [12]. More broadly, the explainable AI (XAI) literature shows that explanation techniques are frequently designed around what researchers can produce rather than what end users need; users instead report wanting practically useful information that improves collaboration, trust calibration, and feedback loops with AI systems [13–15]. This gap motivates rethinking how reasoning is surfaced to end-users in contexts like education, where clarity, verification, and engagement are essential.

Prior work has demonstrated that the design of LLM explanations, which includes factors such as visual structure, interactivity, and information hierarchy, can significantly influence users' trust and understanding [16–19]. However, most existing research on CoT focuses on improving the reasoning process within the model rather than optimizing the presentation of the reasoning [20, 21]. As a result, LLM explanations remain largely static, text-heavy, and difficult to navigate [22]. This disconnect raises critical questions: *How can explanation formats be redesigned to reduce cognitive load and enable users to effectively detect errors and understand LLM-generated reasoning steps?*

Present work. To address these questions, we propose an interactive explanation interface, an approach that explores interactive and structured alternatives to traditional text-based reasoning outputs as shown in Fig.1. Rather than presenting explanations as static and dense paragraphs, we introduce three novel formats designed to enhance clarity and interpretability: i) interactive CoT (iCoT), a format that preserves the text-based explanation but introduces features such as color-coded highlights for key steps, a clear step-by-step breakdown, and a concise summary for quick reference, ii) interactive graph (iGraph), a format where reasoning steps are represented as nodes connected by edges in a left-to-right flow diagram, allowing users to easily identify the logical progression of steps and visually detect errors or inconsistencies and iii) interactive Program-of-Thought (iPoT), a code-based format that presents reasoning as a sequence of pseudo-code lines. Each line represents a logical operation or variable update, showing how intermediate values lead to the final solution.

To evaluate the effectiveness of these formats, we conducted a controlled user study using problems from GSM8K, a multi-step mathematical reasoning benchmark. Each participant was randomly assigned to one of the four explanation formats, including a traditional text baseline, and asked to review ten LLM-generated (using Claude 3.7 Sonnet) explanations. Participants were tasked with determining whether each explanation was correct or incorrect. For incorrect explanations, the user has to identify the exact step where the error occurred. This experimental design allows us to measure two key performance metrics: verification accuracy, which reflects participants' ability to correctly

identify errors and locate faulty steps, and time per question, which indicates the efficiency of error detection. Finally, we performed a post-experiment study, where we took users' subjective feedback and measured the user engagement during the experiment. Our study aims to answer three central research questions:

RQ1) Do interactive explanation formats help users improve verification accuracy compared to traditional text outputs?

RQ2) How do these formats influence the time and effort required for users to verify reasoning?

RQ3) Do users prefer interactive explanation formats over traditional text-based outputs?

By systematically comparing these formats, we aim to uncover design principles that enhance the interpretability of LLM-generated explanations. Our work contributes not only a practical evaluation of interactive explanation designs but also broader implications for building trustworthy systems. When users can efficiently and accurately verify reasoning steps, they are better equipped to detect model errors, avoid over-reliance on incorrect outputs, and make informed decisions. These findings have potential applications in educational tools, auditing systems, and human-AI collaboration platforms, where understanding and validating reasoning is essential. In summary, this paper makes three primary contributions: i) we introduce interactive explanation frameworks, an approach that explores three alternative explanation formats (iCoT, iPoT, and iGraph) to improve the interpretability and usability of LLM reasoning; ii) our findings, from a controlled user study, show the effectiveness of iCoT, iPoT, and iGraph over traditional chain-of-thought on verification accuracy and time efficiency; and iii) we discuss design implications for developing interactive and interpretable AI systems that support user trust, error detection, and effective human-AI collaboration.

2 Related work

Here, we outline the purpose of CoT reasoning in LLMs through the lens of improving user understanding and the role of interactive presentation in supporting comprehension and engagement within educational contexts.

LLMs in Education with Chain-of-Thought (CoT) Prompting. A key prompting technique for LLMs is CoT prompting, which elicits the LLM's reasoning steps and improves its performance on multi-step tasks [23]. A recent study shows that few-shot CoT dramatically boosts arithmetic, commonsense, and symbolic reasoning accuracy [2, 3]. In educational settings, however, how explanations are presented matters. Despite clear opportunities for tutoring and feedback, Khosravi et al. [24] highlights persistent concerns about transparency, interpretability, and over-reliance in LLMs. Beyond transparency, CoT reasoning tends to be quite lengthy, which can increase cognitive load and slow down the user's comprehension [25]. Given the concerns about the interpretability and cognitive demands of LLM-generated reasoning, researchers have begun to explore how reasoning quality and user understanding can be evaluated more systematically. A growing body of work has adopted verification accuracy, which indicates the users' ability to correctly judge whether a model's reasoning is valid, as a key indicator of comprehension in LLM explanations [19, 26–29]. Further, a recent study argues that explanations are only useful when they enable users to verify predictions step-by-step, rather than simply accepting plausible reasoning at face value [30]. In our study, we aim to address this question by measuring how effectively people can detect reasoning errors across different explanation formats, allowing us to assess not only the quality of the reasoning itself but also how various interface designs influence users' ability to verify and understand it.

Interactivity in LLM Explanations. Recent studies have emphasized the importance of interactivity in AI explanations to improve deeper user engagement and understanding[31–33]. Teso et al. [34] provides a comprehensive overview of interactive machine learning, which shows that explanations are most effective when users can query, edit, or refine them rather than passively receive them. They argue that interaction transforms explanations from static outputs into dynamic learning tools that encourage users' cognitive engagement. Similarly, another work reviews human-centered explainability and finds that interactive explanation interfaces significantly enhance engagement, comprehension, and retention compared to one-way text outputs [35]. Their analysis shows that many existing AI systems remain static and fail to leverage interactive elements that foster user engagement and a deeper understanding of the AI reasoning process. In educational settings, a study demonstrates that interactive LLM-driven question–answer systems increase learners' focus and cognitive engagement

without introducing cognitive load [36]. Together, these studies highlight that interactivity in AI explanations not only improves the user's understanding but also keeps the users engaged.

Our work extends these insights by introducing interactive reasoning formats that move beyond passive text outputs. We explore how interactivity can make LLM reasoning both more interpretable and engaging by allowing users to navigate reasoning steps, inspect structure, and verify logic.

3 Method

3.1 Designing Alternative Response Formats for LLM Reasoning

The goal of this project is to design and evaluate alternative response formats for LLM-generated reasoning for math problems. While LLMs have demonstrated strong capabilities in producing step-by-step reasoning through CoT prompting, prior research shows that these outputs often are **lengthy, text-heavy passages** that are difficult for students to parse and verify in practice [37]. Therefore, we investigate whether alternative reasoning formats can enhance comprehension and reduce the cognitive load of users. To address these challenges, we design three alternative interactive reasoning formats that transform the same underlying content into different modes of presentation. We now introduce our three alternative explanation formats alongside the baseline explanation format.

Traditional Chain-of-Thought (CoT) serves as the baseline explanation format. It presents reasoning in a purely text-based form, where the model expresses its thought process step-by-step in natural language. Each step builds upon the previous one and forms a continuous logic flow that leads to the final solution[2]. The interface includes three main sections: the problem statement, the model's step-by-step explanation, and the final answer (see Fig. 2 A). This format represents the standard way large language models naturally generate explanations without additional structural guidance or interactivity.

Interactive Chain-of-Thought (iCoT) demonstrates a more traditional format but restructures the reasoning into discrete, highlighted segments (see Fig. 2 B). Each step is visually separated and includes embedded formulas and calculations, which are revealed sequentially using playback controls. This interface balances the linguistic reasoning with the structured presentation by avoiding long, uninterrupted paragraphs. Users can focus on one step at a time and interactively navigate the solution flow, which makes it easier to detect inconsistencies or mistakes.

