UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA
Denkleiers • Leading Minds • Dikgopolo tša Dihlalefi

COS700 Research Proposal

Automatic Assessment of Cartesian Graphs in the South African Context

Mr. Charl Pieter Pretorius

Supervisors:
Prof. Linda Marshall
Mr. Pula Rammoko
Mr. Cobus Redelinghuys

Abstract

South African teachers operate under sustained workload pressures that constrain timely, individualised feedback, particularly in mathematics where formative assessment is essential. While automated grading has progressed for structured tasks (equations and short answers), the marking of hand-drawn Cartesian graphs remains largely manual. This paper investigates whether multimodal large language models (LLMs) can assess hand-drawn logarithmic graphs in alignment with the National Curriculum and Assessment Policy Statement (CAPS) for Pure Mathematics in South Africa (SA), returning memorandum-aligned marks and concise, rubric-based feedback. A semi-automatic workflow, Automated Cartesian-graph Assessment (ACA), is evaluated under zero-shot and rubric-guided prompting, with prompt evolution via Automatic Prompt Engineering (APE) and a capped human-in-the-loop refinement conducted out of band. The curated dataset comprises 26 sketches of $f(x) = \log_2 x$: 21 core student-like items and five outliers (e.g., missing axes, blank page, prompt-injection attempt). Each core item is redrawn in two groups with controlled variation in handwriting, scale and pencil type, while sharing labels. Two labels frame evaluation: an *expected mark* (professionally assigned ground truth) and an acceptance set of *potential marks* that a competent teacher could reasonably defend under CAPS. The study reports strict and relaxed accuracy, Cohen's κ (nominal and weighted) with bootstrap confidence intervals, bias, stability across decoding parameters and seeds, and an ablation isolating prompt versus decoding effects. The contribution is a reproducible, CAPS-aligned protocol and baseline feasibility evidence for selective assistive automation of hand-drawn graph marking in SA.

Keywords

LLM, automated assessment, mathematics, log, handwriting analysis.

1 Introduction

Teachers in South Africa face persistent workload and resource pressures that constrain timely, individualised feedback to learners. Empirical accounts describe curriculum density, administrative demands, and overcrowded classrooms as recurring challenges that reduce the time available for teaching and marking [1, 2]. In mathematics, feedback is not merely corrective: it signals personal attention and shapes subsequent engagement with tasks [3]. Within this context, scalable, curriculum-aligned assessment assistance has practical significance.

The Curriculum and Assessment Policy Statement (CAPS) frames assessment as a continuous, planned process that combines informal (assessment for learning) and formal (assessment of learning) activities, with evidence recorded to inform instruction [4]. For functions and graphs in the Further Education and Training phase, learners are expected to demonstrate understanding through diagrammatic work as well as symbol manipulation. Hand-drawn Cartesian graphs—especially logarithmic graphs—therefore represent a high-value target: they combine visual features (axes, scale, asymptotes) with symbolic reasoning (intercepts and qualitative shape), yet are labour-intensive to grade consistently at scale.

Prior systems have shown promising agreement for graphs drawn on a structured digital canvas [5]. Classroom artefacts in South Africa, however, are predominantly paper-based; scans and photographs introduce noise and handwriting variation that canvas-based approaches do not address. Related work on AI-assisted grading of handwritten responses highlights both potential and the need for reliability checks and human oversight [6].

This paper investigates whether modern multi-modal large language models can grade hand-drawn logarithmic graphs under CAPS-aligned criteria, and whether such grading can be made reliable and auditable through a controlled process. A curated dataset of 26 artefacts (21 student-like graphs and five outliers) redrawn into two groups for controlled variablility is used to probe typical and boundary errors. The approach, Automated Cartesian-graph Assessment (ACA), requires no model fine-tuning and follows a semi-automatic workflow: structured rubric-first prompting with Automatic Prompt Engineering (APE), a capped out-of-band human-in-the-loop refinement, and evaluation across decoding parameters. Agreement and bias are quantified against an expert-assigned ground truth and a declared acceptance set of potential marks, with stability examined across random seeds.

Paper roadmap: Section 3 situates assessment within the South African context and motivates the focus on logarithmic graphs. Section 4 contrasts manual, semi-automatic, and automatic assessment and positions the present method. Section 5 outlines the experimental process and clarifies expected versus potential marks and common logarithmic-graph errors. Section 6 details ACA: dataset curation, prompts and decoding parameters, APE and human-in-the-loop protocols, metrics (accuracy, Cohen's κ, bias, and confidence intervals), stability, and ablations. Section 7 reports results, and Section 9 concludes with limitations and immediate extensions.

2 The South African Educational Environment

This chapter motivates the need for assistive assessment by outlining teacher workload pressures, clarifying what assessment means under CAPS, and summarising why timely feedback matters. It closes with evidence that South Africa has the infrastructure to support curriculum-aligned digital tools at national scale.

2.1 Definition and Role of Assessment (CAPS)

According to the Curriculum and Assessment Policy Statement (CAPS) for Further Education and Training Phase as it stands in April 2025, assessment is defined as a continuous, structured process that involves collecting, interpreting, and evaluating evidence of student performance. This process encompasses four stages: generating assessment opportunities, evaluating outcomes, recording performance, and using these insights to inform both learning and teaching. CAPS mandates both informal (assessment for learning) and formal (assessment of learning) approaches, with regular feedback being a key component for supporting student development [4].

2.2 Teacher Workload and Feedback Gaps

Manual marking is labour-intensive and constrains meaningful, personalised feedback. In South Africa, chronic workload and capacity pressures reduce teachers' ability to deliver formative responses at pace, especially in mathematics [1, 2, 7]. As a cohort marker, the 2022 NSC briefing reported that approximately 775,630 of 1,337,290 learners graduated Grade 12 (about 58%), with Mathematics uptake at 37.2% [8]. These system-level pressures make scalable, rubric-aligned assistance attractive in high-volume marking contexts.

2.3 Why Feedback Matters

The educational value of feedback is well established. Rowe notes that students regard feedback not merely as correction but as a demonstration of personal attention [3]. Effective feedback fosters self-reflection (the student's ability to evaluate their own understanding), motivation (the drive to improve performance) and deeper engagement with content (active and sustained interaction with learning material) [3].

In practice, however, large classes, administrative load and time constraints in South African schools mean that feedback is often delayed, minimal or absent [1, 2, 7]. This weakens a critical learning mechanism precisely where formative support is most needed.

AI-powered feedback systems, when aligned to curriculum rubrics and used with appropriate oversight, may help restore this personal dimension by delivering instant, personalised comments on student work [9].

2.4 Readiness for Assistive Digital Tools

South Africa has already delivered curriculum-aligned Open Educational Resources at national scale. The Siyavula programme pioneered collaborative authoring and DBE-backed distribution of free print and digital textbooks in mathematics and science, reaching millions of learners and saving the state millions of rand in procurement; for specific print runs, reported savings are on the order of $\sim R107\,\text{million}$ per title at 500 k units [10]. Siyavula has since expanded to adaptive digital practice, mobile-friendly access and data-free delivery partnerships, illustrating the feasibility of curriculum-aligned, technology-supported learning at scale [10]. This ecosystem readiness strengthens the case for lightweight, rubric-aligned grading aids that complement teacher practice as seen in Section 4.

Chapter summary: Systemic workload constraints make timely feedback scarce, despite its recognised pedagogical value [1, 2, 3, 7]. CAPS frames assessment as continuous and feedback-oriented [4]. Prior national OER deployments (for example, Siyavula) demonstrate capacity for curriculum-aligned digital delivery [10], and at the same time, discrepancies between school-based assessment outcomes and external examinations in South Africa have been reported, highlighting variability in internal marking that motivates clearer rubrics and moderation [11], that could be accomplished through automation. The next chapter narrows to the CAPS mechanics most relevant to the focal case, namely the assessment of hand-drawn logarithmic graphs.

3 Assessment in the South African Context

This chapter situates the work within South Africa (SA), outlines what the Curriculum and Assessment Policy Statement (CAPS) expects when learners sketch and analyse functions, and explains why logarithmic graphs are an appropriate focal case for automated assessment. A concise description of the curated dataset anchors the later methodology. Broader motivation (workload, feedback) and national readiness appear in Section 2.

3.1 CAPS Overview (Pure Mathematics, Grades 10–12)

Under CAPS (Pure Mathematics), learners study families of functions and are expected to produce and analyse graphs using standard characteristics. This includes: domain and range, intercepts with the axes, asymptotes, qualitative shape and intervals of increase or decrease [4]. In the senior grades, treatment of inverses, including the inverse of the exponential function, makes logarithmic graphs both relevant and assessable within this framework. These prescriptions naturally support rubric-aligned checks that can be operationalised for automated assessment.

3.2 Why Logarithmic Graphs?

Logarithmic graphs have distinct curricular properties that are assessable from a static sketch. For the parent form $f(x) = \log_b x$ with $b > 1$, the curve is

increasing on the valid domain $x > 0$, has a vertical asymptote at $x = 0$, and intersects the x-axis at $(1, 0)$. These map cleanly to markable items (shape, intercept, asymptote) and to an error taxonomy, for example missing or incorrect asymptote, incorrect intercept placement and wrong qualitative shape. Because these checks are visually and symbolically grounded, they align with CAPS expectations for functions [4].

On the qualitative nature of shape. In practice, *shape* is a qualitative judgement: handwriting and axis scale can make borderline sketches admissible to different raters [5]. Rather than assuming binary certainty, the study handles this ambiguity explicitly in two ways that are used throughout the study: (i) the study assesses against a concise ground truth for the item obtained from a qualified source grading from a memorandum (Appendix 2) and (ii) the study defines *acceptance sets* of marks that are reasonably defensible under a CAPS-aligned interpretation (relaxed evaluation; Appendix 2). Agreement is then quantified in later sections using accuracy (strict and relaxed) and Cohen's κ with bootstrap confidence intervals, separating disagreement due to judgement variance from outright errors (Sections 5.8).

3.3 How Drawn Logarithmic Graphs Are Graded

Under CAPS, graph sketching is specified in terms of characteristics that should be evident on the diagram and interpretable by a marker. For logarithmic graphs, the relevant characteristics include: (i) **intercepts with the axes**, (ii) **asymptotes**, (iii) **domain and range**, and (iv) **qualitative shape and monotonicity** [4]. Logarithms are also treated via their relationship to the exponential function as inverses, which reinforces sketching via the same characteristics [4].

In school examinations these characteristics are commonly operationalised through concise, memorandum-aligned rubrics. For the item "*Sketch* $f(x) = \log_2 x$" (three marks), the scheme used in this study assigns one mark each for:

1. **Shape and monotonicity**: correct increasing logarithmic shape on the valid domain $x > 0$ with appropriate axis scale and legibility;
2. **x-intercept**: the curve passes through $(1, 0)$, with the coordinate explicitly indicated;
3. **Vertical asymptote**: the y-axis ($x = 0$) is shown and respected as an asymptote.

This rubric is consistent with CAPS characteristics at the level of intercepts, asymptotes and qualitative behaviour. The specific point $(1, 0)$ is a *memorandum rule for this item*, not an explicit CAPS requirement; the study therefore treats it as memo-derived while using CAPS as the curricular frame for assessable features [4].

Example marking of Fig. 1 (maximum 3). (1) Shape and monotonicity correctly shown $\rightarrow$ one mark; (2) x-intercept at $(1, 0)$ correctly indicated $\rightarrow$ one mark; (3) Vertical asymptote at $x = 0$ indicated and not crossed $\rightarrow$ one mark.

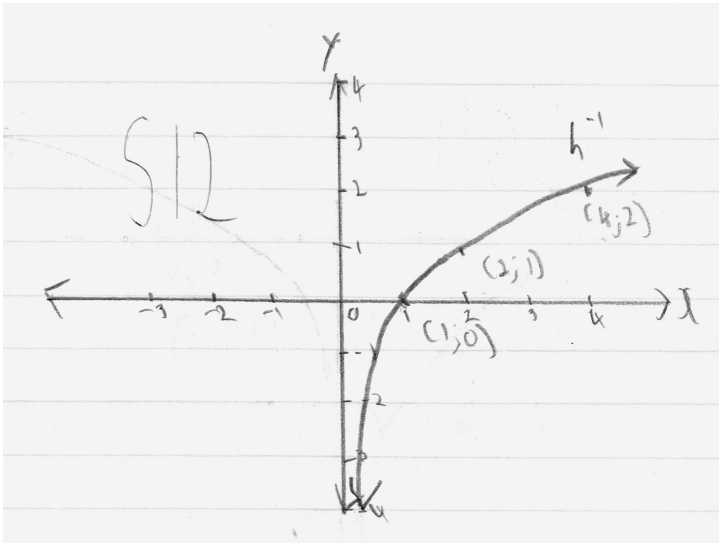


Figure 1: Example student sketch for "Sketch $f(x) = \log_2 x$" (three marks). The drawing shows an increasing logarithmic curve on $x > 0$, the x-intercept at $(1, 0)$, and the vertical asymptote at $x = 0$. (S12 simply denotes student-12's graph is pictured)

Assessing drawn logarithmic graphs therefore involves both **visual checks** (axes, scale legibility, asymptote placement) and **symbolic reasoning** (recognising the intercept and functional behaviour). These align with the CAPS characteristics that structure memorandum items [4].

Borderline sketches may attract different marks across raters; this motivates the relaxed *acceptance-set* evaluation used later alongside ground truth scoring.

3.4 Dataset Basis

This study uses a curated dataset of 26 hand-drawn logarithmic graphs inspired by authentic Grade 12 responses to a Gauteng provincial examination. The set comprises 21 core student-style sketches and 5 deliberate outlier cases: blank axes; no axes and no coordinates; correct shape with missing axes or coordinates; a handwritten prompt-injection attempt; and a blank answer. Each of the 21 core items was redrawn twice with controlled variation in pencil type, handwriting and scale (Group 1 and Group 2), while preserving identical ground truths across the paired items. Two evaluation labels are used throughout: *expected mark* (the professionally assigned ground truth, with one obvious clerical error corrected for consistency) and *potential marks* (any mark reasonably defensible under a CAPS-aligned interpretation of the sketch). These labels support strict and relaxed agreement analyses introduced in Section 5.8. See Appendix 2 for details regarding each dataset item and Appendix B for images of all items.

Section summary: CAPS provides explicit graphing characteristics and includes

inverse or exponential content that makes logarithms a natural, assessable target [4]. A small, curated dataset was constructed to reflect realistic variation and common errors, with dual labelling to support strict and relaxed evaluations. Because shape is inherently qualitative, the study pairs memorandum-based marking with *acceptance sets* and later quantify agreement using accuracy (strict and relaxed) and Cohen's κ with bootstrap confidence intervals.

