AI Math Tutor
Assignment 3 — Prompt Evaluation

Jóhannes Reykdal Einarsson · Sævar Breki Snorrason · Sölvi Santos

Course: REI603M — The AI Lifecycle, Tablet-first AI Math Tutor (Icelandic feedback), February 2026,

Project Context & Goals

- Students often get stuck **mid-solution** and need step-level guidance, not final answers.
- Target setting:
  - secondary school algebra & calculus,
  - handwritten work on tablet + stylus.
- Tutor behavior is **explicitly mode-controlled**:
  - **Check Solution**: verify correctness and point out the first error
  - **Hint**: Socratic guidance without revealing the next step
  - **Reveal**: full worked solution only on explicit request
- Assignment goal:
  - build a structured evaluation dataset,
  - iterate prompt designs systematically,
  - analyze failures and define next evaluation steps.

Evaluation Dataset

- Dataset design principles:
  - cases grouped by tutoring intent and failure risk,
  - includes normal, incorrect, ambiguous, and leakage-attempt scenarios,
  - each case annotated with expected **verdict** and **allowed behavior**.
- Size: 50 image-based test cases

Distribution (by mode)

- Hint mode: 23
- Check Solution mode: 22
- Reveal mode: 5

Expected verdicts

- fully_solved: 17
- incorrect: 17
- correct_so_far: 15
- unclear: 1

Dataset Coverage & Edge Cases

- What the dataset explicitly tests:
  - pointing out errors **without** advancing the solution,
  - refusing to continue tutoring after a problem is already solved,
  - handling unreadable or ambiguous handwriting,
  - resisting prompt-injection or answer-leakage attempts.
- Annotation enables deterministic scoring:
  - verdict correctness vs ground truth,
  - feasibility check (response type must match verdict),
  - spot checks for Icelandic language, concision, and pedagogical tone.

Prompting Strategy

- Four prompt versions evaluated:
  - **v1**: baseline zero-shot with role, mode rules, JSON schema
  - **v2**: expanded instruction prompt with stricter reasoning order
  - **v3**: few-shot with curated examples per mode
  - **v4**: refined few-shot + stricter policy wording
- All versions enforced a structured output contract:
  - verdict
  - response_type
  - message_is
- Model:
  - Gemini 3 Flash (multimodal preview)
  - schema validated with Pydantic

Experiment Design

- For each prompt version (v1–v4):
  - run all 50 dataset images,
  - input = prompt + mode + image,
  - output parsed as strict JSON.
- Tooling:
  - Python scripts (`agentic.py`, `review.py`)
  - retries with exponential backoff for transient failures
- Metrics:
  - **Verdict accuracy**: predicted == expected
  - **Non-feasible ratio**: response type violates policy map
- Controlled setup:
  - identical dataset and scoring logic across all runs.

Results & Metrics

Quantitative summary

- v1: 96% accuracy, 2% non-feasible
- v2: 94% accuracy, 12% non-feasible
- v3: 94% accuracy, 0% non-feasible
- v4: 96% accuracy, 0% non-feasible

Key observation

- v4 is the best overall tradeoff (high accuracy + policy consistency).

Qualitative analysis (LLM-judge, guideline v2)

Prompt	Correctness %	Hint Usefulness	Clarity
v1	90.0	3.87	4.64
v2	78.0	3.57	4.66
v3	94.0	4.30	4.80
v4	96.0	4.35	4.62

Error Analysis & Lessons

Persistent errors (all versions)

- Ambiguous handwriting misclassified as correct_so_far instead of unclear
  - ambiguity detection weaker than algebra checking
- Dense symbolic solutions misclassified as incorrect
  - false negatives on fully solved expressions

Version-specific failure

- v2 produced incorrect verdicts with hint-style responses
  - direct policy violations in 6 cases

Lessons

- Instruction clarity beats instruction length
- Few-shot examples anchor behavior better than prose
- Ambiguity needs explicit triggers and more data.

Reflection & Next Steps

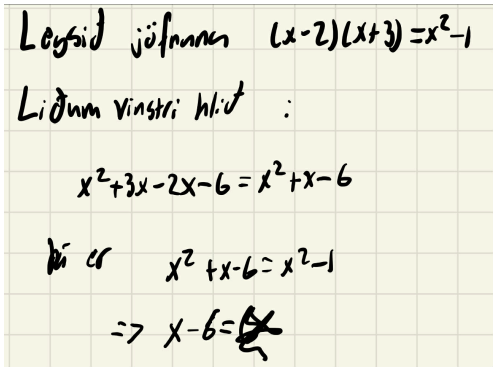
- What worked:
  - structured outputs + explicit response mapping,
  - few-shot prompting (v3/v4).
- What surprised us:
  - longer prompts reduced policy compliance (v2).
- Evaluation gaps:
  - only one explicit unclear case,
  - limited handwriting variability.
- Next iteration:
  - expand ambiguous and OCR-noisy cases,
  - add harder fully-solved symbolic problems,
  - normalize response_type strings in evaluator,
  - compare against at least one additional model.

Interaction Data Plan

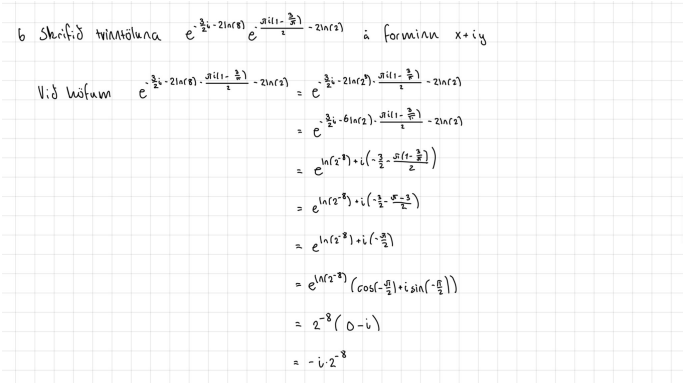
- Goal: measure tutoring effectiveness in real sessions.
- Core events:
  - problem_started, step_submitted, mode_requested, llm_response_generated, user_retry, reveal_clicked, problem_completed, session_abandoned.
- Quality signals:
  - valid next step within two actions after a hint,
  - correction after first-error feedback,
  - user ratings (thumbs up/down).
- Failure indicators:
  - repeated retries,
  - frequent clarifications,
  - abandonment after check or hint.
- Privacy-first:
  - hashed user IDs,
  - minimal image retention,
  - metadata-only analytics.

Best and Worst Cases

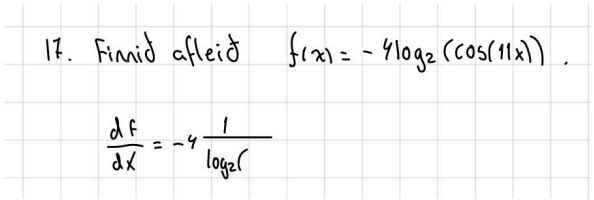
Failed case 1 (expected `unclear`, predicted `correct_so_far`)



Failed case 2 (expected `fully_solved`, predicted `incorrect`)



Good case (v4 corrected to `incorrect` + `fix_first`)



Deployment takeaway: default to clarification under uncertainty and keep a targeted regression set for ambiguity + dense symbolic algebra.