Interactive Program-of-Thought (iPoT) shows explanations in a structured, code-like format(see Fig. 2 C). Each reasoning step is represented as a line of pseudo-code, where variables are defined, updated, and computed systematically. This format allows users to trace how each intermediate value is derived through reasoning steps. Playback controls enable users to execute the reasoning "step-by-step," similar to debugging or running a script. This interface design draws inspiration from the "Program of Thoughts Prompting" framework proposed by Chen et al. [38], which demonstrated that representing reasoning as executable programs can improve interpretability and consistency in LLMs.

Interactive Graph (iGraph) renders reasoning as a node-link diagram(see Fig. 2 D). Each node corresponds to either a problem variable, an intermediate value, or a final result. The edges represent operations connecting them. This approach aims to show the links between steps more clearly and help learners understand the solution through structured visuals.

The three interactive reasoning formats retain the content of a traditional CoT, restructure how it is presented to support verification, and reduce cognitive load. Each interactive explanation format follows a consistent interface template with specific design features to minimize variation in presentation. Next, we outline the design features of our interface to explain their functionality and relevance.

1) Dual-Panel Layout. All interactive explanations adopt a standardized dual-panel design. The left panel displays the problem statement and summary, while the right panel presents the step-by-step explanation. This separation ensures that users can always reference the original problem context **without scrolling or losing track** of the reasoning process.

2) Problem Summarization. A concise problem summary is displayed beneath the problem statement in the left panel. It highlights the key variables along with their numerical values, providing the users a quick reference to check against as they follow each reasoning step.

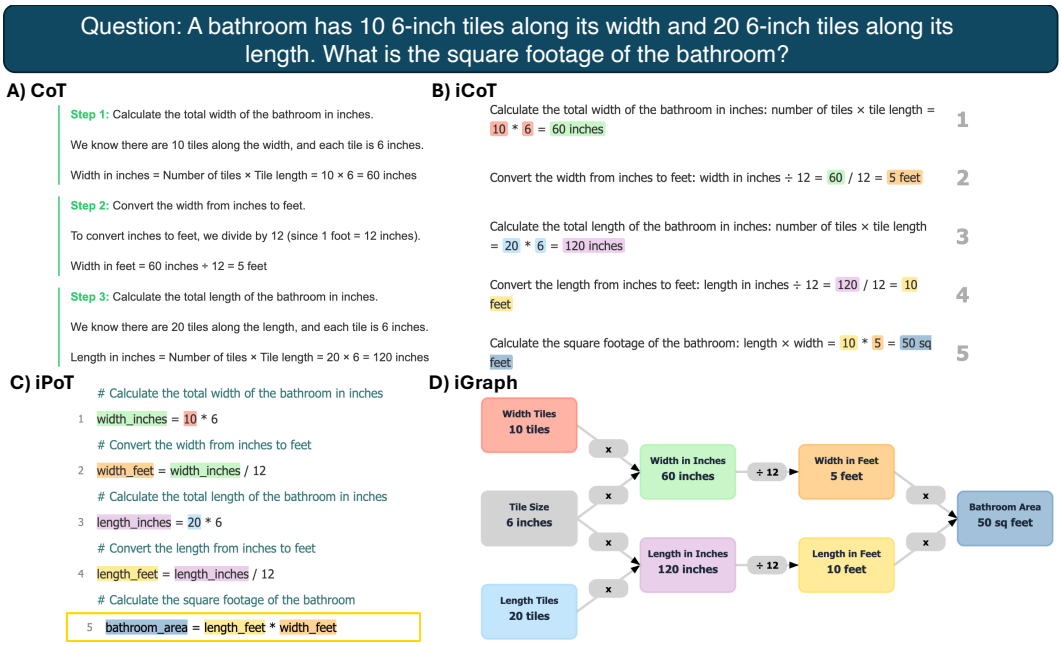

Figure 2: Examples of four explanation formats used in the study: (A) traditional Chain-of-Thought (CoT), (B) interactive Chain-of-Thought (iCoT), (C) interactive Program-of-Thought (iPoT), and (D) interactive Graph (iGraph). Each format presents the same reasoning steps in a different modality (textual, structured, code-like, or visual). For consistency, all four formats present the same mathematical problem. Complete interfaces for these four example explanation formats are displayed in Appendix A (Fig. 9,12).

3) Colored Variables. Variables are consistently color-coded across both panels. Each variable is assigned a unique color that is retained wherever it appears (problem statement, problem summary, or step-by-step explanation). This helps the users trace variables along with their values through the entire reasoning chain.

4) Step-by-Step Display with Controls. The explanation is displayed one step at a time in the right panel(except for the iPoT interface). Users can move forward and backward using playback control buttons. This design choice is intended to reduce cognitive overload and help users focus on the reasoning process incrementally rather than being overwhelmed by the entire solution at once.

3.2 Dataset Construction

To evaluate how alternative explanation formats aid users in detecting LLM-generated errors, we construct a dataset derived from GSM8K, a widely used benchmark of grade school math word problems [39]. GSM8K is chosen because it contains a large and diverse set of multi-step reasoning tasks that resemble problems students might encounter in classroom and tutoring settings. Each GSM8K problem typically involves 3–8 steps of reasoning with natural language explanations, which makes it well-suited for transformation into interactive formats.

A central aspect of our design is the controlled injection of errors into the explanations. Rather than relying on naturally occurring model mistakes, following Kamoi et al. [40], we adopt a taxonomy of nine error categories: Calculation (CA), Counting (CO), Context Value (CV), Contradictory Step (CS), Missing Step (MS), Hallucination (HA), Unit Conversion (UC), Operator (OP), and Formula Confusion (FC). By systematically injecting each of these error types, we creat a consistent and fine-grained framework that enable us to assess not only whether participants could recognize when an explanation is incorrect, but also whether certain categories of errors are more easily detected across different interface formats. For each error type, we randomly sample 50 problems from GSM8K. We ensure that each selected problem contains between four to seven reasoning steps. This range is chosen because shorter problems may make error detection trivial, while longer problems risk overburdening participants. Altogether, this process produced 450 erroneous explanations (9

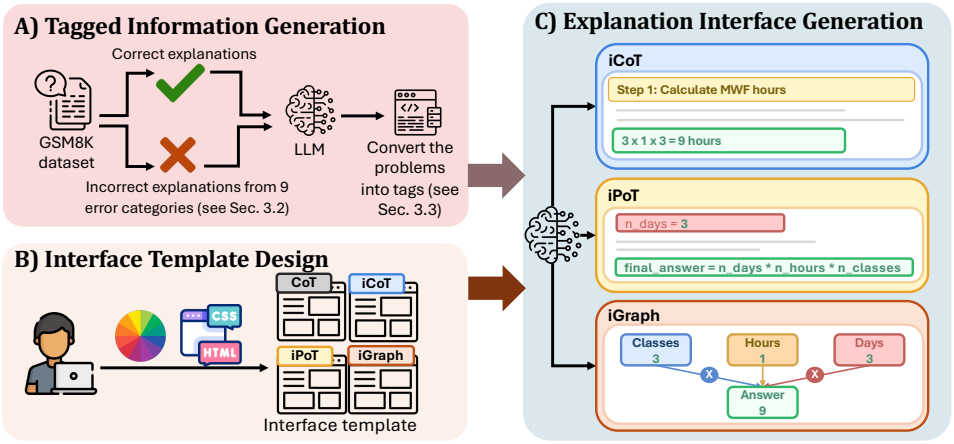

Figure 3: Our Explanation Generation pipeline consists of three stages: **(A)** Tagged Information Generation, where an LLM produces correct and erroneous GSM8K explanations annotated with reasoning tags; **(B)** Interface Template Design, where standardized HTML/CSS templates ensure consistent structure and interactivity across formats; and **(C)** Explanation Interface Generation, where tagged data and templates are combined to create interactive explanations in iCoT, iPoT, and iGraph formats for our user study.

error types × 50 examples each). To balance the dataset and establish a baseline for comparison, we also sample an additional 50 correct explanations from GSM8K. Like the erroneous cases, these problems also contain 4–7 reasoning steps in length and served to evaluate whether participants could correctly recognize valid reasoning when presented in different formats. By combining erroneous and correct cases, we obtain a dataset of 500 explanations in total.