4 From Manual to Automatic Assessment

This chapter traces the progression from teacher-only marking to automated support for hand-drawn Cartesian graphs, reviews prior work across deterministic and learning-based approaches, and positions the present method on the automation continuum.

4.1 Manual, Semi-automatic, and Automatic Assessment

Manual denotes teacher-only marking with no algorithmic decision support. *Semi-automatic* denotes workflows where a model proposes marks or feedback, while human choices configure and evolve the workflow, for example prompt design, candidate selection, and targeted refinements. *Automatic* denotes end-to-end algorithmic marking and feedback generation without human configuration changes between runs.

Suitability is judged on scalability and latency, consistency and reliability, transparency and auditability, and maintainability. LLM-based autograding studies report both potential and variability on these criteria, which motivates careful, auditable design and explicit oversight [12, 13, 14, 15, 16].

4.2 Deterministic Systems for Graph Grading

Rule-based systems have demonstrated feasibility for automatic grading of graphs created on a structured digital canvas. Martelly [5] developed a browser-based application that evaluates digitally drawn sketches against mathematical and visual criteria, including shape matching, monotonicity, domain coverage and, in some experiments, derivative comparisons. Agreement with teacher marks was promising under controlled, online-classroom conditions.

However, this line of work presupposes structured digital input and therefore does not address paper artefacts typical of school submissions. It bypasses scan noise, incomplete or ambiguous labels, handwriting variability and pencil artefacts. Feedback is returned via predefined comments rather than dynamic, curriculum-aligned explanations. These constraints limit scalability to real classrooms where pen and paper remain dominant.

Note on human variance: Even with structured, digital input, human agreement is not perfect. In Martelly's evaluation, teacher grades were only retained if at least five of six graders agreed ($\geq 80\%$); lower-agreement judgments were discarded to establish ground truth [5]. Independent South African studies also

document discrepancies between school-based and standardised assessments and highlight measurement/marking variability [11, 17]. This underscores the need to quantify agreement explicitly when comparing humans and automation.

4.3 LLMs for Assessment of Text

LLMs have been used to mark short textual answers and to provide short explanations of grading decisions, with varied reliability [9, 12]. Below, four lines of prior work are highlighted that inform the experiment discussed in CH 5.

1. **Zero-shot and course-scale deployments:** Recent reports explore grading without fine-tuning or exemplars (*zero-shot*, as defined in Section 5.2) while relying on prompt engineering (systematically improving the prompt without any model training, as defined in Section 5.2); for example, Yeung *et al.* : "leverage prompt engineering to evaluate student submissions without requiring additional training or fine-tuning" [13] and show a carefully engineered prompt plus a refined marking scheme working in practice without any model training. Semester-scale use in higher education is likewise documented by Poličar *et al.* [16] : in a 2024 Introduction to Bio-informatics course, they ran a randomized, blind, semester-long evaluation where six commercial and open-source LLMs graded most of 36 text-based questions once after the deadline; with well-designed prompts, LLMs achieved grading accuracy and feedback quality comparable to human TAs, and open-source models performed on par with commercial alternatives. Accordingly, decoding settings and prompts are fixed, and all runs are cataloged.

2. **Short-answer autograding:** Schneider *et al.* [12] report that generic LLMs can score short answers but may diverge from human judgements unless prompts and rubrics are carefully specified; they also emphasise recording model rationales alongside scores for audit. Speiser and Weng [9] demonstrate practical deployment via OpenAI APIs, using zero-shot prompts against reference answers and noting limits when responses are partially correct but unanticipated, underscoring the need for rubric anchoring and post-hoc review.

3. **LLM-as-judge with explainable ranking:** Deshpande *et al.* [18] propose an evaluator that ranks responses against rubric criteria and produces rubric-linked highlights/explanations rather than a bare score (also referred to as matching the error tags). This emphasis on explicit evidence motivates concise, rubric-anchored justifications rather than opaque marks.

4. **Reliability and governance:** Studies that run similar prompt enviroments (all settings remain consistent besides minor prompt edits) across multiple seeds and datasets indicate that strong models can still vary with prompt framing or context. Hackl *et al.* [14] evaluate GPT-4 as a rater of higher-education macroeconomics short-answer tasks, scoring content and style with a fixed prompt and sample solutions across repeated runs over 14 weeks, and report very high inter-rater reliability (intraclass correlation, ICC $\approx$ 0.94–0.99) together with a strong content–style correlation

($r \approx 0.87$); when style was deliberately degraded, content scores remained stable while style scores dropped, indicating criterion sensitivity and suggesting potential for LLMs to function as consistent graders in other domains when tasks are well specified and prompts are fixed. Governance analyses of educational automation call for shifting from general enthusiasm to concrete regulation and democratic oversight; accordingly, transparent and auditable pipelines are essential in assessment contexts [15].

Implications: Given this background, the study adopts rubric-first prompts, fixes decoding settings and seeds, catalogs outputs, and reports strict and relaxed agreement with confidence intervals. Specifically, accuracy is accompanied by Wilson 95% intervals and Cohen's κ (nominal and quadratic-weighted) is accompanied by bootstrap 95% intervals, as defined in Section 5.8; the full experimental protocol appears in Chapter 5.

4.4 multi-modal LLMs for Diagrams and Handwriting

Progress in multi-modal reasoning has introduced visual "chains of thought" and sketch-augmented problem solving. SKETCHPAD enables models to produce intermediate sketches during reasoning, improving performance on visual tasks [19]. In mathematics specifically, multi-modal benchmarks such as CMM–Math probe image-based mathematical reasoning in vision–language models [**?**]. These works signal a broader trajectory towards assessing visual mathematical artefacts, although they do not directly address curriculum-aligned marking of classroom scans.

Where handwriting is involved, two strands are relevant. First, models that ingest online handwriting signals demonstrate how stroke trajectories and curvature can inform recognition in vision–language systems [20]. Second, benchmarking of large language models for handwritten text recognition reports improving yet uneven performance across scripts and settings, reinforcing the need for auditable pipelines when handwriting enters the loop [21].

In a domain-adjacent, real-exam setting, Kortemeyer *et al.* [6] investigate AI-assisted grading of 252 handwritten thermodynamics exams using a pipeline of scanning, OCR (MathPix), and GPT-4V/GPT-4 for scoring and explanations. They report that the greatest challenge is converting handwriting to a machine-readable format, that grading hand-drawn graphics (e.g., process diagrams) is less reliable than symbolic derivations, and that LLM grading should be treated probabilistically with multiple runs to summarise confidence.

4.5 Handwriting and Diagram Recognition before LLMs

Before LLMs, classical OCR and deep models supported grading or assistance for handwritten educational artefacts. Examples include transfer learning for hand-drawn geometric shape classification [22]. Broader diagram-recognition work highlights challenges introduced by informal notation, visual noise and drawing variability [23]. These works indicate that visual understanding is feasible, yet authentic school artefacts present non-trivial robustness issues that

must be handled explicitly.

4.6 Prompt Design, APO (incl. APE), and HITL Discipline

Recent work argues for moving from ad hoc prompt tinkering to a *prompt science* that is documented, testable, and conducted with humans in the loop [24]. In parallel, automatic prompt optimisation (APO) techniques, including automatic prompt engineering (APE), search over prompts to elicit improved performance [25, 26]. To support auditability and reproducibility, the literature recommends disciplined human-in-the-loop practice in which prompt edits are documented and candidate prompts are compared on a development subset [24, 26].

4.7 Parameters, Decoding Choices and Output Stability

Sampling decisions influence both accuracy and stability. Naïve likelihood maximisation can produce degenerative text; truncation methods such as nucleus (top-p) sampling are therefore recommended [27].

For completeness, the decoding distributions are defined. Let $\mathbf{y} = (y_1, \ldots, y_V)$ be the logits (unnormalised scores) over a vocabulary of size V.

Temperature: Probabilities are obtained by a softmax over temperature $T > 0$:

$$q_i^{(T)} = \frac{\exp(y_i/T)}{\sum_{j=1}^{V} \exp(y_j/T)}.$$

Lower T concentrates mass on high-probability tokens; higher T flattens the distribution [28, 27].

Nucleus (top-p) sampling: Let the $q_i^{(T)}$ be sorted in descending order $q_{(1)}^{(T)} \geq \cdots \geq q_{(V)}^{(T)}$. Define the minimal m such that $\sum_{j=1}^{m} q_{(j)}^{(T)} \geq p$. With $\mathcal{P}_p = \{(1), \ldots, (m)\}$,

$$q_i^{(p)} = \frac{q_i^{(T)}\, \mathbb{I}\{i \in \mathcal{P}_p\}}{\sum_{j \in \mathcal{P}_p} q_j^{(T)}}.$$

Top-k sampling: Let $\mathcal{K}_k$ be the index set of the k largest $q_i^{(T)}$. Probabilities outside $\mathcal{K}_k$ are set to zero and the remainder renormalised:

$$q_i^{(k)} = \frac{q_i^{(T)}\, \mathbb{I}\{i \in \mathcal{K}_k\}}{\sum_{j \in \mathcal{K}_k} q_j^{(T)}}.$$

Top-k uses a fixed candidate set size; top-p adapts the set size to the local entropy of the distribution and is recommended to mitigate degeneration [27].

Empirical studies report mixed temperature effects: some find skill-specific and model-size-dependent behaviour, and propose automatic selectors that choose a

temperature per prompt [28], while others show little statistically significant impact from $T \in [0, 1]$ on problem-solving across models and prompts [29]. More recently, *Monte Carlo Temperature* (MCT) has been introduced for uncertainty quantification: rather than fixing T, temperatures are sampled from a distribution across multiple draws, which yields more robust uncertainty estimates without expensive temperature tuning and can match oracle, hand-tuned settings [30]. Subsequent sections treat decoding as a controlled factor and report variability under the chosen settings.

4.8 What the Author Believes to be Missing in the Literature

Despite these advances, curriculum-aligned assessment of *hand-drawn Cartesian graphs* remains unexplored. Prior deterministic graph graders concentrate on digital canvases [5]. multi-modal LLM papers focus on structured benchmarks or synthetic tasks rather than curriculum-aligned, memorandum-based grading of classroom answers, and they typically do not quantify defensible variance in human judgement found in real-world assessment. There is a gap for an auditable, rubric-grounded pipeline that targets paper-origin artefacts under curriculum and memorandum constraints that also considers allowable variance.

4.9 Positioning of the Present Method

The method operates in two phases on the automation continuum. *Development mode* is semi-automatic: prompts are designed and, where appropriate, refined using automatic prompt optimisation and automatic prompt engineering under a documented, human-in-the-loop discipline. Decoding settings are selected and then locked.

Deployment mode is fully automatic for the present micro-domain. Once the best-performing configuration is fixed, the system batch-grades new images without intervention, returning a mark and a short, rubric-aligned rationale per item with complete logs for audit and reproducibility.

Scope and justification. The study focuses on hand-drawn sketches of $f(x) = \log_2 x$ within Grade 12 Pure Mathematics under CAPS. The scope is narrow for three practical reasons. First, logarithmic graphs are representative of graph-based tasks that require both symbolic interpretation and visual checking: increasing shape on $x > 0$, the x-intercept at $(1, 0)$, and the vertical asymptote at $x = 0$. These properties map to a compact, memorandum-style rubric that is consistent with CAPS characteristics and everyday marking practice [4]. Second, authentic Grade 12 responses from a Gauteng preparatory examination, graded by a professional mathematics teacher, enabled the construction of a curated dataset whose items were re-drawn with controlled variation in handwriting, scale and pencil type while preserving identical ground truths. This provides realistic inputs with known labels for rigorous testing. Third, the tight rubric and small-n design allow structured evaluation of agreement, bias and stability, and make moderation transparent.

Free-form inputs remain challenging and sometimes poorly annotated. Two labels are therefore maintained throughout: an *expected* mark (professionally assigned ground truth) and *potential* marks that acknowledge reasonable classroom latitude under CAPS. Demonstrating that a prompt-engineered multimodal workflow can produce consistent marks together with concise, rubric-aligned feedback on this item serves as a proof of concept for selective assistive automation. CH 5 formalises the experimental process and introduces the ACA workflow evaluated in the remainder of the paper.

The anonymised, re-drawn sketches and item-level labels required to reproduce the analyses are provided in Appendices 2-B. Original student scripts are withheld to protect personal information and in line with institutional guidance and POPIA [31].

5 Process (Experimentation Overview)

This chapter outlines a high-level process for automated, curriculum-aligned grading of hand-drawn logarithmic graphs; named : ACA (Automated Cathesian-graph Assessment). The objective is to test whether a modern multi-modal large language model (MLLM), under a documented prompt and decoding recipe, can assign marks that align with CAPS-style memoranda and human judgement, and can do so reliably across natural variation in handwriting, scale, and pencil artefacts.

Proof-of-concept intent. The design is minimal and auditable so that performance on a single, tightly scoped logarithmic item serves as the foundation for broader portability. Strong agreement and stability in this micro-domain would indicate that the same prompt-and-decoding discipline can extend to other logarithmic tasks and, with rubric substitutions, likely to other function families. (More detail on this matter follows in CH 10)

5.1 Dataset, Pairing, Labels, and Privacy

The evaluation set comprises 26 item ground truths for $\log_2 x$, each redrawn twice to introduce controlled variation in handwriting, scale, and pencil artefacts, forming Group 1 and Group 2 (52 total sketches with paired indices across groups). Five deliberate outliers are included in each group to probe failure modes: blank axes; no axes and no coordinates; correct shape with missing axes or coordinates; a handwritten prompt-injection attempt; and a blank page.

Two complementary labels are used per sketch:

- **Expected mark** (ground truth): the professionally assigned memorandum mark.
- **Potential marks** (acceptance set): all marks reasonably defensible under a CAPS-aligned interpretation of the sketch.

Strict evaluation tests exact agreement with the expected mark. **Relaxed** evaluation tests membership in the acceptance set. Original student scripts are

not reproduced for privacy reasons [31]; per-item labels and redrawn surrogates appear in Appendix 2 and Appendix B.

5.2 Prompt Evolution Sequence

The sequence progresses from least to most constrained supervision to separate gains due to instruction content from those due to sampling noise:

1. **Zero-shot:** Minimal instruction to establish a baseline commonly used in autograding [12, 13].
2. **Few-shot:** Three annotated exemplars (correct, partial, incorrect) to guide in-context behaviour and improve rubric adherence [13, 16].
3. **Structured reasoning:** A checklist that makes the rubric explicit (axes, scaling, curve shape and monotonicity, x-intercept at $(1, 0)$, vertical asymptote at x=0), eliciting stepwise evaluation in visual–symbolic marking [6, 19].
4. **Structured reasoning without examplars*:** This stage follows the exact steps used for structured reasoning but breaks the iterative nature of the evolution sequence, this stage was added during execution after it was discovered that the structured reasoning stage performed worse than the few-shot stage on average.
5. **APO/APE with HITL.** Automatic prompt optimisation, including automatic prompt engineering, generates K candidate prompts. Candidates are compared on the full 26-item set (including the five outliers). The top configuration advances to a capped human-in-the-loop refinement with a documented change log [24, 25, 26].

All stages are evaluated separately on the two dataset groups without carry-over of model state between stages.

Definition: Automatic Prompt Engineering (APE) is an automated procedure that, given a seed prompt and a small set of task examples, *(i)* generates candidate prompt variants using an LLM (each model will use itself), *(ii)* scores each candidate on held-out examples using a task metric, and *(iii)* selects the top candidate for the next round or for deployment. APE instantiates the broader class of Automatic Prompt Optimisation (APO) methods that search the discrete space of natural-language instructions without model fine-tuning [26].

5.3 Parameters Overview

Decoding choices are limited to temperature and nucleus probability (*top-p*). Fixed *top-k* is not pursued due to brittleness across contexts [27]. Formal definitions and background appear in Chapter 4.7.

5.4 Searching Parameter Values

A full factorial over temperature and *top-p* across all prompt stages and models is infeasible. Prior work indicates that temperature primarily governs variance

and reasoning diversity, while nucleus sampling constrains decoding to a minimal high-mass set [27, 28, 29]. Sequential and Monte-Carlo temperature strategies provide efficient alternatives to exhaustive grids [30]. Accordingly, a compact sequential search over a set range adopted.

5.5 Chosen Temperature and Top-p Values

For ChatGPT models (API range $[0, 1]$), temperatures are set to $T \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, targeting the mid-range where reasoning performance and stability are jointly optimised in analytical tasks [28, 29]. Very low temperatures ($T \leq 0.2$) are excluded due to brittle, overly deterministic outputs that struggle to understand nuance [29], while higher temperatures ($T \geq 1.0$) increase diversity at the expense of reproducibility [28]. The commonly used vendor midpoint (T=0.7) is included as a practical prior, while recognising that optimal values are task- and model-dependent [28].

For Gemini models (API range $[0, 2]$), the corresponding set is $T \in \{0.6, 0.8, 1.0, 1.2, 1.4\}$.

Top-p is tested at $\{0.85, 0.90, 0.95\}$, following evidence that $p \approx 0.9$–0.95 maintains coherence [27] while permitting further exploration, with p=0.85 providing a more constrained baseline.

5.6 Execution Order and Early-Stop Policy

Within each prompt stage and for each model–group (groups are defined in 3.4 pair for a single set seed:

1. **Phase A: temperature sweep.** Run each T once with *top-p*=0.95; select by highest accuracy, with ties broken by higher Cohen's κ.
2. **Phase B: *top-p* sweep.** Fix T at the Phase-A winner; run the remaining *top-p* values and select using the same criteria.
3. **Interaction check:** Re-run the passing *top-p* at the Phase-A runner-up T to detect strong interaction if the passing *top-p* is not 0.95

Final candidates are then re-run with five different seeds to assess stability [30]. Figure 2 summarises the stage-wise flow. After each stage, the selected configuration is compared to the success criteria in Section 5.9. Meeting all thresholds permits early termination for that model–group to conserve budget; otherwise, the next prompt stage is entered.

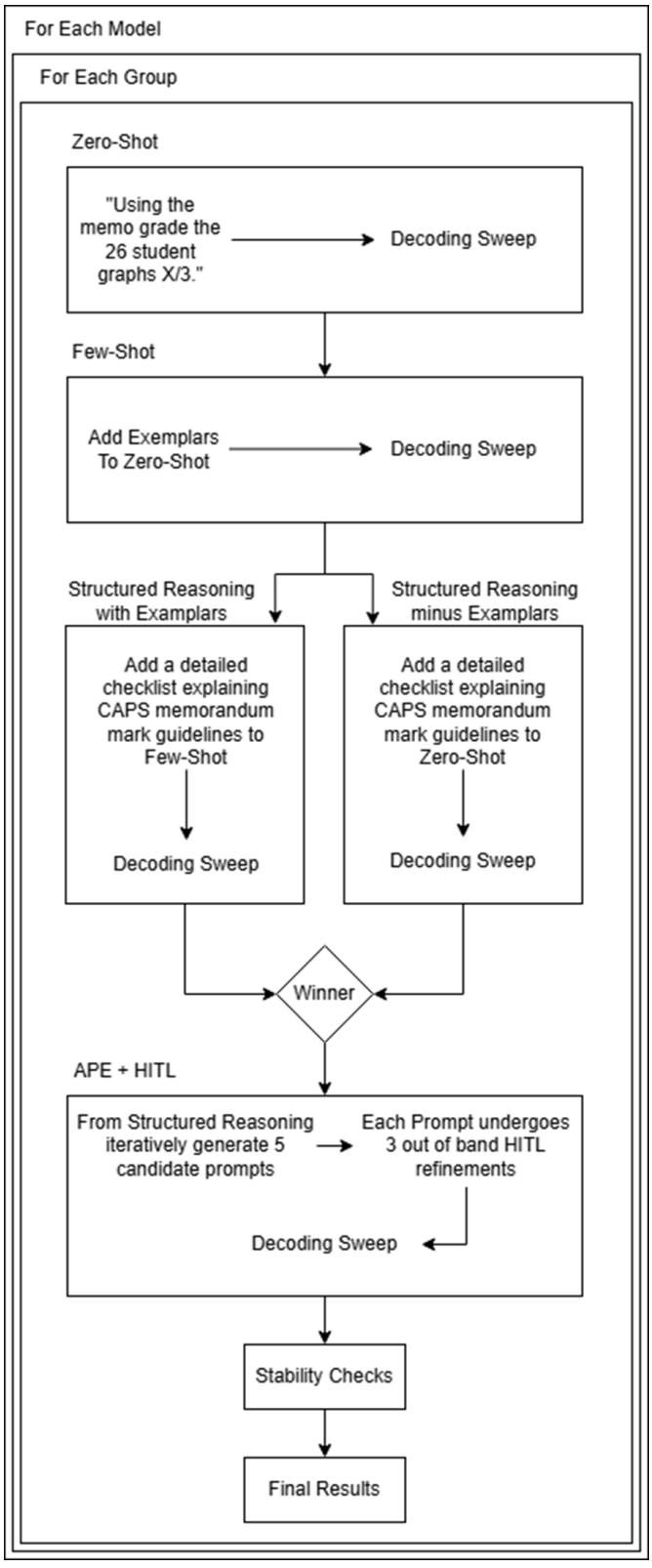


Figure 2: Execution flow per model and per group. As explained in Section 5.2 and 5.6

5.7 Run Structure and Logging

For each configuration, the model grades each dataset group. Logged fields include `item_id`, group, `model_mark`, `expected_mark`, `prompt_id`, `model_id`, temperature, *top-p*, and seed. Batch summaries report correct matches, confusion patterns, decoding settings, and all metrics as seen in related research [12, 13, 6].

5.8 Evaluation Metrics

Per configuration and per prompt stage, the following are reported; informed by prior findings:

- **Accuracy.** Exact match rate in strict and relaxed modes; Wilson score 95% confidence intervals. [5, 9]
- **Stability.** Range of accuracy across seeds for the same configuration. [6]
- **Cohen's κ.** Agreement beyond chance, reported as point estimates (nominal and quadratic-weighted); and with percentile bootstrap 95% confidence via item-level resampling intervals [24]. Nominal is reported for completeness but is not applicable as a success criteria considering the small dataset.
- **Bias.** Mean signed error (assigned minus expected) with a 95% interval. [9, 16]

5.9 Success Criteria

A configuration is considered successful if it satisfies:

1. **Accuracy:** Strict accuracy $\geq 75\%$, with relaxed accuracy $\geq 95\%$.
2. **Reliability:** Cohen's $\kappa > 0.75$ (quadratic-weighted), with CIs reported.
3. **Bias:** Mean signed error (model $-$ expected) not significantly different from 0 (95% CI includes 0).
4. **Stability:** Across-seed failure to meet other success criteria $\leq 5\%$ over repeated runs.

Rationale: Recent LLM auto-grading studies report accuracies in the ~ 0.8 band and note that human–human agreement can be comparable; [9] thus, the study requires at least 75% strict accuracy and expects a clearly higher relaxed score ($\geq 95\%$). Reporting accuracy with confidence intervals and analysing average grading differences is standard practice in course-scale autograding [5, 9], which is followed here; κ is additionally reported as an agreement-beyond-chance statistic as seen previously done by Shah [24].

Because LLM outputs are stochastic, repeated runs and dispersion summaries are recommended [6]; therefore, across-seed variation is capped at a small engineering target (5%) to ensure that the result found was not due to seed overfitting.

Finally, real-world assessors have been found to assign marks differently [5, 11], motivating the strict/relaxed distinction and bias checks.

5.10 Feasibility and Scope Controls

The budget is bounded by

$$(\textbf{2 models} \times \textbf{4 prompt stages} \times (5T + 3p + \textbf{5 stability runs}) \times \textbf{1 initial seed} \leq 104 \textbf{ configurations executed}.$$

after which only top candidates are subjected to multi-seed stability testing. No model fine-tuning is performed; observed changes are attributable to prompt design and decoding settings [12, 13]. All input images remain fixed.

Section summary ACA is framed as assistive, auditable automation that proposes CAPS-aligned marks and short feedback on hand-drawn logarithmic graphs. The procedure specifies dataset groups, staged prompt evolution, compact decoding searches, logging, and evaluation. Success in this micro-domain provides initial evidence for extending the same disciplined prompting and decoding protocol to broader drawn-graph assessment.

6 Automated Cartesian-graph Assessment (ACA) in Detail

This section specifies the ACA workflow at the implementation level. The high-level overview with justification can be found in Section 5. The description covers dataset layout, prompting configurations, decoding search, inference and retry logic, output parsing, statistical reporting, stability protocol, and archived artefacts.

6.1 Experimental Setup

Environment: Python 3.13.7 with `google-genai`, `numpy`, `pandas`, and `scikit-learn`. API credentials are supplied via environment variables and cannot be publicly shared.

Determinism and seeds: A generation seed is provided to the vendor API. Bootstrap procedures use fixed pseudo-random seeds. Residual stochasticity is expected for $T > 0$.

6.2 Dataset and Labels (Operational)

File layout: Two folders, `Group1` and `Group2`, contain 26 JPEG sketches each (`S{n}_{tag}.jpg`, $n \in \{1, \ldots, 26\}$). The same n across groups refers to the same underlying ground truth. The memorandum is provided as `Q5.2.4Memo.jpg` and is the official Gauteng Department of Education memorandum for the question: "*Sketch* $f(x) = \log_2 x$ *; indicate all intercepts*".

Ground-truth JSON: For each item i, a JSON entry stores the *expected mark* $y_i \in \{0, 1, 2, 3\}$ and the *acceptance set* $\mathcal{A}_i \subseteq \{0, 1, 2, 3\}$. Items $n \in$

$\{22, 23, 24, 25, 26\}$ constitute the outlier set. All stages, including prompt selection and parameter sweeps, grade all 26 items per group. Each unique item is redrawn in both groups, hence labels are identical across redraws.

6.3 Input Packaging and Prompting

Batch composition: Each run forms one multi-modal request containing: (i) a short instruction string, (ii) the 26 student images in item-ID order, and (iii) the memo image.

Prompting configurations: Four prompt stages are evaluated sequentially on the same fixed images; the models have no memory.

- **P0 (zero-shot):** Minimal instruction to establish a baseline [12, 13].
- **P1 (few-shot):** Three labelled exemplars (correct, partial, incorrect) are added to P0 to elicit rubric-aligned behaviour [13, 16].
- **P2 (structured reasoning):** A checklist guiding checks of axes/scale legibility, increasing shape on $x > 0$, the x-intercept at $(1, 0)$, and the vertical asymptote at $x = 0$, followed by a one-sentence rubric-aligned rationale are added to P1 [6, 19].
- **P3 (APE + HITL):** Automatic prompt engineering (APE) generates K candidates from the best P2 prompt [25, 26]. The top candidate advances to a capped human-in-the-loop refinement with a documented change log [24].

APE protocol used in this study; Starting from the best P2 prompt:

1. **Candidate generation:** the LLM in question proposes K edited or paraphrased prompts subject to rubric and format constraints.
2. **Scoring:** each candidate is evaluated on the development set using *Accuracy* as the primary metric; ties are broken by higher quadratic-weighted κ.
3. **Selection:** the top-1 candidate advances. A single **HITL** pass applies up to five targeted edits with a documented change log, guided by rubric-coded failure modes.

This small-n setup uses stratified leave-one-out scoring to limit optimism bias while avoiding model fine-tuning [26].

6.4 Decoding Settings and Search Protocol

Parameters varied: Temperature T and nucleus probability p (top-p).

Search grid: ChatGPT uses $T \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$; Gemini (API range $[0, 2]$) uses $T \in \{0.6, 0.8, 1.0, 1.2, 1.4\}$. Nucleus values are $p \in \{0.85, 0.90, 0.95\}$. See Section 4.7 for more details.

Sequential sweep (per prompt stage, per model).

1. Phase A: sweep T at p=0.95; select by Accuracy, then κ.
2. Phase B: sweep p at the selected T.

Final candidates are re-run with multiple seeds for stability checks.

6.5 Inference, Retry, and Run Orchestration

Execution unit: A configuration is the 7-tuple (group, model, prompt_id, T, p, seed, images).

API call and retry: The configuration is executed as one batch request. Overload or HTTP 503 responses trigger exponential back-off with bounded retries.

Sweep driver: Configurations are enumerated externally by updating the attempt label, T, p, and the active group; this is equivalent to a for-loop over the grid and groups and helps to avoid API 500 errors.

6.6 Output Parsing and Validation

Expected format: The model emits 26 final marks of the form "$k/3$" with $k \in \{0, 1, 2, 3\}$, in item order.

Extraction: A regular expression `(\d+)\s*/\s*3` is applied to the returned Markdown. The first 26 matches are retained. A parsing log is saved.

Basic validation: Counts are checked. If fewer than 26 marks are extracted, the run is flagged for a retry or exclusion according to the pre-declared policy.

6.7 Evaluation Metrics and Statistical Reporting

Let N denote the number of graded items, $\hat{y}_i$ the model mark, y_i the expected mark, and $\mathcal{A}_i$ the acceptance set.