3.3 Tagging as an Intermediate Representation

To ensure the interface formats are consistent, we develop an intermediate tagging layer that acts as a structured guideline for problem statements and explanations. Each problem is first encoded into tags for facts, steps, formulas, calculations, and variables. For example, **<fact>** elements capture problem parameters, **<step>** elements describe reasoning in natural language, **<formula>** captures symbolic relations, **<calculation>** contains explicit numeric operations, and **<var>** defines intermediate or final results. Wrong steps are annotated with **<wrongstep>**, which indicates the steps containing an injected error. The **<output>** tags specify the final solution. This tagging schema allows us to separate independent information. Each tagged problem can be rendered into the four interface formats without altering the underlying logic. Errors are introduced systematically by referencing the **<wrongstep>** index, allowing us to maintain strict control over consistency while still enabling interactive elements, like graph animations, code execution, and step playback.

3.4 Interface Templates for Consistency

While the tagging layer ensures that explanations across formats are generated from the same underlying content, an additional step is required to control for internal variations in raw LLM output. Specifically, we develop interface-specific templates because LLMs can produce responses with differences in phrasing, ordering, or formatting even when prompted identically. These templates define the layout, structure, and presentation rules for each explanation format, ensuring that all outputs within a format adhere to the same design principles. The templates serve two purposes: first, they **ensure a consistent user experience** within each format by standardizing how reasoning steps, variables, and intermediate values are presented; second, they **reduce the risk** that uncontrolled LLM variations would bias the evaluation. The templates allow us to isolate the effects of interface design from artifacts of model output by constraining visual and structural differences.

For the traditional CoT interface, the template fixes the structure of step-by-step textual explanations as sequential paragraphs with aligned formulas and calculations. For the iCoT format, the template segments explanations into discrete blocks with color-coded variable highlights and playback controls for navigating step by step. In the iPoT format, the template ensures that each reasoning step appears

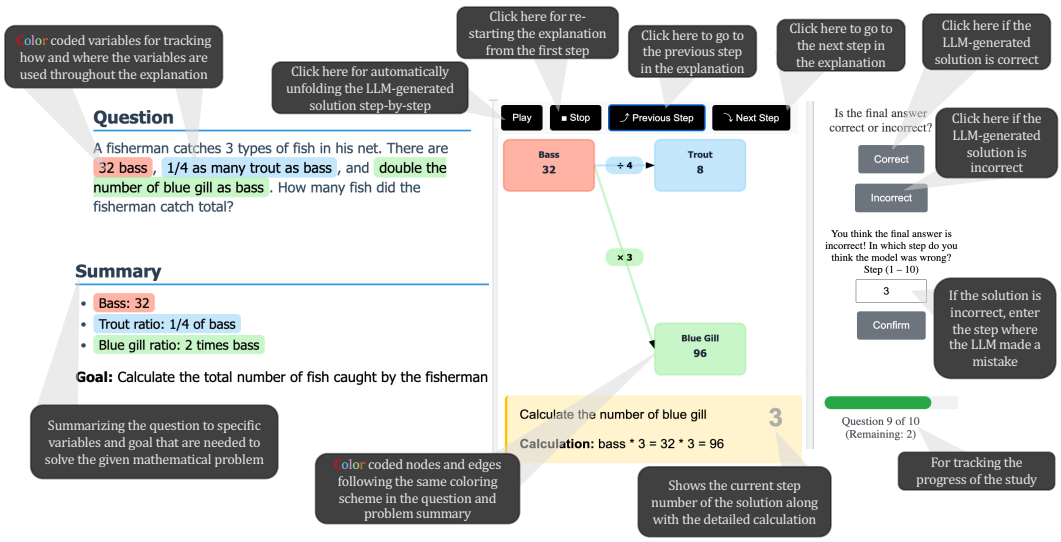

Figure 4: Our proposed interactive interface (using iGraph as an example) shows reasoning as connected nodes and edges, using consistent color-coded variables across the question, summary, and explanation. Interactive buttons let users follow each step, while a verification panel allows them to mark whether the explanation is correct and identify any errors. See Appendix A for more examples of Chain-of-Thought (CoT), Interactive Program-of-Thought (iPoT), Interactive Chain-of-Thought (iCoT) and IGraph explanations.

as a line of pseudocode with comments and that a synchronized variable panel updates consistently across explanations. For the iGraph interface, the template standardizes the node-link arrangement, left-to-right flow, and inclusion of textual support beneath the graph.

3.5 Data Flow and Interface Generation

Our proposed pipeline from problem input to rendered explanation involves two key stages, as illustrated in Fig.3. First, each math problem and its solution are passed to the LLM (Claude 3.7 Sonnet [41] in all our experiments), which produces a tagged intermediate representation. This stage ensures that the reasoning content is captured in a structured format. In the second stage, the tagged representation is combined with the interface-specific templates, where the tags provide the logical content and the templates enforce layout rules and structural consistency. This combined input is then passed to another LLM, which generated the final rendered interface in one of four formats: traditional CoT, iCoT, iPoT, or iGraph.

Our two-stage process allowed us to standardize explanation content through tagging before rendering, thereby eliminating uncontrolled differences that naturally occur from direct LLM outputs. Further, it modularizes the workflow by separating tags from templates, which ensures that the same reasoning can be faithfully expressed across multiple interface modalities. The result is a controlled and reproducible pipeline in which variations in participant performance could be attributed to interface design, rather than inconsistencies in problem or explanation generation.

4 User Study

We conduct a large-scale user study to evaluate how different explanation formats influence interpretability, error detection, and engagement when reviewing LLM-generated math explanations. The study follows a between-subjects design, where each participant was randomly assigned to one of four interface formats (CoT, iCoT, iPoT, or iGraph). IGraph interface is displayed as an example in Fig. 4 with detailed explanation for each interface design features. Each participant reviews ten explanations presented in their assigned interface format. The set of explanations contains both correct and incorrect solutions to grade-school math problems from the GSM8K dataset, with errors systematically injected. One hundred and twenty-nine undergraduate students ($n = 129$) from

two U.S. universities are recruited through class announcements. After the experiment, we plot a histogram showing how long participants take to answer each question. Data from four participants are excluded because their completion times are either unusually short or excessively long, suggesting that they may have rushed through the task or become distracted. Including these cases can bias both the time and the accuracy results. After the screen, the results from 125 participants are used for later analysis. All participants have completed at least one mathematics course beyond high school. Participation is voluntary, and students are informed that their answers will remain completely anonymous.

4.1 Setup and Procedure

The study is conducted in a controlled classroom setting. Each participant accessed the experiment through an external link and was provided with both written and verbal instructions. The experiment is deployed on HuggingFace Spaces, which hosts both the frontend and backend within a unified environment. This cloud-based setup allows participants to run the study directly in a standard web browser without requiring any local installation or configuration.

On the front end, we let LLM pre-generate a total of 2000 explanation interfaces in HTML format, which corresponds to 300 explanations in each of the four interface formats. These HTML interfaces are then stored on HuggingFace and accessed during the study. A separate experiment wrapper is implemented in HTML, which embeds the explanation interface files through an iframe. This wrapper provides the environment in which participants complete their tasks, and it contains additional experimental features. Specifically, the wrapper presents the (in)correct buttons for judging correctness. A text box is also added for participants to input the specific step at which the error occurs. A progress bar is implemented to inform participants of the number of explanations they have completed, helping them keep track of their progress during the experiment. On the backend, HuggingFace records participants' responses and interaction data. Each submission is anonymized and stored in a HuggingFace Dataset, which also contains automatically logged statistics such as verification accuracy, time spent on each question, and the full click history of button interactions. These logs are then converted into structured JSON files and stored within the dataset for later analysis.