6.7.1 Accuracy (strict and relaxed)

$$\mathrm{Acc}_{\mathrm{strict}} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = y_i), \qquad \mathrm{Acc}_{\mathrm{relaxed}} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\hat{y}_i \in \mathcal{A}_i).$$

Confidence intervals. Accuracy is reported with 95% Wilson score intervals (binomial proportion).

6.7.2 Bias (mean signed error)

$$\mathrm{Bias} = \bar{e} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i),$$

with a 95% interval from a non-parametric bootstrap over items.

6.7.3 Cohen's κ (nominal and quadratic-weighted)

Given a $K \times K$ confusion matrix $\{n_{ij}\}$ with $N = \sum_{ij} n_{ij}$,

$$p_o = \frac{1}{N} \sum_{i=1}^{K} n_{ii}, \quad p_e = \sum_{k=1}^{K} \left(\frac{n_{k\cdot}}{N} \right) \left(\frac{n_{\cdot k}}{N} \right), \quad \kappa = \frac{p_o - p_e}{1 - p_e}.$$

Quadratic-weighted κ_w uses weights $w_{ij} = 1 - \left(\frac{i-j}{K-1} \right)^2$ and the corresponding weighted p_o and p_e. For both strict and relaxed targets, percentile bootstrap 95% confidence intervals are obtained by resampling items 5000 times.

Agreement versus variance: κ quantifies agreement beyond chance for a fixed configuration. Between-run variance is summarised by the Accuracy range across repeated seeds for final candidates (see §6.8).

6.8 Stability Protocol

Stability is defined as the range of *strict* Accuracy across repeated seeds for the same configuration. The acceptance threshold is $\leq 5\%$. The generation seed is varied across five distinct values for final candidates.

6.9 Logging and Artefacts

Per-run directory: Each configuration writes to a unique folder named by attempt, model, group, T, and p.

Saved files: Raw model output (Markdown); `all_student_marks.csv` with $(\hat{y}_i, y_i)$; strict and relaxed confusion matrices (CSV); `per_student_appendix.csv` with acceptance-set flags; bootstrap samples for κ (CSV); a summary text with all metrics and the prompt used.

Reproducibility: All images are immutable. Prompts and decoding settings uniquely identify a run. Do note that vendors may deprecate models in the future.

7 ACA Results Overview

7.1 Overall performance and winners

Configurations that satisfied all predeclared criteria are listed in Table 1. The strongest configuration achieved strict 96.15% and relaxed 100.00%, with quadratic-weighted Cohen's κ=0.98. The remaining winners lie in strict 88.46%-96.15% and relaxed 96.15%-100.00%, with κ_q in 0.89-0.98.

Passing configurations (more details in Table 1).

- **Gemini-2.5-Pro Group 1**: APE+HITL Iteration 1. (Gemini_APE_Iteration1)
- **Gemini-2.5-Pro Group 2**: APE+HITL Iteration 3. (Gemini_APE_Iteration3)

- **ChatGPT-4.1 Group 1**: Out of ACA; Used Gemini APE+HITL Iteration 1's prompt. (GPT_GeminiPrompt_Iteration1)
- **ChatGPT-4.1 Group 2**: Out of ACA; Used Gemini APE+HITL Iteration 2's prompt. (GPT_GeminiPrompt_Iteration2)

Table 1: Final passing configurations with accuracy (strict, relaxed; 95% Wilson CIs).

Model	**Group**	**Temp**	**Top-p**	**Strict [CI]**	**Relaxed [CI]**
GPT-4.1	Group 1	0.90	0.95	0.96 [0.81-0.99]	1.00 [0.87-1.00]
GPT-4.1	Group 2	0.90	0.95	0.85 [0.66-0.94]	1.00 [0.87-1.00]
Gemini	Group 1	1.00	0.95	0.92 [0.76-0.98]	0.96 [0.81-0.99]
Gemini	Group 2	0.60	0.90	0.88 [0.71-0.96]	1.00 [0.87-1.00]

Table 1: Final passing configurations (continued): Cohen's κ (nominal and quadratic; 95% bootstrap CIs) and bias (mean signed error with 95% CI). All five stability-check seeds were passed.

Model	**Group**	**κ_{nom} [CI]**	**κ_q [CI]**	**Bias [CI]**
GPT-4.1	Group 1	0.94 [0.81-1.00]	0.96 [0.85-1.00]	-0.08 [-0.23, 0.00]
GPT-4.1	Group 2	0.77 [0.55-0.94]	0.92 [0.80-0.99]	-0.04 [-0.27, 0.15]
Gemini	Group 1	0.89 [0.73-1.00]	0.98 [0.94-1.00]	0.00 [-0.12, 0.12]
Gemini	Group 2	0.82 [0.62-1.00]	0.97 [0.91-1.00]	-0.04 [-0.15, 0.08]

Note: For all winners, κ CIs have lower bounds > 0, indicating agreement beyond chance.

7.2 Stage-wise behaviour (top-p=0.95)

Fig. 3 and Fig. 4 summarise pooled stage accuracies (Group 1+2). On Gemini, strict accuracies move from zero-shot 81.2% through few-shot 70.8%, structured 68.5%, structured-minus-exemplar 71.9%, to APE+HITL 82.1%. On ChatGPT-4.1 the corresponding ladder is 56.2%, 37.3%, 50.0%, 72.7%, and 76.2%. Relaxed follows the same ordering (right-hand column of each heatbar).

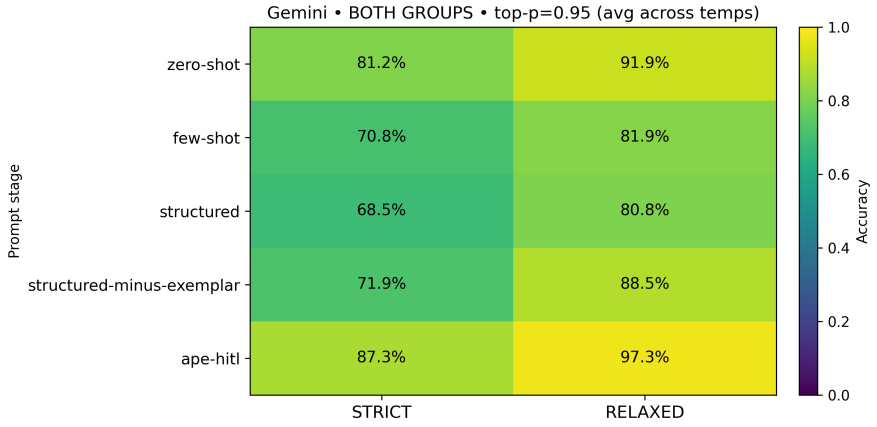


Figure 3: Pooled (Group 1+2) stage accuracies at top-p=0.95 for **Gemini-2.5-Pro**.

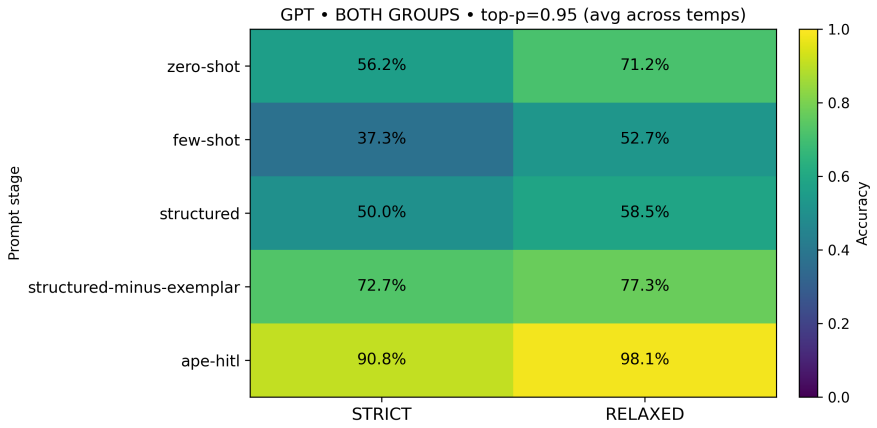


Figure 4: Pooled (Group 1+2) stage accuracies at top-p=0.95 for **ChatGPT-4.1**.

Per-group temperature grids (strict). To preserve decoding sensitivity without pooling across groups, strict grids are shown per model and group.

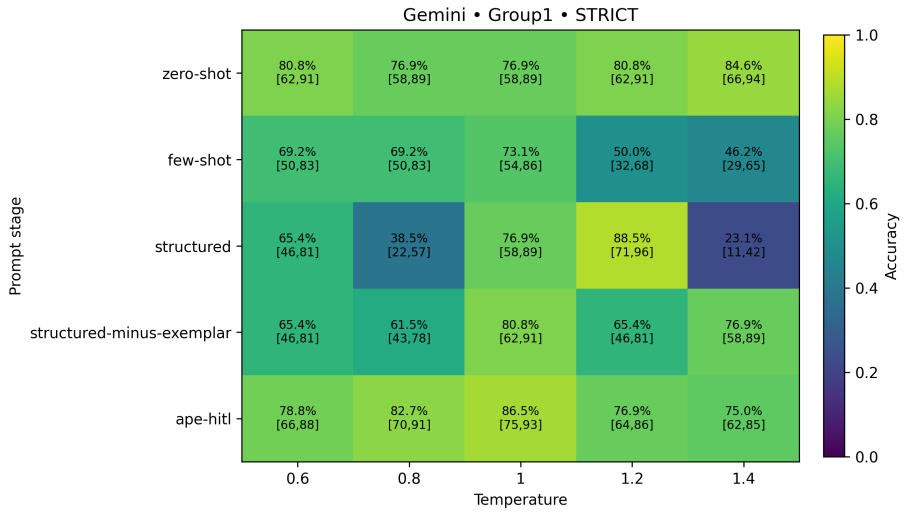


Figure 5: Temperature grid (**strict**) for **Gemini-2.5-Pro**, Group 1.

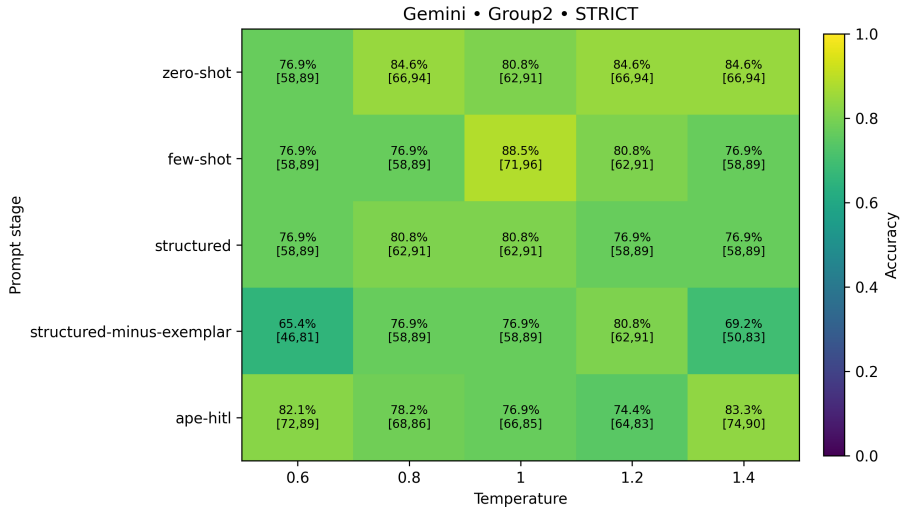


Figure 6: Temperature grid (**strict**) for **Gemini-2.5-Pro**, Group 2.

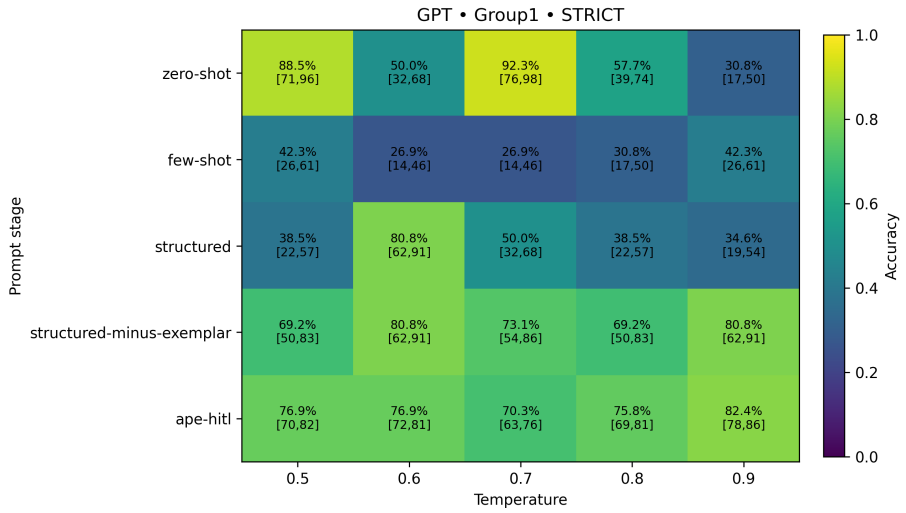


Figure 7: Temperature grid (**strict**) for **ChatGPT-4.1**, Group 1.

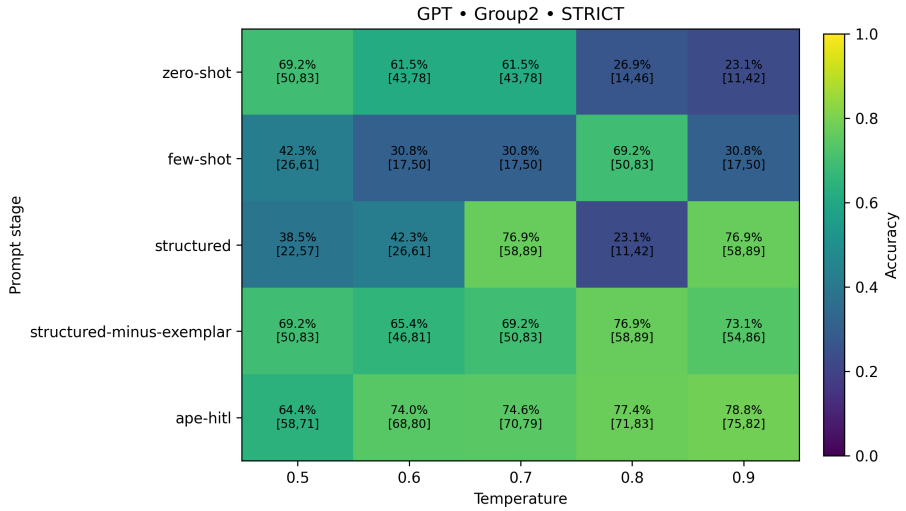


Figure 8: Temperature grid (**strict**) for **ChatGPT-4.1**, Group 2.