Experiment. The experiment contains three stages. First, participants are introduced to the study and informed of its goals, procedures, and their right to withdraw at any time without penalty. Consent is obtained before proceeding. Each participant is then randomly assigned to one of the four explanation formats and shown a sequence of ten explanations drawn from the LLM pre-generated explanation pool. Each sequence contains one correct explanation and nine erroneous explanations, each corresponding to a distinct error type. For each explanation, participants are asked to determine its correctness, and if incorrect, to identify the specific error step. Participants can use the playback controls to revisit previous steps within a single explanation, but once their judgment is submitted, they can not return to revise their answers. There is no time limit imposed on individual questions. Finally, after completing all ten explanations, participants are asked to complete a brief survey reflecting on their overall experience, including ratings of clarity, usability, and engagement.

4.2 Evaluation Metrics

We collect quantitative and qualitative measures to capture both performance outcomes and user experiences: *i) Verification Accuracy [19]:* For each explanation, we record whether participants correctly labeled the solution as correct or incorrect, providing a direct indicator of how effectively each interface supports error detection and overall comprehension; *ii) Error Localization [40]:* In cases where the user identifies an explanation as incorrect, they are required to indicate the specific step at which the error occurred. This allows us to assess not only error detection but also participants' ability to pinpoint the exact reasoning breakdown (avoiding the user answering by guessing); *iii) Response Time [8]:* The system automatically logs the total time spent on each explanation, which we use as a proxy for cognitive effort, with longer times potentially indicating greater difficulty in comprehension or in identifying error; *iv) Interaction Behavior [42]:* All interface interactions, including play, pause, step forward, and step back, are recorded at the click level. These logs gave us insights into how participants navigate the explanations; and *v) Subjective Feedback [43]:* After completing the tasks, participants rate their assigned interface on dimensions such as clarity, ease of use, and overall satisfaction using Likert-scale questions. They are also invited to provide open-ended feedback about what they found helpful or confusing, helping us understand participants' perceptions of strengths and weaknesses in each format.

4.3 Post survey Questionnaire

After the experiment, participants will complete 2 questionnaires displayed in Appendix B to assess both their interaction experience and their perceptions of the explanation quality. The questionnaires are designed to address the study's research questions regarding interpretability, efficiency, and engagement across different explanation formats. All items use a 7-point Likert scale (1 = Strongly Disagree, 7 = Strongly Agree) to capture nuanced user perceptions. Two types of questionnaires are given: *Design Feature* and a *Post-Study*. The question sets are inspired by established measures of engagement, usability, and interpretability in human-AI interaction research and are adapted for the design features of our interfaces [43–45].

Post-Study Questionnaire. All participants will complete a post-study questionnaire after finishing all explanation tasks. This evaluates overall satisfaction, perceived clarity of explanations, and perceived learning effectiveness (see Appendix B, Figure 13). Participants rate more general statements such as: "*This explanation format helped me understand the reasoning behind the solution*" or "*I found the interface engaging to use.*" For the interactive conditions, the questionnaire also captures perceived smoothness of interaction and cognitive effort compared to traditional text-based formats. The CoT format includes only the post-study questionnaire, as it does not include interactive elements that allow feature-specific evaluation. Responses across all questionnaires are analyzed to understand how different interaction mechanisms affect interpretability, efficiency, and user engagement.

Design Feature Questionnaires. Participants assigned to one of the interactive formats (iCoT, iPoT, or iGraph) will complete a design feature questionnaire immediately after finishing their tasks. These questionnaires aim to assess user perceptions of key design components introduced in the interactive interfaces, including the dual-panel layout, problem summarization, color-coded variables, and step-by-step playback controls (see Appendix B, Figure 14). Each feature is evaluated through one targeted question, *e.g.,* participants are asked whether the dual-panel layout helped them maintain context or whether the playback controls support reasoning flow at their own pace. The design feature questionnaire allows us to capture which design aspects contributed most strongly to participants' perceived interpretability and usability.

5 Result & Discussion

Next, we present experimental results for our user studies and address the research questions introduced in Sec. 1.

5.1 Quantitative Result

RQ1) Interactive explanations improve verification accuracy. A Kruskal–Wallis ANOVA is conducted to examine whether verification accuracy differed significantly across the four explanation formats (CoT, iCoT, iPoT, and iGraph). The results indicate a statistically significant difference among the groups ($H(3) = 13.12$, $p = 0.004$).

To further examine differences in verification accuracy between the traditional CoT format and each of the interactive explanation formats, we perform pairwise Mann–Whitney U tests with Bonferroni correction to account for multiple comparisons. The analysis shows that **iCoT(**$U = 323.5, p = 0.047, r = 0.32$**), iPoT (**$U = 306.0, p = 0.023, r = 0.38$**), and iGraph (**$U = 221.5, p = 0.003, r = 0.51$**) all achieve significantly higher verification accuracy compared to the traditional CoT** format. However, differences between the interactive formats themselves (iCoT, iPoT, iGraph) are not statistically significant after Bonferroni correction ($p > 0.05$).

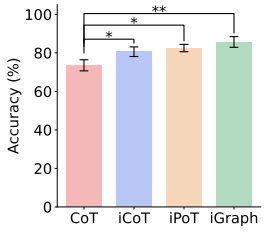

Figure 5: Verification accuracy. Results show that participants achieve the highest verification accuracy using iGraph.

Results in Fig. 5 demonstrate that interactive explanation formats substantially improve participants' ability to verify reasoning accuracy compared to the traditional CoT. In particular, the iGraph interface achieves the highest verification accuracy (85.6%), followed by the iPoT (82.5%) and iCoT (80.6%) formats, whereas CoT only achieves 73.5%. This pattern indicates that interactive explanation formats help users better detect and localize logical faults. These findings align with recent work emphasizing the

importance of human-centered explainability, which explains understanding as a process of active answer-seeking rather than passive reception [46].

Next, we use a Kruskal–Wallis ANOVA to examine whether participants' ability to identify the specific step containing an error varies across the four explanation formats. The analysis reveals a statistically significant difference among formats ($H(3) = 13.12, p = 0.004$). Pairwise Mann–Whitney U tests with Bonferroni correction show that **iCoT**($U = 300.8, p = 0.045$)**, iPoT** ($U = 271.5, p = 0.021$)**, and iGraph** ($U = 153.0, p < 0.001$) **interfaces all achieve significantly higher wrong-step identification accuracy compared to the traditional CoT** format. Results in Fig. 6 show that IGraph has achieved the highest wrong-step identification accuracy (85.2%), followed by iPoT (80.1%) and iCoT (79.3%), compared to only 66.1% reached by traditional CoT. The higher wrong-step identification accuracy on iGraph further suggests that interactive explanations helped users understand each step more thoroughly. Prior works in explainable artificial intelligence (XAI) argue that unstructured explanations often overwhelm users with text and often make it hard for users to keep track of each reasoning step [47]. Our findings support these arguments by showing that interactive explanations allow users to accurately identify the origin of an error, rather than simply noticing that one exists. Further, visualizing reasoning steps as interconnected nodes and edges transforms purely textual information into a two-dimensional spatial format, making the logic flow easier to follow and understand (this is further verified by our post survey rating results in Fig. 8).

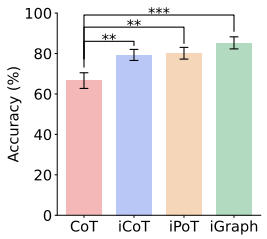

Figure 6: Error localization. Results show that participants were able to detect the exact error step in the LLM's explanation with higher accuracy using iGraph.

RQ2) Participants using interactive explanations respond faster on average, but the time differences are not statistically significant. We also use Kruskal–Wallis ANOVA to see if the time taken by participants to evaluate explanations significantly differed across the four interface formats.

The test shows a significant overall difference ($H(3) = 13.12, p = 0.004$), which suggests that the interface type has some effect on how much time participants spend on completing each task. However, the follow-up Mann–Whitney U tests with Bonferroni correction show that none of the pairwise comparisons reach statistical significance. Results in Fig. 7 show an interesting pattern, where participants assigned to the traditional CoT format spend the most time on each question ($\mu = 64.7$, $\sigma = 40.9$), but participants assigned to the iGraph interface have the fastest response time ($\mu = 57.9, \sigma = 41.4$). Participants who are assigned to the iPoT ($\mu = 60.1, \sigma = 37.9$) and iCoT ($\mu = 59.5, \sigma = 35.8$) interfaces spend slightly less time compared to traditional CoT. This trend hints that interactive formats may have helped participants interpret explanations more efficiently, even if the differences aren't statistically significant.