Per-group temperature grids (relaxed). Relaxed grids mirror the strict patterns and are provided for completeness.

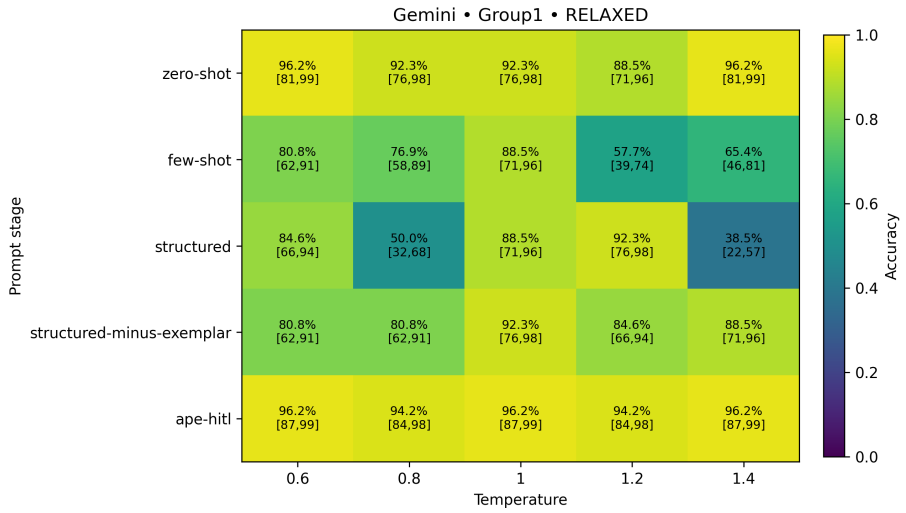


Figure 9: Temperature grid (**relaxed**) for **Gemini-2.5-Pro**, Group 1.

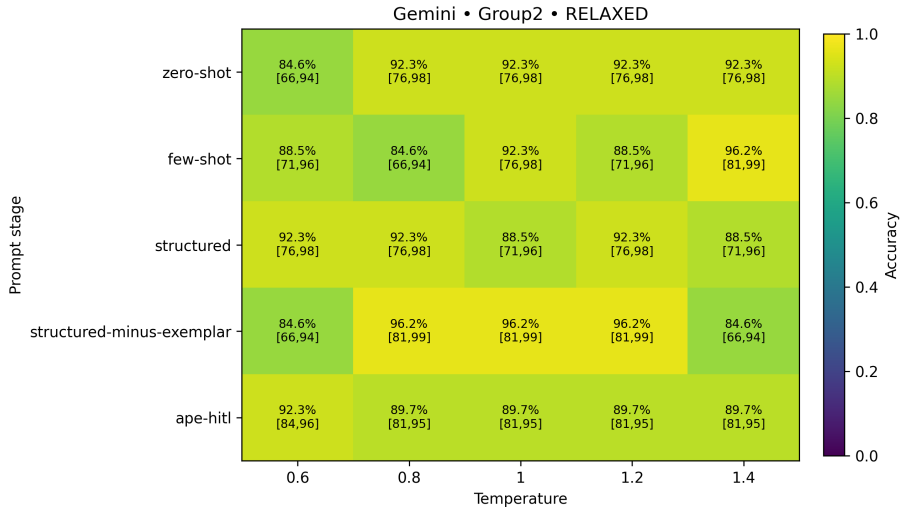


Figure 10: Temperature grid (**relaxed**) for **Gemini-2.5-Pro**, Group 2.

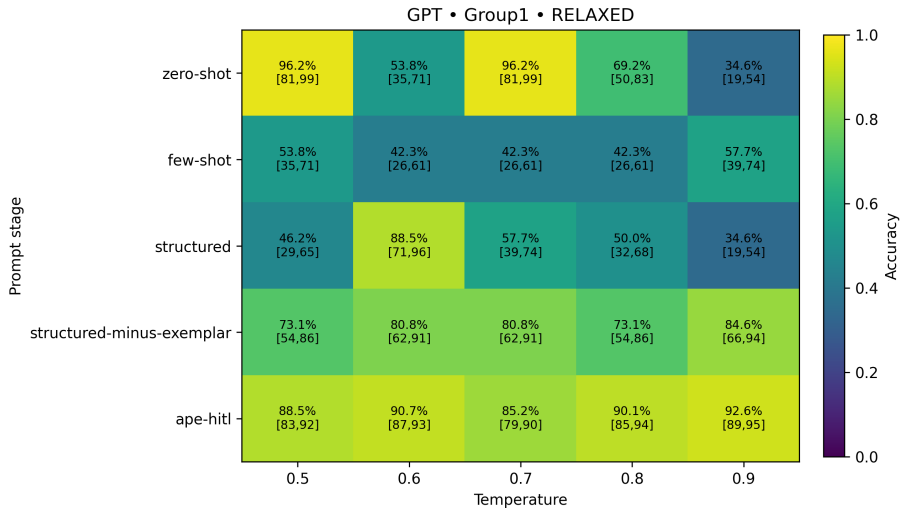


Figure 11: Temperature grid (**relaxed**) for **ChatGPT-4.1**, Group 1.

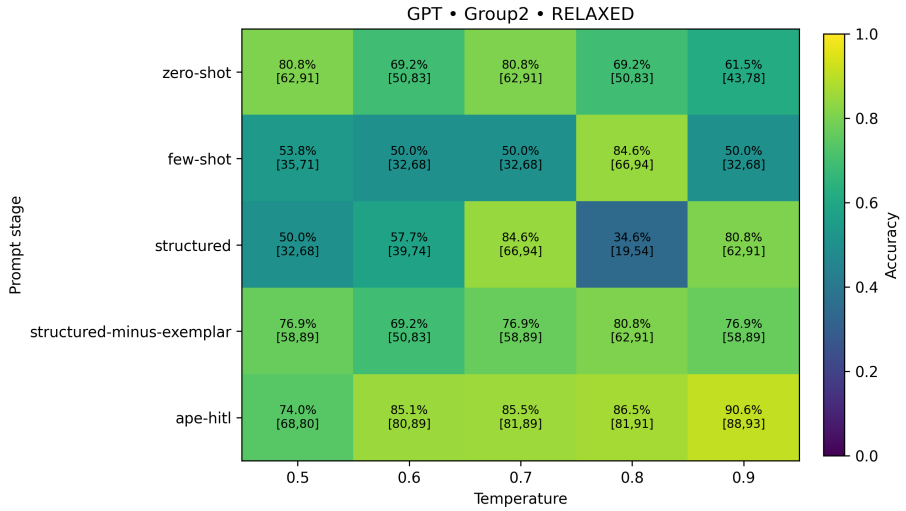


Figure 12: Temperature grid (**relaxed**) for **ChatGPT-4.1**, Group 2.

7.3 Stability across seeds

Each winner was re-run on six seeds (selection seed plus five checks); all seeds met the thresholds. The maximum across-seed ranges observed among winners were 15.39% (strict) and 3.85% (relaxed). Per-student outcomes across the original winner and its five stability checks are visualised in Fig. 13.

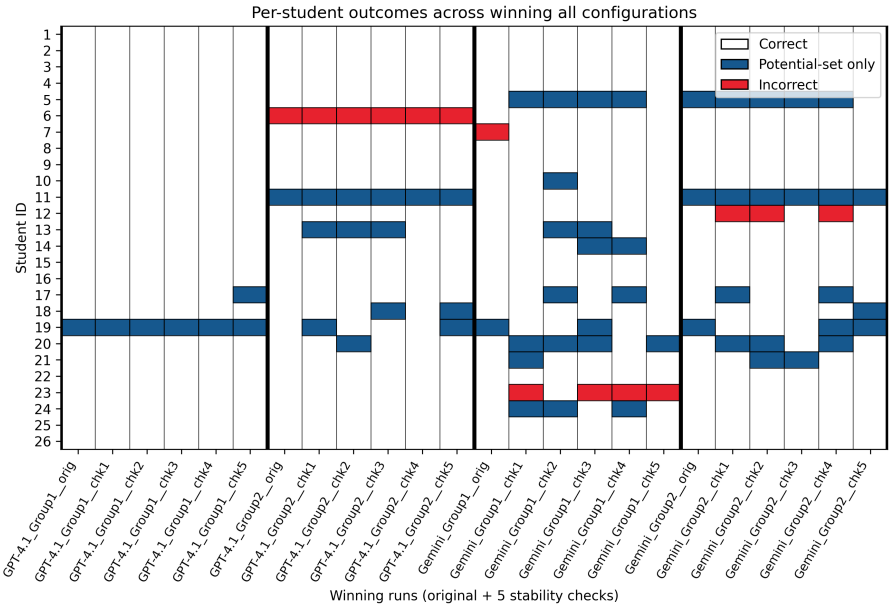


Figure 13: Per-student outcomes for each passing configuration and its five stability checks. White = strictly correct mark assigned; blue = potential-set match for the mark only; red = incorrect mark assigned.

Note: Items S22-S26 are the designed outliers. Refer to Appendix B for all dataset items' visual represention.

7.4 Examples of model feedback

The winners produced rubric-aligned feedback alongside marks. Short excerpts illustrate typical behaviour on correct, borderline, and outlier items from the passing configurations.

ChatGPT-4.1, Group 1:

- **S7, 3/3:** "Strictly increasing with concave-down curvature; (1,0) labelled and crossed; approaches the y-axis without crossing."
- **S6, 0/3:** "Two branches, not a single function; no clear (1,0); does not show the correct vertical asymptote."

ChatGPT-4.1, Group 2:

- **S1, 2/3:** "Predominantly linear between key points; (1,0) correct; vertical asymptote behaviour correct."
- **S8, 0/3:** "Mostly linear, no dot at (1,0), and does not approach the y-axis as an asymptote."

Gemini-2.5-Pro, Group 1:

- **S18, 2/3:** "Slight concave-down shape accepted per leniency rule; misses (1,0); approaches the y-axis correctly."
- **S5, 2/3:** "Drawn almost as a straight line; (1,0) correct; asymptote behaviour correct."

Gemini-2.5-Pro, Group 2:

- **S2, 3/3:** "Continuous, gentle concave-down curve; (1,0) labelled; approaches the y-axis from the right without crossing."
- **S23, 0/3:** "Rule 0 fail: no axes; x-intercept and asymptote cannot be shown."

7.5 Cross-model prompt transfer

Google's Gemini-2.5-Pro achieved its passing configurations through the ACA with final configurations produced by its own APE+HITL stage. OpenAI's ChatGPT-4.1 passed when given the Gemini-derived APE+HITL prompt and exceeded Gemini on strict accuracy by 3.84% on the final evaluation (see Table 1), however, it nearly produced passing configurations with its Group one only failing a single run during the stability check and its Group two failing two.

7.6 Section summary

Within a classroom-sized dataset of n=26 hand-drawn logarithmic graphs split into two redraw groups with shared labels, all four final winners satisfied the predeclared success criteria on accuracy, agreement beyond chance, stability, and bias. These results establish a proof of concept that multi-modal LLMs should be able to assess CAPS-aligned hand-drawn logarithmic graphs and return rubric-aligned feedback.

8 Discussion

8.1 Summary of principal findings

This study tested whether ACA-guided prompting enables reliable automatic grading of CAPS-aligned hand-drawn $\log_2(x)$ graphs. The final configurations of ChatGPT-4.1 and Gemini-2.5-Pro met all predeclared criteria on accuracy, agreement beyond chance, bias, and seed stability (Table 1). Zero-shot often approached the thresholds but was unstable. Few-shot, structured, and structured-minus-exemplar underperformed zero-shot. Only APE with a small, capped number of out-of-band HITL edits yielded stable, passing performance (Fig. 3, Fig. 4, Fig. 13). Patterns were consistent across both redraw groups.

8.2 Stage-wise behaviour and what it implies

Zero-shot: Several settings neared the strict and relaxed targets but showed higher variance across seeds, indicating sensitivity to sampling and wording rather than robust competence. Zero-shot is useful as a probe, not as an operational setting.

Few-shot and structured: These stages did not improve over zero-shot on this micro-domain and often performed worse (Fig. 3, Fig. 4). Small, heterogeneous exemplars likely anchored models to idiosyncratic handwriting or scaling artefacts rather than the rubric.

Structured-minus-exemplar: Removing exemplars reduced undesirable anchoring but did not surpass zero-shot on average. Structural cues alone, without APE refinement, were insufficient under handwriting variation.

APE + HITL: APE with limited HITL concentrated prompts on rubric primitives (shape, x-intercept, asymptote), clarified acceptance-set logic, and discouraged over-reading of ambiguous strokes. Gains exceeded those from changing temperature or top-p (Fig. 3, Fig. 4). All winners passed the seed-stability check (Fig. 13).

8.3 Feedback: why the outputs are acceptable in context

Passing configurations produced concise, rubric-aligned justifications alongside marks (Section 7.4). Although not a pedagogical evaluation, the feedback is acceptable as a technical artefact because it references rubric components, avoids hallucinated coordinates in the inspected cases, and flags the same failure modes seen in strict-wrong items.

8.4 Cross-model prompt transfer and its implications

Gemini satisfied ACA with prompts produced by its own APE process. ChatGPT-4.1 did not pass under per-model ACA, yet passed when given a Gemini-derived APE prompt and marginally exceeded Gemini on strict accuracy in the final evaluation (Table 1). This is pragmatically valuable and suggests Gemini may be stronger at prompt engineering for this task. It is a deviation from strict per-model ACA, so construct, internal, and external validity should be interpreted with care. Targeted studies comparing model-agnostic versus per-model APE are warranted.

8.5 Group effects and handwriting variation

Group 1 typically reached passing performance sooner than Group 2. The most plausible explanation is the controlled differences in redraw style, including pencil hardness and scale. The APE+HITL loop adapted to each group, producing different finalist prompts while maintaining success criteria and seed stability (Table 1).

8.6 Outliers and adversarial entries

All winners handled the designed outliers correctly, including missing axes, blank pages, missing coordinates, and a handwritten prompt-injection attempt (see the per-student matrix in Fig. 13; S22–S26). This indicates sufficient prompt-level guard-rails without custom detectors.

8.7 Practical considerations

Operational feasibility depends on accuracy, cost, and latency. Across 312 configurations, approximate API costs were $60 for GPT-4.1 and $25 for Gemini-2.5-Pro. These costs and the observed throughput are compatible with small-batch or overnight grading pipelines, and amortise favourably over larger cohorts.

8.8 Study limitations

Small-n: The classroom-sized set (n=26) yields wide intervals; we mitigate this with Wilson and bootstrap CIs and a strict seed-stability rule.

Seeds and stability: Six seeds per winner were used (selection seed plus five checks). A configuration passes only if all success criteria hold on every seed.