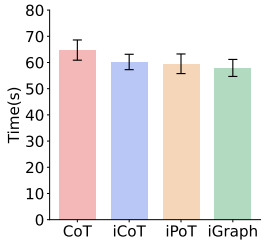

Figure 7: Response time. Results show that participants were able to respond faster using iGraph.

Across all quantitative measures, the data show that interactive explanation formats consistently improved verification-related performance and completion time. In our interfaces, features such as playback controls and problem summaries allow participants to focus on relevant steps rather than reading the entire explanation.

5.2 Qualitative Result

Participants complete a post-experiment questionnaire to evaluate their understanding, engagement, satisfaction, and perceptions of design effectiveness across the four explanation formats. While all four formats are assessed on general measures of comprehension and engagement (G1–G5), only the three interactive formats included additional design-specific questions (D1–D4). Using the responses from these questions, we answer the research question: "*Do users prefer interactive explanation formats over traditional text-based outputs?*"

1) General Measures Assessment. Across the first two questions, assessing the interface's ability to improve **user understanding (G1)** and **user ability to identify errors (G2)**, participants consistently rate iGraph the highest (Fig. 8; see plot C), suggesting that the iGraph interface is most effective in helping users understand the explanations and follow the reasoning process. In contrast, both CoT

and iCoT receive lower scores on these items, meaning text-based formats are less helpful for understanding and error detection compared to the visual and structured formats(Fig. 8; see plots B and D).

The third question directly assesses **self-perceived engagement(G3)** during the experiment. All interactive interfaces achieved strong ratings for engagement (G3) compared to CoT, where a large proportion of participants gave low ratings. The results show that iPoT receives the highest engagement rating, followed by iCoT and iGraph, suggesting that participants find the structured, code-like format of iPoT more engaging(Fig. 8; see plot A-C). Its explicit variable tracking and step-by-step logic may have helped users feel more in control of the reasoning process, which is consistent with research discussing that structured and rule-based explanations promote active engagement and comprehension [48]. Our interfaces foster engagement by allowing users to control reading pace with interactive elements such as playback control.

RQ3) Users prefer interactive interfaces (iGraph, iPoT, iCoT) over the traditional CoT format. The last two questions measure **future use preference (G4)** and **overall satisfaction (G5)**. Overall, participants give iGraph the highest satisfaction, while iCoT is the preferred explanation format to study math(Fig. 8; see plot B and C). This pattern indicates that the **interactive and visually organized formats are perceived as more engaging and enjoyable learning experiences** compared to the static CoT baseline(Fig. 8; see plot D). While iPoT receives high engagement scores, its syntactic structure may have introduced additional interpretive effort for some participants who are not familiar with interpreting coding-based explanations(Fig. 8; see plot A). In contrast, participants consistently rate the traditional CoT interface **lowest** across both questions, suggesting that long textual explanations are less effective at interpreting math reasoning problems.

2) Design Feature Evaluation. Participants who interact with iGraph, iPoT, and iCoT also rate four design-specific features: **dual-panel layout (D1)**, **problem summarization (D2)**, **color-coded variables (D3)**, and **step-by-step presentation (D4)**. The design feature evaluation (D1–D4) is developed to isolate and evaluate how specific interactive design elements within the explanation interfaces contributed to users' comprehension and engagement.

For D1 (dual-panel layout), most participants think that viewing the problem on the left while seeing the explanation on the right makes it easier to connect reasoning steps to the original problem. Participants who are assigned to the iGraph interface show the strongest positive sentiment for this design, with 85% of participants giving neutral or positive ratings, followed by iCoT with 79% and iPoT with 74% (Fig. 8; see plot E-G). This dual representation reduces the extraneous load by **eliminating the need to scroll and reorient**. Our design aligns with the results from Sweller's study, which claims that working memory has a limited capacity and instructional designs should aim to minimize unnecessary mental effort [49].

In D2 (problem summarization), most participants find that concise summaries helped them identify critical variables and constraints before reading the explanation. The interface allows users to focus on understanding the reasoning rather than recalling what each variable represents. iGraph interface has received the most positive rating with 85% of participants giving a neutral or positive rating, followed by iCoT with 74%, and iPoT with 69% (Fig. 8; see plot E-G). This design is inspired by a recent study, which suggests that providing learners with key information in advance helps them form a mental model before engaging with complex reasoning [53].

In D3 (color coding), participants reported that consistent color coding across panels helped them trace variables throughout multi-step reasoning processes. iGraph interface received the most positive feedback on this feature, with 85% participants giving neutral or positive rating, followed by iPoT with 81% and iCoT with 74% (Fig. 8; see plot E-G). This feature draws inspiration from a recent study, which links reasoning steps to supporting facts through consistent highlighting [8]. Each variable in our interface is assigned a distinct color that persists across the problem statement, summary, and explanation. This visual consistency aims to help users trace relationships between inputs and reasoning steps.

Finally, D4 (step-by-step presentation) also received uniformly high ratings across three interactive interfaces. iGraph again has received the most positive rating with 97% participants giving neutral or positive rating, followed by iCoT with 82% (Fig. 8; see plot F and G). However, only 57% of participants who are assigned to iPoT give a neutral or positive rating (Fig. 8; see plot E). This step-by-step reveal design is inspired by the principle of progressive disclosure, which argues that gradually exposing information helps reduce cognitive load and avoid overwhelming users [54]. In the XAI

Figure 8: Post Survey Questionnaire results for comparing different explanation formats using utility questions (G1-G5) and design-based questions for iPoT, iCoT, and iGraph (D1-D4). Results from the utility questions show that participants find the interactive formats more effective and engaging, reporting improvements in understanding, error detection, engagement, preference, and overall satisfaction (G1–G5) compared to the traditional CoT. Similarly, results from the design-based questions indicate that participants across all interactive formats consider all four design elements helpful during the experiment. The visualization follows methodology from [50–52]. For the general measures assessment, we find that iGraph (A) achieves the highest rating as compared to CoT (B), iCoT (C).

literature, interactive and incremental explanation designs are recognized as more human-friendly than monolithic explanation dumps, as they allow users to control over depth and pacing [55].

6 Conclusion and Future Work

Traditional CoT explanations in LLMs often appear as long blocks of text that make it difficult for learners to trace reasoning or identify mistakes. To address this limitation, we develop three interactive explanations–iGraph, iPoT, and iCoT–to improve the interpretability and engagement of LLM explanations for math problem-solving. Through a large-scale user study with 125 participants, we find that interactive formats significantly improve users' ability to verify explanations and locate reasoning errors. Overall, the graph interface achieves the highest verification accuracy, which indicates that visually structured reasoning supports a more intuitive understanding of logical dependencies. Our work provides an empirical foundation for designing interactive reasoning systems that foster deeper engagement and better understanding of AI-generated explanations. However, the study also highlights trade-offs such as visual complexity in graphs, syntactic demands in code, and sequential navigation in text, all of which require careful design calibration. Future research should explore adaptive explanation systems that personalize interactivity based on users' cognitive preferences and learning styles.

References

[1] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. 1, 2

[2] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 1, 3, 4

[3] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022. 1, 3

[4] Shen Wang, Tianlong Xu, Hang Li, Chaoli Zhang, Joleen Liang, Jiliang Tang, Philip S Yu, and Qingsong Wen. Large language models for education: A survey and outlook. *arXiv preprint arXiv:2403.18105*, 2024.