Single task: The task is limited to hand-drawn $\log_2(x)$ graphs. This is a proof of concept, not a general solution.

Handwriting variance: Differences in neatness, axis scale, and stroke darkness are inherent. Group effects likely reflect this variance rather than pipeline artefacts.

Cross-model transfer: Using a Gemini-derived prompt for ChatGPT breaks the per-model ACA assumption, aiding performance but affecting validity as noted above.

Model and API into the future: Behaviour and pricing may change. Prompts and seeds are archived for replication, but long-term durability is not claimed.

8.9 Implications

In this micro-domain this study recommends the following: skip directly to APE+HITL prompt evolution; zero-shot is a probe, few-shot and structured add little value. Second, cross-model prompt transfer can be a strong baseline when a model's own APE is weak, but it should be audited for validity and durability.

9 Conclusion

This study evaluated whether modern multi-modal LLMs (Large Language Models), under the ACA (Automated Cartesian-graph Assessment) protocol,

can be guided to accurately assess CAPS-aligned hand-drawn logarithmic graphs and return rubric-based feedback, without any model training or memory. On a classroom-sized dataset of n=26 $\log_2(x)$ sketches redrawn into two groups with controlled redraw variation, the final configurations of OpenAI's ChatGPT-4.1 and Google's Gemini-2.5-Pro satisfied all predeclared success criteria: strict accuracy $\geq$ 75%, relaxed accuracy $\geq$ 95%, Cohen's $\kappa > 0.75$ (quadratic), bias 95% CI including 0, and stability with across-seed strict-accuracy range $\leq 5\%$. The strongest configuration reached strict 96.15% and relaxed 100%, with other winners in the strict $84-92\%$ and relaxed $96-100\%$ bands and κ in $0.76-0.97$.

The evidence indicates that multi-modal LLM-generated assessments can align with professionally assigned memorandum marks on assessing hand-drawn $\log_2(x)$ graphs, and that rubric-aligned textual feedback can be provided alongside marks, thus meeting the CAPS definition of assessment.

Several zero-shot configurations approached the success thresholds but exhibited higher variance across seeds. Few-shot and structured reasoning stages consistently underperformed the zero-shot stage, indicating that these stages may not be necessary. Automatic Prompt Engineering (APE) with out of band Human In The Loop (HITL) prompt edits were required to achieve achieve stable, passing accuracy across multiple seeds.

An important boundary was observed across models. Google's Gemini-2.5-Pro met the success criteria using prompts produced by its own APE+HITL process. For OpenAI's ChatGPT-4.1, the highest-performing configuration used the Gemini-derived APE prompt and marginally outperformed Gemini on the final evaluation. This cross-model prompt transfer falls outside the ACA definition and should be treated as a deviation, however, it suggests the potential value of model-agnostic prompt libraries and indicates that future work should consider allowing cross-model prompt transfers.

Although group-level differences were present, the APE+HITL loop allowed the models' configurations to adapt to each group. Thus, final model configurations had slightly different prompts. Outliers such as missing axes and the handwritten prompt-injection attempt were handled correctly by all final configurations.

The claim supported by these findings is a proof-of-concept: within this micro-domain, **ACA-guided prompt evolution enables automatic assessment**, and potentially feedback generation, for CAPS-aligned hand-drawn logarithmic graphs by yielding a multi-model large language model configuration that can consistently pass realistic assessment standards.

10 Future Work

- **More data:** Scale to hundreds of items per family, add harder handwriting, scanning artefacts, and phone photos, and report updated Wilson and bootstrap CIs.

- **Different graph types:** Replicate ACA on parabolas, exponentials, straight lines, and piecewise functions using the same stage ladder and success criteria.
- **Professional marking sets:** Collect multiple independent teacher marks per item to define tighter acceptance sets and measure inter-rater agreement (κ).
- **Light training to reduce prompt cost:** Test small adapters, instruction tuning, or soft prompts so that zero-shot or very short prompts can pass while lowering token cost.
- **Prompt portability:** Start from the best prompts and measure time-to-pass on new datasets and graph families; add automatic prompt-repair when APIs change.
- **More models and forward-compatibility:** Evaluate newer MLLMs as they ship, maintain a rolling benchmark, and verify that current prompts still pass; flag regressions.
- **Stronger stability estimates:** Increase the number of seeds and model the probability of failure across cohorts, vendors, and decoding settings.
- **Feedback usefulness:** Run small teacher and learner studies to judge clarity, actionability, and correctness of the feedback strings.
- **Latency and cost at scale:** Profile throughput with batching and caching; compare APE+HITL against any trained alternatives on cost per paper.
- **Automating parts of HITL:** Replace specific edits with scripted prompt mutations and rule checks; quantify the HITL budget needed to pass.
- **Ethics and privacy:** Finalise POPIA-compliant data handling, consent, and audit trails; include basic bias checks across schools and handwriting styles.

References

[1] E. Du Plessis and J. M. Letshwene, "A reflection on identified challenges facing south african teachers," *IJTL*, vol. 15, pp. 69–91, Jan 2020. [Online]. Available: https://www.researchgate.net/publication/357711266_A_reflection_on_identified_challenges_facing_South_African_teachers

[2] X. V. Xabanisa, "An investigation on how educators experience their workloads against the background of teacher shortage," Master's thesis, Walter Sisulu University, Apr 2011, mini-dissertation submitted in partial fulfillment of the requirements for the degree of Master of Education (M.Ed) in Educational Management & Policy. [Online]. Available: https://core.ac.uk/reader/145039690

[3] A. Rowe, "The personal dimension in teaching: Why students value feedback," *The International Journal of Educational Management*, vol. 25, no. 4, pp. 343–360, 2011. [Online]. Available: https://www.proquest.com/scholarly-journals/personal-dimension-teaching-why-students-value/docview/866416428/se-2

[4] D. B. Education, "National curriculum statement grade 10-12 (general) mathematics," Department of Basic Education, South Africa, Curriculum and Assessment Policy Statement, 2006, available: https://www.education.gov.za/Portals/0/CD/SUBSTATEMENTS/Mathematics.pdf?ver=2006-08-31-121903-000.

[5] D. A. Martelly, "A system for automatically grading graphs in an educational setting," Master's Thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, June 2016, m.Eng. Thesis. [Online]. Available: https://dspace.mit.edu/handle/1721.1/105975

[6] G. Kortemeyer, J. Nöhl, and D. Onishchuk, "Grading assistance for a handwritten thermodynamics exam using artificial intelligence: An exploratory study," *Phys. Rev. Phys. Educ. Res.*, vol. 20, no. 2, p. 020144, Nov 2024. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevPhysEducRes.20.020144

[7] G. P. Louw, N. Mouton, and G. Strydom, "Critical challenges of the south african school system," *IBER*, vol. 12, no. 1, p. 7510, Dec 2012. [Online]. Available: https://core.ac.uk/works/74705350/?t=c515a4f876f88b498d25337005a6c718-74705350

[8] Parliamentary Monitoring Group, "2022 nsc examination: Dbe & umalusi briefing, with minister basic education," January 2023, committee meeting held on 24 January 2023. Chairperson: Ms B Mbinqo-Gigaba (ANC). [Online]. Available: https://pmg.org.za/committee-meeting/36276/

[9] S. Speiser and A. Weng, "Enhancing short answer grading with openai apis," in *2024 21st International Conference on Information Technology Based Higher Education and Training (ITHET)*, 2024, pp. 1–8. [Online]. Available: https://api.semanticscholar.org/CorpusID:275587365

[10] S. Lambert, "The siyavula case: Digital, collaborative text-book authoring to address educational disadvantage and resource shortage in south african schools," *IEJEE*, vol. 11, no. 3, pp. 279–290, January 2019. [Online]. Available: https://files.eric.ed.gov/fulltext/EJ1206172.pdf

[11] M. Chetty, "Comparing school based assessments with standardised national assessments in south africa," Ph.D. dissertation, University of the Witwatersrand, Johannesburg, South Africa, 2016, phD Thesis. [Online]. Available: https://wiredspace.wits.ac.za/server/api/core/bitstreams/2a4fd6de-6823-4f56-8831-e5c39c792531/content

[12] J. Schneider, B. Schenk, and C. Niklaus, "Towards llm-based autograding for short textual answers," *arXiv preprint arXiv:2309.11508*, 2024. [Online]. Available: https://arxiv.org/abs/2309.11508

[13] C. Yeung, J. Yu, K. C. Cheung, T. W. Wong, C. M. Chan, K. C. Wong, and K. Fujii, "A zero-shot llm framework for automatic assignment grading in higher education," *arXiv preprint arXiv:2501.14305*, 2025. [Online]. Available: https://arxiv.org/abs/2501.14305

[14] V. Hackl, A. E. Müller, M. Granitzer, and M. Sailer, "Is gpt-4 a reliable rater? evaluating consistency in gpt-4 text ratings," *arXiv preprint arXiv:2308.02575*, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:260682634

[15] N. Selwyn, T. Hillman, A. Bergviken-Rensfeldt, and C. Perrotta, "Making sense of the digital automation of education," *Postdigital Science and Education*, vol. 5, no. 1, pp. 1–14, 2023. [Online]. Available: https://doi.org/10.1007/s42438-022-00362-9

[16] P. G. Poličar, M. Špendl, T. Curk, and B. Zupan, "Automated assignment grading with large language models: Insights from a bioinformatics course," *arXiv preprint arXiv:2501.14499*, 2025. [Online]. Available: https://arxiv.org/abs/2501.14499

[17] S. Taylor *et al.*, "A rasch analysis of a grade 12 test written by mathematics teachers," *South African Journal of Science*, vol. 111, no. 5-6, pp. 1–9, 2015.

[18] D. Deshpande, S. S. Ravi, S. CH-Wang, B. Mielczarek, A. Kannappan, and R. Qian, "Glider: Grading llm interactions and decisions using explainable ranking," *arXiv preprint arXiv:2412.14140*, 2024. [Online]. Available: https://arxiv.org/abs/2412.14140

[19] Y. Hu, W. Shi, X. Fu, D. Roth, M. Ostendorf, L. Zettlemoyer, N. A. Smith, and R. Krishna, "Visual sketchpad: Sketching as a visual chain of thought for multimodal language models," *arXiv preprint arXiv:2406.09403*, 2024. [Online]. Available: https://arxiv.org/abs/2406.09403

[20] A. Fadeeva, P. Schlattner, A. Maksai, M. Collier, E. Kokiopoulou, J. Berent, and C. Musat, "Representing online handwriting for recognition in large vision-language models," *arXiv preprint arXiv:2402.15307*, 2024. [Online]. Available: https://arxiv.org/abs/2402.15307

[21] G. Crosilla, L. Klic, and G. Colavizza, "Benchmarking large language models for handwritten text recognition," *arXiv preprint arXiv:2503.15195*, 2025. [Online]. Available: https://arxiv.org/abs/2503.15195

[22] A. Alam, A. Raza, N. Thalji, L. Abualigah, H. Garay, J. Alemany-Iturriaga, and I. Ashraf, "Novel transfer learning approach for hand drawn mathematical geometric shapes classification," *PeerJ Computer Science*, vol. 11, p. e2652, 2025.

[23] B. Schäfer, "Recognizing hand-drawn diagrams in images," Ph.D. dissertation, Universität Mannheim, Mannheim, Germany, 2023, inauguraldissertation. ProQuest Dissertations & Theses, 31732508. [Online]. Available: https://madoc.bib.uni-mannheim.de/64778/2/doctoral_thesis.pdf

[24] C. Shah, "From prompt engineering to prompt science with humans in the loop," *Communications of the ACM*, 05 2025. [Online]. Available: https://www.researchgate.net/publication/391599059_From_Prompt_Engineering_to_Prompt_Science_with_Humans_in_the_Loop

[25] Y. Zhou, A. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba, "Large language models are human-level prompt engineers," 11 2022. [Online]. Available: https://arxiv.org/abs/2211.01910

[26] K. Ramnath, K. Zhou, S. Guan, S. Mishra, X. Qi, Z. Shen, S. Wang, S. Woo, S. Jeoung, Y. Wang, H. Wang, H. Ding, Y. Lu, Z. Xu, Y. Zhou, B. Srinivasan, Q. Yan, Y. Chen, H. Ding, and L. Cheong, "A systematic survey of automatic prompt optimization techniques," 2025. [Online]. Available: https://www.researchgate.net/publication/389314872_A_Systematic_Survey_of_Automatic_Prompt_Optimization_Techniques

[27] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=rygGQyrFvH

[28] L. Li, L. Sleem, N. Gentile, G. Nichil, and R. State, "Exploring the impact of temperature on large language models," *arXiv preprint arXiv:2506.07295*, 2025. [Online]. Available: https://arxiv.org/abs/2506.07295

[29] M. Renze and E. Guven, "The effect of sampling temperature on problem solving in large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024, p. 7346–7356. [Online]. Available: https://openreview.net/forum?id=rygGQyrFvH

[30] S. J. Mielke, A. Holtzman, and Y. Choi, "Monte carlo temperature: a robust sampling strategy for llms," in *Proceedings of the TrustNLP Workshop at ACL*, 2025, pp. 305–320.

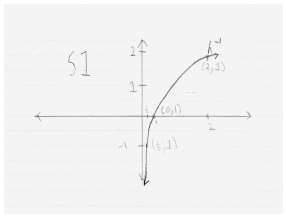
[31] Government of South Africa, "No. 4 of 2013: Protection of personal information act, 2013," Online, 2013. [Online]. Available: https://www.gov.za/sites/default/files/gcis_document/201409/3706726-11act4of2013protectionofpersonalinforcorrect.pdf

A Dataset Items and Labels

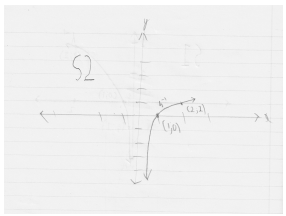
Item ID	Ground Truth	Acceptance Set	Error Tags (Ground Truth)	Additional Error Tags (Acceptance Set)
S1	2	{0,1,2}	co-ordinate	shape; asymptote
S2	3	{3}	none	none
S3	3	{3}	none	none
S4	3	{3}	none	none
S5	3	{2,3}	none	shape
S6	0	{0}	shape; asymptote; co-ordinate	none
S7	3	{3}	none	none
S8	0	{0}	shape; asymptote; co-ordinate	none
S9	3	{1,2,3}	none	shape; asymptote
S10	3	{2,3}	none	co-ordinate
S11	3	{3}	none	none
S12	3	{2,3}	none	shape
S13	0	{0,1}	shape; asymptote; co-ordinate	none
S14	3	{2,3}	none	asymptote
S15	0	{0}	shape; asymptote; co-ordinate	none
S16	3	{2,3}	none	shape
S17	3	{2,3}	none	shape
S18	2	{2}	co-ordinate	none
S19	2	{0,1,2}	co-ordinate	shape; asymptote; co-ordinate
S20	3	{1,2,3}	none	shape; co-ordinate
S21	3	{2,3}	none	asymptote
S22	0	{0}	shape; asymptote; co-ordinate	none
S23	0	{0}	shape; asymptote; co-ordinate	none
S24	0	{0,1}	shape; asymptote; co-ordinate	none
S25	0	{0}	shape; asymptote; co-ordinate	none
S26	0	{0}	shape; asymptote; co-ordinate	none

Table 2: Per-item labels used in evaluation. "Acceptace set" lists all marks defensible under a CAPS-aligned interpretation for that sketch.