[5] Ali Forootani. A survey on mathematical reasoning and optimization with large language models. *arXiv preprint arXiv:2503.17726*, 2025. 1

[6] Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.153. URL https://aclanthology.org/2023.acl-long.153/. 1

[7] Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. Faithfulness vs. plausibility: On the (un) reliability of explanations from large language models. *arXiv preprint arXiv:2402.04614*, 2024. 1

[8] Tin Nguyen, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Hot: Highlighted chain of thought for referencing supporting facts from inputs. *arXiv preprint arXiv:2503.02003*, 2025. 2, 8, 11

[9] Jenny Kunz and Marco Kuhlmann. Properties and challenges of llm-generated explanations. *arXiv preprint arXiv:2402.10532*, 2024. 1

[10] Elias Hedlin, Ludwig Estling, Jacqueline Wong, Carrie Demmans Epp, and Olga Viberg. Got it! prompting readability using chatgpt to enhance academic texts for diverse learning needs. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference*, pages 115–125, 2025.

[11] Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023. 1

[12] Venkatesh Mishra, Bimsara Pathiraja, Mihir Parmar, Sat Chidananda, Jayanth Srinivasa, Gaowen Liu, Ali Payani, and Chitta Baral. Investigating the shortcomings of llms in step-by-step legal reasoning. *arXiv preprint arXiv:2502.05675*, 2025. 2

[13] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019. 2

[14] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y Lim, and Mohan Kankanhalli. Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pages 1–18, 2018.

[15] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–14, 2020. 2

[16] Fumeng Yang, Zhuanyi Huang, Jean Scholtz, and Dustin L Arendt. How do visual explanations foster end users' appropriate trust in machine learning? In *Proceedings of the 25th international conference on intelligent user interfaces*, pages 189–201, 2020. 2

[17] Gulsum Alicioglu and Bo Sun. A survey of visual analytics for explainable artificial intelligence methods. *Computers & Graphics*, 102:502–520, 2022.

[18] Peiling Jiang, Jude Rayan, Steven P Dow, and Haijun Xia. Graphologue: Exploring large language model responses with interactive diagrams. In *Proceedings of the 36th annual ACM symposium on user interface software and technology*, pages 1–20, 2023.

[19] Giang Nguyen, Ivan Brugere, Shubham Sharma, Sanjay Kariyappa, Anh Totti Nguyen, and Freddy Lecue. Interpretable llm-based table question answering. *arXiv preprint arXiv:2412.12386*, 2024. 2, 3, 8

[20] Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. *Advances in Neural Information Processing Systems*, 37:66383–66409, 2024. 2

[21] Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. *arXiv preprint arXiv:2212.10001*, 2022. 2

[22] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38, 2024. 2

[23] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*, 2022. 3

[24] Hassan Khosravi, Simon Buckingham Shum, Guanliang Chen, Cristina Conati, Yi-Shan Tsai, Judy Kay, Simon Knight, Roberto Martinez-Maldonado, Shazia Sadiq, and Dragan Gašević. Explainable artificial intelligence in education. *Computers and education: artificial intelligence*, 3:100074, 2022. 3

[25] Sania Nayab, Giulio Rossolini, Marco Simoni, Andrea Saracino, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. Concise thoughts: Impact of output length on llm reasoning and cost. *arXiv preprint arXiv:2407.19825*, 2024. 3

[26] Menaka Narayanan, Emily Chen, Jeffrey He, Been Kim, Sam Gershman, and Finale Doshi-Velez. How do humans understand explanations from machine learning systems? an evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1802.00682*, 2018. 3

[27] Rhema Linder, Sina Mohseni, Fan Yang, Shiva K Pentyala, Eric D Ragan, and Xia Ben Hu. How level of explanation detail affects human performance in interpretable intelligent systems: A study on explainable fact checking. *Applied AI Letters*, 2(4):e49, 2021.

[28] Mahsan Nourani, Chiradeep Roy, Tahrima Rahman, Eric D Ragan, Nicholas Ruozzi, and Vibhav Gogate. Don't explain without verifying veracity: an evaluation of explainable ai with video activity recognition. *arXiv preprint arXiv:2005.02335*, 2020.

[29] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J Gershman, and Finale Doshi-Velez. Human evaluation of models built for interpretability. In *Proceedings of the AAAI conference on human computation and crowdsourcing*, volume 7, pages 59–67, 2019. 3

[30] Raymond Fok and Daniel S Weld. In search of verifiability: Explanations rarely enable complementary performance in ai-advised decision making. *AI Magazine*, 45(3):317–332, 2024. 3

[31] Byung Hyung Kim, Seunghun Koh, Sejoon Huh, Sungho Jo, and Sunghee Choi. Improved explanatory efficacy on human affect and workload through interactive process in artificial intelligence. *IEEE Access*, 8:189013–189024, 2020. 3

[32] Mouadh Guesmi, Mohamed Amine Chatti, Shoeb Joarder, Qurat Ul Ain, Rawaa Alatrash, Clara Siepmann, and Tannaz Vahidi. Interactive explanation with varying level of details in an explainable scientific literature recommender system. *International Journal of Human–Computer Interaction*, 40(22):7248–7269, 2024.

[33] Giang Nguyen, Mohammad Reza Taesiri, Sunnie SY Kim, and Anh Nguyen. Allowing humans to interactively guide machines where to look does not always improve human-ai team's classification accuracy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8169–8175, 2024. 3

[34] Stefano Teso, Öznur Alkan, Wolfgang Stammer, and Elizabeth Daly. Leveraging explanations in interactive machine learning: An overview. *Frontiers in Artificial Intelligence*, 6:1066049, 2023. 3

[35] Yuhao Zhang, Jiaxin An, Ben Wang, Yan Zhang, and Jiqun Liu. Human-centered explainability in interactive information systems: A survey. *arXiv preprint arXiv:2507.02300*, 2025. 3

[36] Suleyman Ozdel, Can Sarpkaya, Efe Bozkir, Hong Gao, and Enkelejda Kasneci. Examining the role of llm-driven interactions on attention and cognitive engagement in virtual classrooms. *arXiv preprint arXiv:2505.07377*, 2025. 4

[37] Vagelis Plevris, George Papazafeiropoulos, and Alejandro Jiménez Rios. Chatbots put to the test in math and logic problems: A comparison and assessment of chatgpt-3.5, chatgpt-4, and google bard. *Ai*, 4(4):949–969, 2023. 4

[38] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*, 2022. 4

[39] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 5

[40] Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Ranran Haoran Zhang, Sujeeth Reddy Vummanthala, et al. Evaluating llms at detecting errors in llm responses. *arXiv preprint arXiv:2404.03602*, 2024. 5, 8

[41] Anthropic. Anthropic. URL https://assets.anthropic.com/m/785e231869ea8b3b/original/claude-3-7-sonnet-system-card.pdf. 7

[42] Jeroen Ooge, Shotallo Kato, and Katrien Verbert. Explaining recommendations in e-learning: Effects on adolescents' trust. In *Proceedings of the 27th International Conference on Intelligent User Interfaces*, pages 93–105, 2022. 8

[43] Sunnie SY Kim, Elizabeth Anne Watkins, Olga Russakovsky, Ruth Fong, and Andrés Monroy-Hernández. " help me help the ai": Understanding how explainability can support human-ai interaction. In *proceedings of the 2023 CHI conference on human factors in computing systems*, pages 1–17, 2023. 8, 9

[44] Yao Rong, Tobias Leemann, Thai-Trang Nguyen, Lisa Fiedler, Peizhu Qian, Vaibhav Unhelkar, Tina Seidel, Gjergji Kasneci, and Enkelejda Kasneci. Towards human-centered explainable ai: A survey of user studies for model explanations. *IEEE transactions on pattern analysis and machine intelligence*, 46(4):2104–2122, 2023.