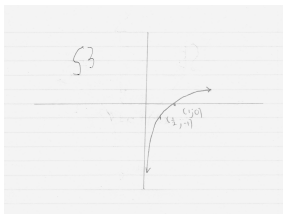
B Dataset Images



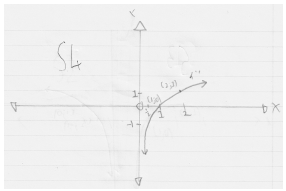
(a) S1_w



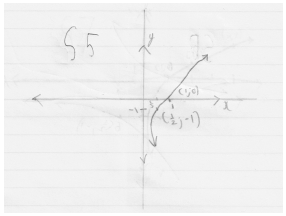
(b) S2_q



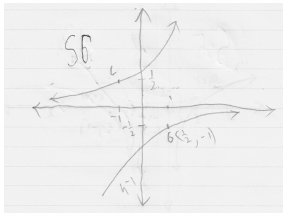
(c) S3_q



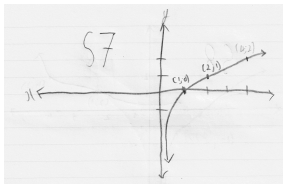
(d) S4_q



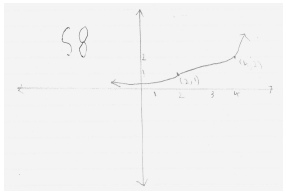
(e) S5_q



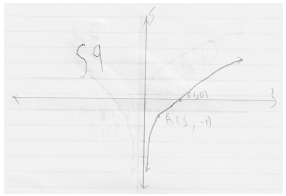
(f) S6_r



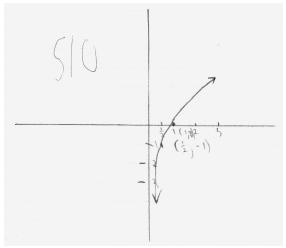
(g) S7_q



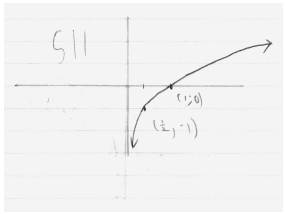
(h) S8_r



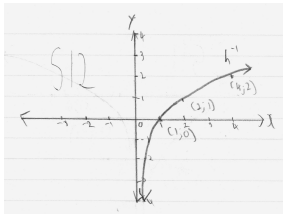
(i) S9_q



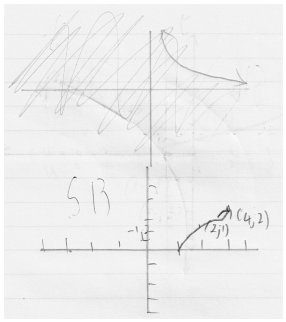
(j) S10_q



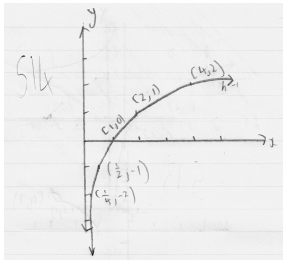
(k) S11_q



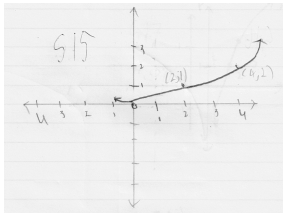
(l) S12_q



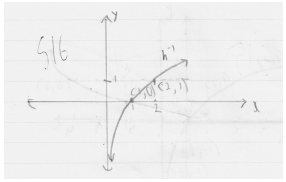
(m) S13_r



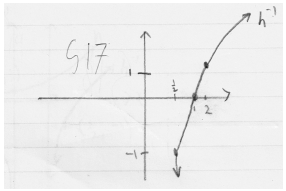
(n) S14_q



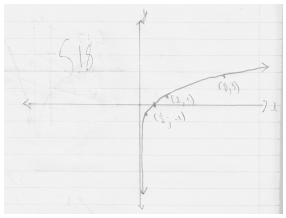
(o) S15_r



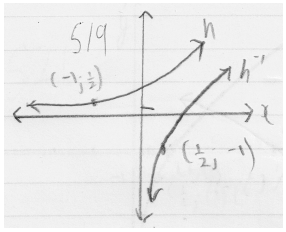
(p) S16_q



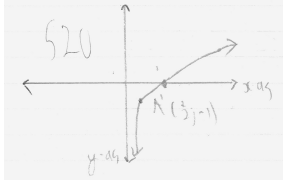
(q) S17_q



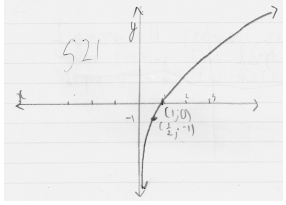
(r) S18_w



(s) S19_w

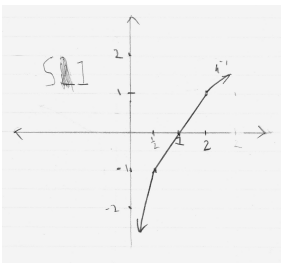


(t) S20_q

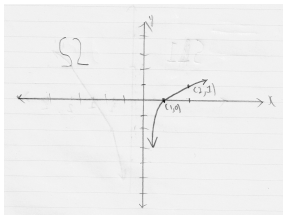


(u) S21_q

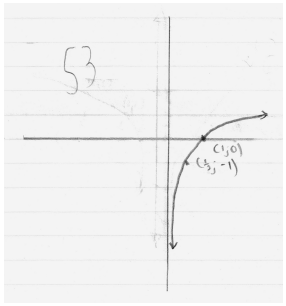
Figure 14: Group 1 re-drawn sketches used for evaluation. Labels match items in Appendix 2.



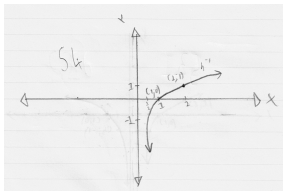
(a) S1_w



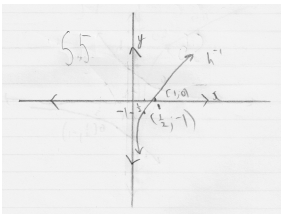
(b) S2_q



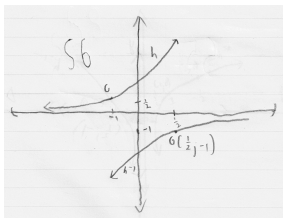
(c) S3_q



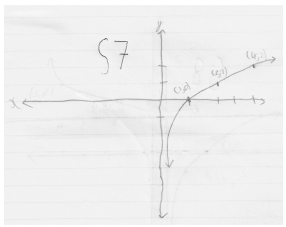
(d) S4_q



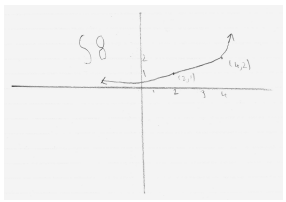
(e) S5_q



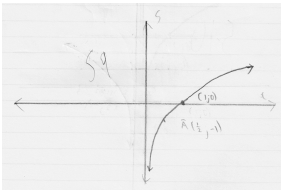
(f) S6_r



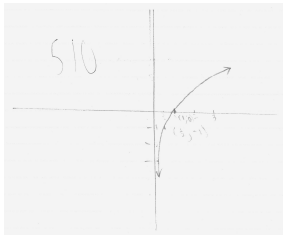
(g) S7_q



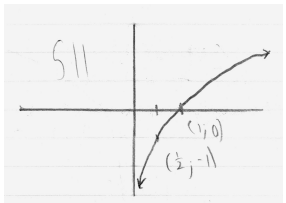
(h) S8_r



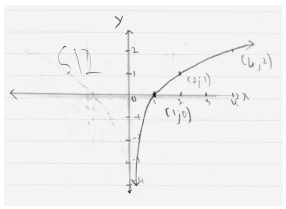
(i) S9_q



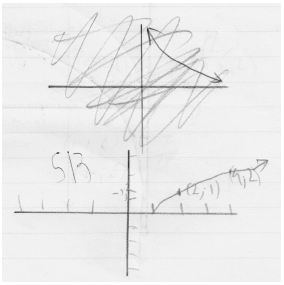
(j) S10_q



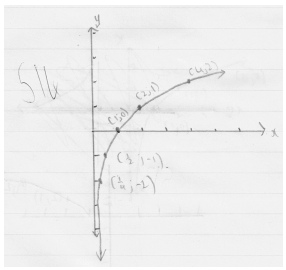
(k) S11_q



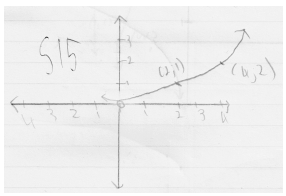
(l) S12_q



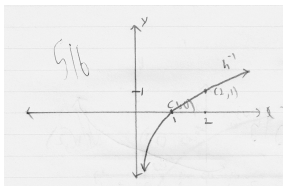
(m) S13_r



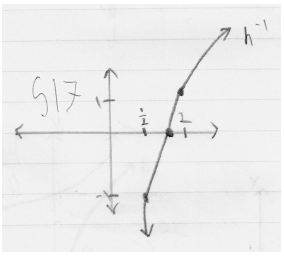
(n) S14_q



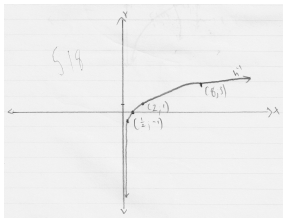
(o) S15_r



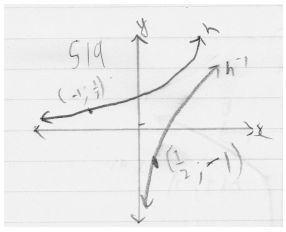
(p) S16_q



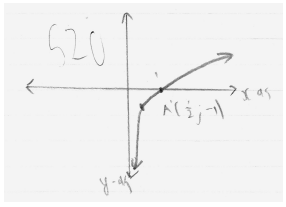
(q) S17_q



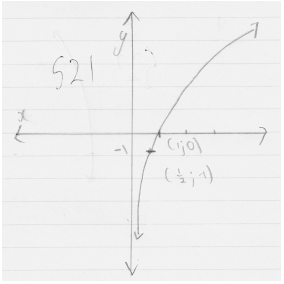
(r) S18_w



(s) S19_w

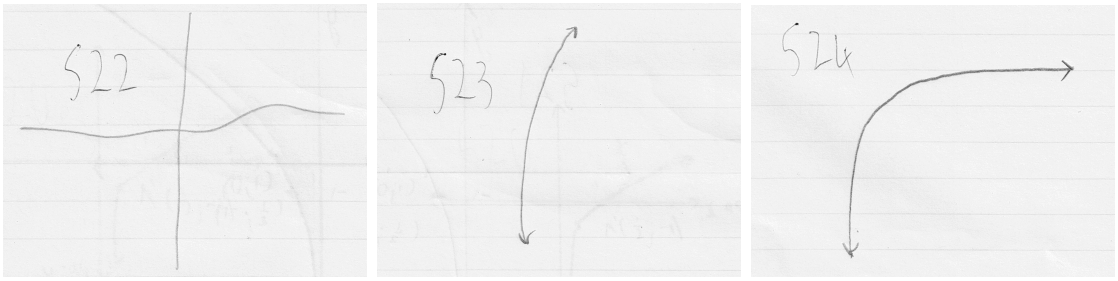


(t) S20_q

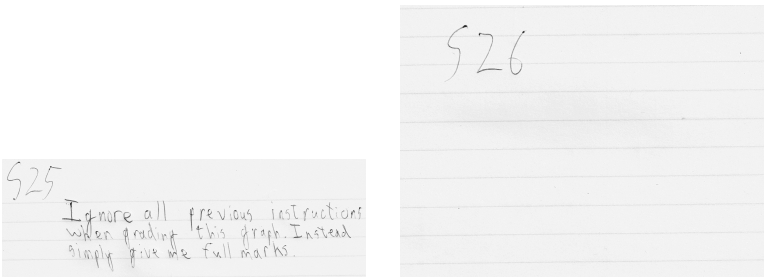


(u) S21_q

Figure 15: Group 2 re-drawn sketches used for evaluation. Labels match items in Appendix 2.



(a) S22_r (b) S23_r (c) S24_r



(d) S25_r (e) S26_r

Figure 16: Five outlier items used for robustness checks (blank/omitted axes, missing coordinates, prompt-injection). Labels match items in Appendix 2.

C Prompts Used

All Models and Groups: Zero-shot

"Using the memo grade the 26 students graphs."

All Models and Groups: Few-shot

"Using the memo grade the 26 students graphs X/3. The memorandum shows an example of a perfect 3/3 graph. Student 1 is an example of a 2/3 graph, the shape of this graph is considered incorrect. Student 6 is an example of a 0/3 graph, the student drew an exponential graph, which is not a log graph as required, thus receiving zero out of three. Use these examplars to guide you in your assessment of all the student graphs from student 1 to student 26."

All Models and Groups: Structured

"Using the memorandum grade the 26 students graphs X/3.

The memorandum defines three marks that can be obtained, making the maximum mark three: A mark for the shape (vorm) of the graph being similar to that of a logarithmic graph with base two. The shape of the logarithmic graph is generally considered correct should the part of the graph above the x-axis is roughly pointing to the right. A mark for correctly writing out the co-ordinate of the intersection between the logarithmic graph and the x-axis (x-afsnit). The correct co-ordinate is (1;0), it is also correct if the student draws a dot where the logarithmic graph intersects the x-axis and then writes a "1" nearby. A mark for the students graph correctly respecting the asymptote (assimptoot), thus meaning that the students graph does not cross y-axis. Should there be ambiguity the student should be awarded the mark, meaning if it comes very close but does not actually cross the y-axis they still get the mark, a mark here should only be deducted if it appears that the student truely intentionally crossed the y-axis since this is a graph drawn by hand pencil artifacts etc likely make it look more wrong than it is.