[45] Leming Zhou, Jie Bao, I Made Agus Setiawan, Andi Saptono, and Bambang Parmanto. The mhealth app usability questionnaire (mauq): development and validation study. *JMIR mHealth and uHealth*, 7(4):e11500, 2019. 9

[46] Jenia Kim, Henry Maathuis, and Danielle Sent. Human-centered evaluation of explainable ai applications: a systematic review. *Frontiers in Artificial Intelligence*, 7:1456486, 2024. 10

[47] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. Faithful and customizable explanations of black box models. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 131–138, 2019. 10

[48] Q Vera Liao and Kush R Varshney. Human-centered explainable ai (xai): From algorithms to user experiences. *arXiv preprint arXiv:2110.10790*, 2021. 11

[49] John Sweller. Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2):257–285, 1988. 11

[50] Peter Andrews, Oda Elise Nordberg, Stephanie Zubicueta Portales, Njål Borch, Frode Guribye, Kazuyuki Fujita, and Morten Fjeld. Aicommentator: A multimodal conversational agent for embedded visualization in football viewing. In *Proceedings of the 29th International Conference on Intelligent User Interfaces*, pages 14–34, 2024. 12

[51] Peter Andrews, Njål Borch, and Morten Fjeld. Aimoderator: A co-pilot for hyper-contextualization in political debate video. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pages 137–156, 2025.

[52] Chen Zhu-Tian, Qisen Yang, Jiarui Shan, Tica Lin, Johanna Beyer, Haijun Xia, and Hanspeter Pfister. iball: Augmenting basketball videos with gaze-moderated embedded visualizations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2023. 12

[53] Albert Fenteng. Online learning: A cognitive tool for learning, an alternative to traditional learning style. *Psychology*, 14(5):676–686, 2023. 11

[54] Aaron Springer and Steve Whittaker. Progressive disclosure: designing for effective transparency. *arXiv preprint arXiv:1811.02164*, 2018. 11

[55] Frank Spillers. Progressive disclosure, 2010. 12

A Interface Displays

This appendix presents examples of all four explanation interface formats evaluated in our study: iGraph, iCoT, iPoT, and CoT.

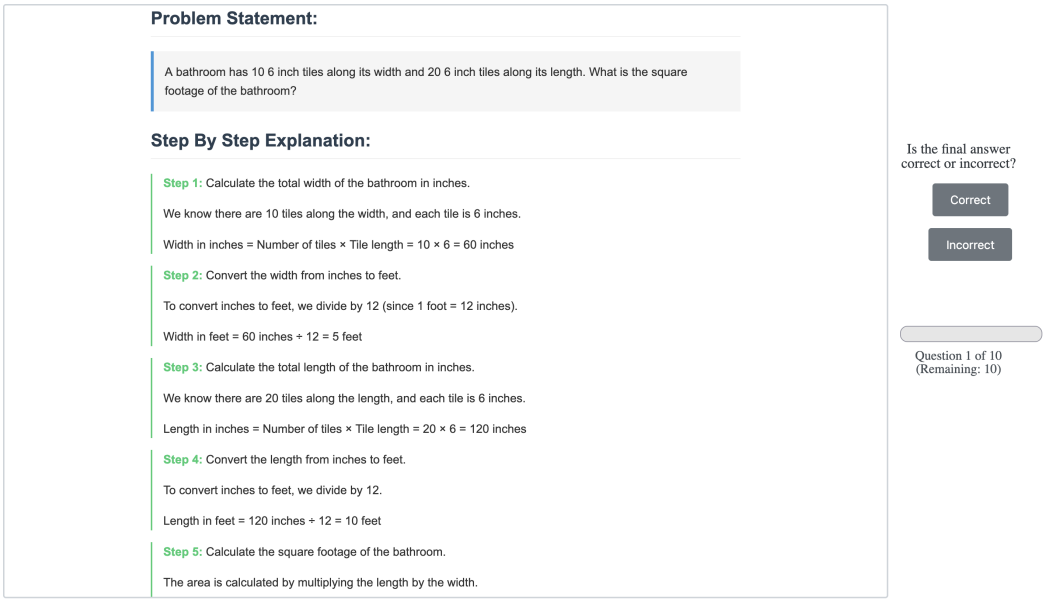

Figure 9: CoT interface

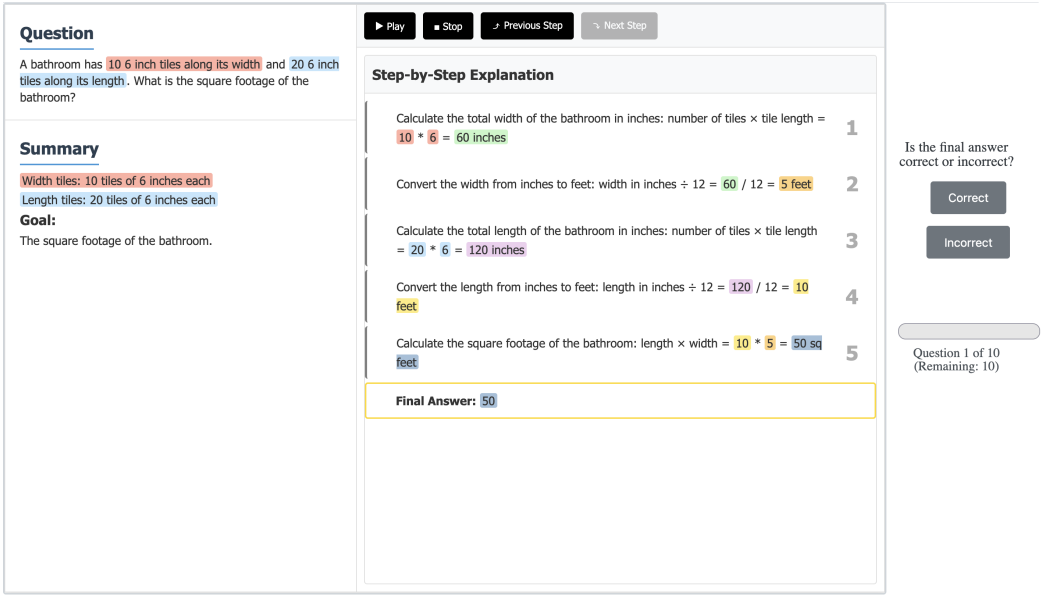

Figure 10: iCoT interface

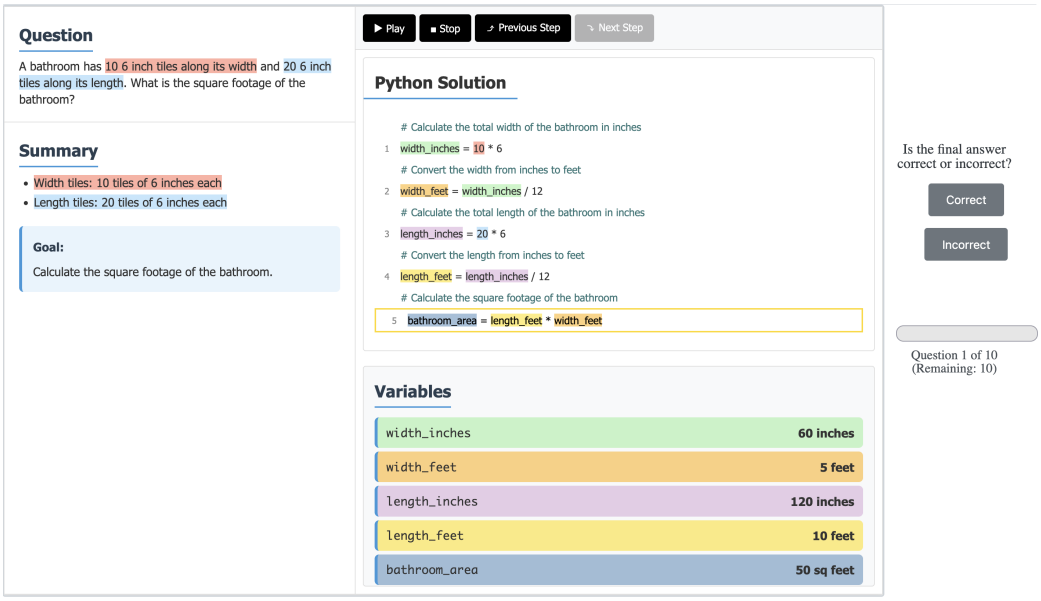

Figure 11: iPoT interface

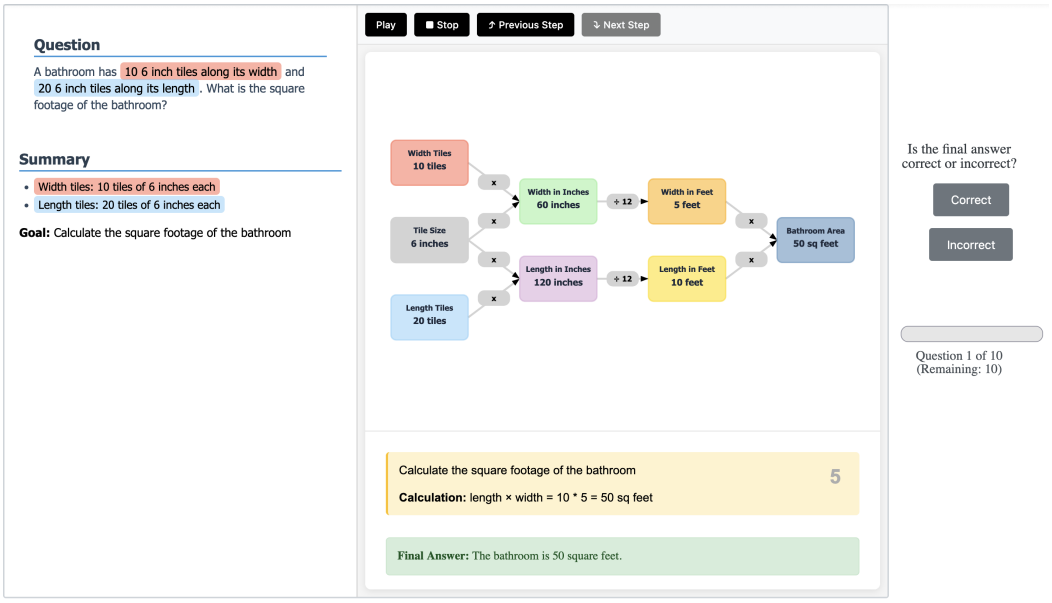

Figure 12: iGraph interface

B Questionnaire Display

This appendix presents two post-study questionnaires used in the experiment. The Post-Study Questionnaire focuses on assessing overall engagement and satisfaction with the explanation format, while the Design Features Questionnaire focuses on evaluating specific design features of the interactive interfaces.

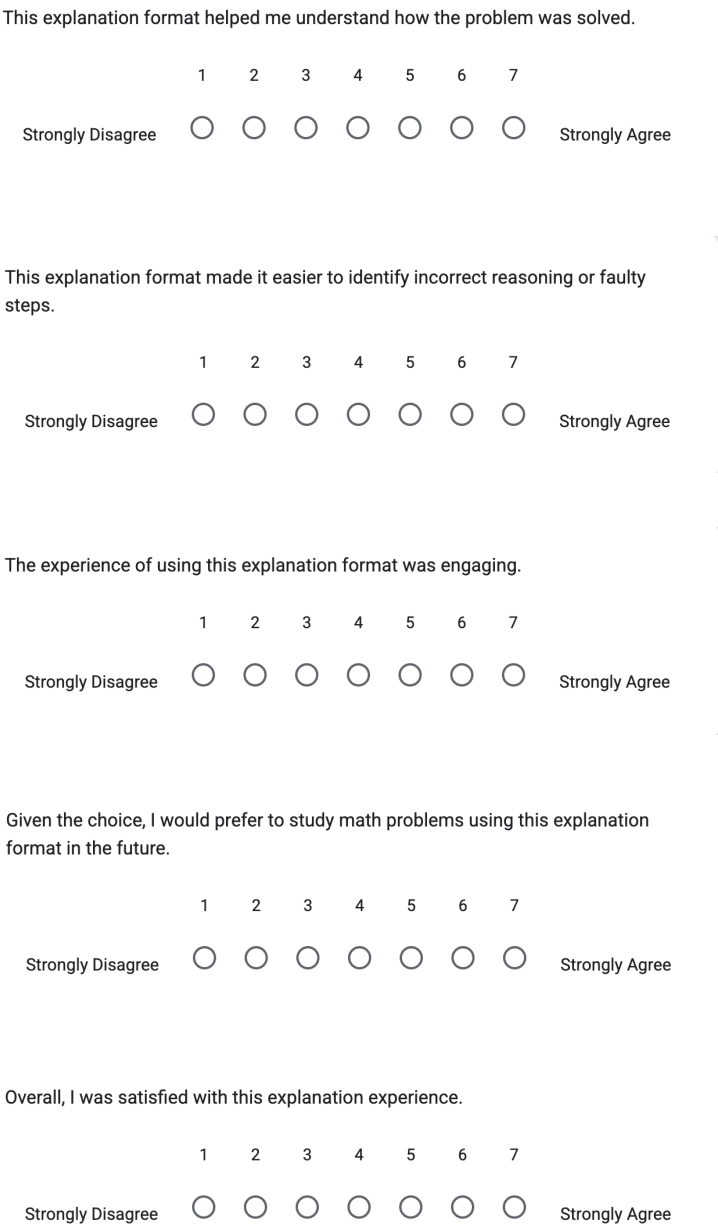

Figure 13: Post-Study Questionnaire

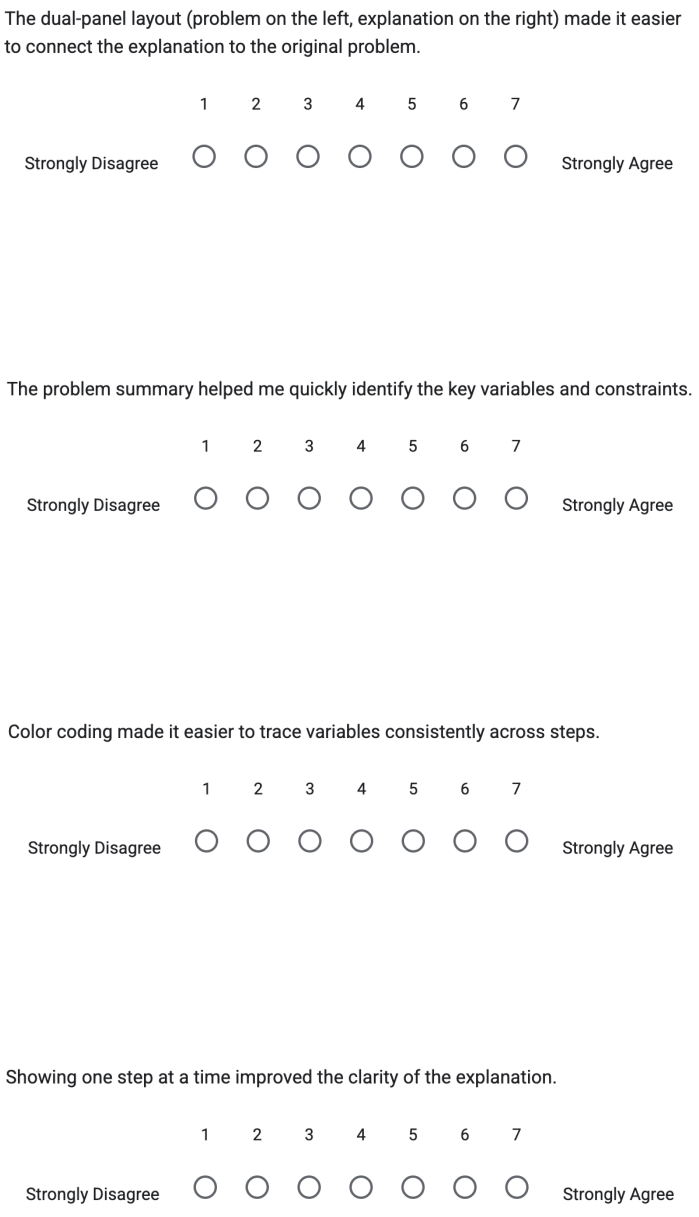

Figure 14: Design Features Questionnaire