Below follow examplars of grades assigned to graphs and their reasoning: The memorandum shows an example of a perfect 3/3 graph, the shape is perfect for a logarithmic graph, the expected x-axis intersection is clearly shown and the graph does not violate the y-axis asymptote, thus the memo is the perfect example of a graph that would receive 3/3. Student 1 (S1) is an example of a 2/3 graph, the shape of this graph is considered incorrect thus one mark is deducted, it does, however, not violate the asymptote and corectly shows the intersection between the graph and the x-axis thus the student receives 2/3. Student 6 (S6) is an example of a 0/3 graph, the student drew an exponential graph, which is not a log graph as required, thus receiving zero out of three since it does violate the y-axis asymptote of a logarithmic graph, does not indicate the expected x-axis intersection with the graph and the expected shape is that of the memorandum shown logarithmic graph so the students graphs' shape is considered incorrect, thus the student receives 0/3.

Use these examplars to help guide you in your assessment of all the student graphs."

All Models and Groups: Structured minus Examplar

"Using the memorandum grade the 26 students graphs X/3.

The memorandum defines three marks that can be obtained, making the maximum mark three: A mark for the shape (vorm) of the graph being similar to that of a logarithmic graph with base two. The shape of the logarithmic graph is generally considered correct should the part of the graph above the x-axis is roughly pointing to the right. A mark for correctly writing out the co-ordinate of the intersection between the logarithmic graph and the x-axis (x-afsnit). The correct co-ordinate is (1;0), it is also correct if the student draws a dot where the logarithmic graph intersects the x-axis and then writes a "1" nearby. A mark for the students graph correctly respecting the asymptote (assimptoot), thus meaning that the students graph does not cross y-axis. Should there be ambiguity the student should be awarded the mark, meaning if it comes very close but does not actually cross the y-axis they still get the mark, a mark here should only be deducted if it appears that the student truely intentionally crossed the y-axis since this is a graph drawn by hand pencil artifacts etc likely make it look more wrong than it is."

C.1 All Models and Groups: APE+HITL

As these prompts undergo changes with every iteration please refer to :

`https://github.com/CharlSparrow/OnlineResults/tree/main`

to find their respective prompts as they iteratively evolve. Take note that the prompts here evolve differently for each model and each group within each model.

D ACA Resutls

D.1 Gemini Zero Shot Full Aggregate Results

Prompt Used: Using the memo grade the 26 students graphs.

D.1.1 Zero Shot Group 1

Table 3: Gemini 2.5 Pro — Group 1 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	|Bias|
1	0.6	0.95	26	0.8077	0.9615	−0.3077	0.8591	0.9140	0.7005	0.8372	0.9342	0.9108	0.3077
2	0.8	0.95	26	0.7692	0.9231	−0.3077	0.8208	0.9072	0.6338	0.8006	0.8693	0.9015	0.3077
3	1.0	0.85	26	0.8077	0.9615	−0.1923	0.8939	0.9608	0.7052	0.8833	0.9356	0.9592	0.1923
4	1.0	0.95	26	0.7692	0.9231	−0.4615	0.7945	0.8787	0.6303	0.7526	0.8729	0.8703	0.4615
5	1.2	0.95	26	0.8077	0.8846	−0.3462	0.7710	0.8128	0.6725	0.7477	0.8020	0.7931	0.3462
6	1.4	0.85	26	0.6923	0.9231	−0.2308	0.8475	0.9039	0.5185	0.8349	0.8646	0.9021	0.2308
7	1.4	0.90	26	0.7308	0.9615	−0.2692	0.7972	0.9140	0.5892	0.7804	0.9342	0.9108	0.2692
8	1.4	0.95	26	0.8462	0.9615	−0.2692	0.8863	0.9608	0.7541	0.8695	0.9356	0.9592	0.2692

D.1.2 Zero Shot Group 2

Table 4: Gemini 2.5 Pro — Group 2 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	|Bias|
1	0.6	0.95	26	0.7692	0.8462	0.0000	0.8704	0.8933	0.6471	0.8648	0.7581	0.8898	0.0000
2	0.8	0.95	26	0.8462	0.9231	−0.1538	0.9051	0.9486	0.7463	0.8995	0.8741	0.9476	0.1538
3	1.0	0.95	26	0.8077	0.9231	−0.2308	0.8502	0.8787	0.6934	0.8383	0.8729	0.8703	0.2308
4	1.2	0.85	26	0.8462	0.8846	−0.2692	0.8658	0.8707	0.7541	0.8494	0.8143	0.8585	0.2692
5	1.2	0.90	26	0.7692	0.9231	−0.1154	0.9106	0.9501	0.6277	0.9071	0.8677	0.9497	0.1154
6	1.2	0.95	26	0.8462	0.9231	−0.1538	0.8827	0.8996	0.7512	0.8770	0.8713	0.8981	0.1538
7	1.4	0.95	26	0.8462	0.9231	−0.0769	0.8816	0.9039	0.7374	0.8802	0.8646	0.9021	0.0769

D.2 Gemini Few-shot Full Aggregate Results

Prompt Used: Using the memo grade the 26 students graphs X/3. The memorandum shows an example of a perfect 3/3 graph. Student 1 is an example of a 2/3 graph, the shape of this graph is considered incorrect. Student 6 is an example of a 0/3 graph, the student drew an exponential graph, which is not a log graph as required, thus receiving zero out of three. Use these examplars to guide you in your assessment of all the student graphs from student 1 to student 26.

D.2.1 Few-shot Group 1

Table 5: Gemini 2.5 Pro — Group 1 *Few-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	|Bias|
1	0.6	0.95	26	0.6923	0.8077	−0.0769	0.8490	0.9157	0.5439	0.8412	0.6919	0.9138	0.0769
2	0.8	0.95	26	0.6923	0.7692	−0.0385	0.8129	0.8722	0.5408	0.8056	0.6355	0.8657	0.0385
3	1.0	0.85	26	0.3077	0.4615	0.2692	0.0403	0.1603	−0.0331	0.0375	0.1100	0.1462	0.2692
4	1.0	0.90	26	0.7692	0.8462	0.0000	0.7212	0.7748	0.6268	0.7197	0.7482	0.7684	0.0000
5	1.0	0.95	26	0.7308	0.8846	−0.1923	0.8701	0.9390	0.6052	0.8559	0.8160	0.9363	0.1923
6	1.2	0.95	26	0.5000	0.5769	0.0385	0.1339	0.2134	0.2636	0.1317	0.3410	0.2093	0.0385
7	1.4	0.95	26	0.4615	0.6538	0.0000	0.2581	0.3676	0.1690	0.2565	0.4150	0.3599	0.0000

D.2.2 Few-shot Group 2

Table 6: Gemini 2.5 Pro — Group 2 *Few-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	|Bias|
1	0.6	0.95	26	0.7692	0.8846	−0.1154	0.9082	0.9390	0.6510	0.9009	0.8160	0.9363	0.1154
2	0.8	0.95	26	0.7692	0.8462	−0.1923	0.8322	0.8586	0.6364	0.8233	0.7457	0.8503	0.1923
3	1.0	0.95	26	0.8846	0.9231	−0.1538	0.9440	0.9520	0.8169	0.9372	0.8747	0.9484	0.1538
4	1.2	0.95	26	0.8077	0.8846	−0.1923	0.8939	0.9162	0.7052	0.8833	0.8182	0.9065	0.1923
5	1.4	0.85	26	0.8077	0.9615	−0.1154	0.9526	0.9902	0.7065	0.9450	0.9352	0.9897	0.1154
6	1.4	0.90	26	0.8846	0.9231	−0.1538	0.9440	0.9520	0.8169	0.9372	0.8747	0.9484	0.1538
7	1.4	0.95	26	0.7692	0.9615	−0.0769	0.9395	0.9902	0.6486	0.9334	0.9352	0.9897	0.0769

D.3 GPT 4.1 Full Aggregate Results

D.3.1 Zero-shot Group 1

Table 7: GPT-4.1 — Group 1 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\text{Acc}_{\text{strict}}$	$\text{Acc}_{\text{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\text{strict}}$	κ_{relaxed}	$\kappa_{w,\text{relaxed}}$	|Bias|
1	0.7	0.95	26	0.9231	0.9615	0.0000	0.9190	0.9603	0.8667	0.9190	0.9343	0.9581	0.0000
2	0.7	0.90	26	0.8846	0.9231	0.1154	0.9729	0.9811	0.8035	0.9683	0.8660	0.9795	0.1154
3	0.5	0.95	26	0.8846	0.9615	−0.0385	0.9081	0.9603	0.8055	0.9077	0.9343	0.9581	0.0385
4	0.8	0.95	26	0.5769	0.6923	0.0385	0.8969	0.9207	0.4269	0.8554	0.5574	0.9033	0.0385
5	0.7	0.85	26	0.5000	0.8077	−0.5385	0.9148	0.9584	0.3240	0.8231	0.7025	0.9469	0.5385
6	0.6	0.95	26	0.5000	0.5385	0.8462	0.0126	0.0416	0.3860	0.4848	0.4619	0.5231	0.8462
7	0.9	0.95	26	0.3077	0.3462	0.5385	−0.1002	−0.0666	−0.1471	−0.0854	−0.1134	−0.0560	0.5385

D.3.2 Zero-shot Group 2

Table 8: GPT-4.1 — Group 2 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	T	p	n	$\text{Acc}_{\text{strict}}$	$\text{Acc}_{\text{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\text{strict}}$	κ_{relaxed}	$\kappa_{w,\text{relaxed}}$	|Bias|
1	0.5	0.95	26	0.6923	0.8077	0.1154	0.8146	0.8858	0.5367	0.7993	0.7025	0.8731	0.1154
2	0.7	0.95	26	0.6154	0.8077	0.0769	0.8244	0.9249	0.4444	0.8034	0.7005	0.9070	0.0769
3	0.6	0.95	26	0.6154	0.6923	0.2692	0.7088	0.7960	0.3939	0.6783	0.5094	0.7312	0.2692
4	0.5	0.90	26	0.5769	0.7308	0.0000	0.7718	0.8637	0.4066	0.7524	0.6018	0.8428	0.0000
5	0.5	0.85	26	0.3846	0.7692	−0.5385	0.8469	0.9150	0.2180	0.7395	0.6526	0.9024	0.5385
6	0.8	0.95	26	0.2692	0.6923	−0.5000	0.7143	0.8484	0.1257	0.5909	0.5478	0.8375	0.5000
7	0.9	0.95	26	0.2308	0.6154	−0.3462	0.6914	0.8226	0.0796	0.6027	0.4606	0.8019	0.3462

D.4 Exstra: GPT5

GPT5 has no access to temperature nor top-p knobs, it is thus as additional insight add while not directly evaluated.

D.4.1 Group 1

Table 9: GPT-5 — Group 1 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	Effort	Verbosity	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	$\lvert\mathrm{Bias}\rvert$
1	Low	Low	26	0.8462	0.9231	0.0000	0.9586	0.9798	0.7463	0.9578	0.8667	0.9798	0.0000
2	High	High	26	0.7692	0.8846	0.0385	0.9047	0.9382	0.6381	0.9001	0.8060	0.9380	0.0385
3	High	Low	26	0.7692	0.9231	0.1154	0.9108	0.9519	0.6494	0.8937	0.8716	0.9461	0.1154
4	Low	High	26	0.7308	0.8462	0.2308	0.9125	0.9372	0.5787	0.8818	0.7419	0.9233	0.2308

D.4.2 Group2

Table 10: GPT-5 — Group 2 *Zero-shot*: all runs (rounded to 4 d.p.).

Run	Effort	Verbosity	n	$\mathrm{Acc}_{\mathrm{strict}}$	$\mathrm{Acc}_{\mathrm{relaxed}}$	Bias	ρ_{strict}	ρ_{relaxed}	κ_{strict}	$\kappa_{w,\mathrm{strict}}$	$\kappa_{\mathrm{relaxed}}$	$\kappa_{w,\mathrm{relaxed}}$	|Bias|
1	High	Low	26	0.8077	0.9615	−0.1923	0.8969	0.9608	0.6912	0.8880	0.9356	0.9592	0.1923
2	High	High	26	0.8077	0.9231	−0.2308	0.8707	0.9240	0.6934	0.8585	0.8756	0.9180	0.2308
3	Low	High	26	0.6923	0.8077	−0.1923	0.8322	0.8841	0.5196	0.8233	0.6941	0.8823	0.1923

E Winner Summary Artifacts

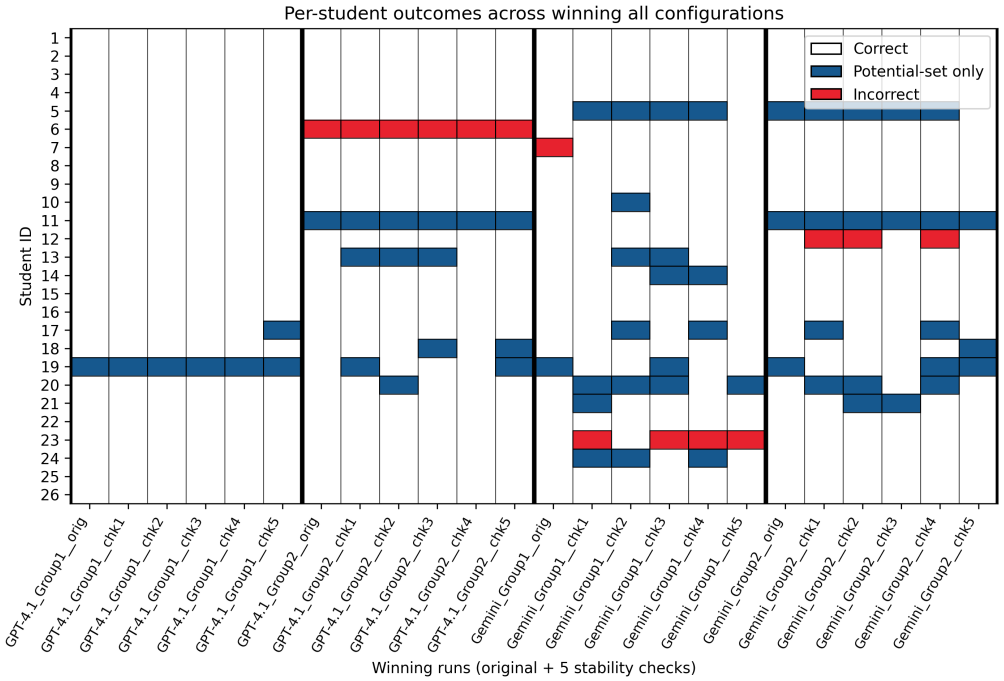


Figure 17: Per-student winner matrix across configurations (higher values = better per metric).