ThinkGrade

An approachable AI companion that explains math and grades essays—straight from your laptop, tablet, or even a low-cost cloud GPU.

1. Why ThinkGrade?

Most learning apps do one thing well: they either **show you how to solve a math problem** *or* **tell you whether your writing is any good**.

ThinkGrade does both. We fine-tuned Google's open-weight **Gemma-3 n-E2B-it** model so it can:

1. **Walk through math step by step**—with LaTeX and clear reasoning.
2. **Score short essays or summaries** on a 0–5 scale—and explain *why* a score was assigned.

Because everything runs in **4-bit quantization**, ThinkGrade fits comfortably on two modest T4 GPUs (or a single A100). That means schools—or parents—can host it privately without handing student data to a third-party API.

---

2. How It Works (Plain English!)

1. **You ask a question.**
   "Solve ∫x² dx." or *"Summarize this article in three sentences."*
2. **ThinkGrade builds a tidy prompt.**
   It prepends a "system" persona that keeps the model polite, safe, and organized, then adds your query.
3. **Gemma answers.**
   Behind the scenes, the model carries two LoRA "hats":
   - **Math-Tutor LoRA**: generates step-by-step solutions.
   - **Essay-Judge LoRA**: scores writing on logic, conciseness, and relevance.
4. **You get a friendly response.**
   - Math replies come in three parts: *Approach* → *Calculation* → *Final Answer* (all LaTeX-ready).
   - Essay replies return a score (0–5) plus two or three sentences of feedback.

3. Data We Fed the Model

Domain	Source & Size	What We Did
Math tutoring	30 k–100 k problems from **MathX-5M**	Stripped noisy <think> tags, kept final answers, wrapped each example in a chat template.
Essay tutoring	20 k CNN/DailyMail articles + highlights	Turned each article into a “write a summary” prompt; target = reference summary.
Math grading	10 k MathX-5M problems	Used GPT to write student-like answers, auto-labeled ✔/✘ via cosine similarity; balanced to 5 k.
Essay grading	2 k news articles	GPT-generated summaries; Sentence-BERT mapped similarity to scores 0–5.

Everything streams straight from Hugging Face, so we never blow past Kaggle’s RAM limit.

4. Training in a Nutshell

Part	Trick	Why It Matters
4-bit NF4 quantization	Shrinks memory ×4 with minimal quality loss.	

LoRA + QLoRA	Only 0.2 % of weights are trainable—perfect for hackathon budgets.	
Rank & α	r=4, α=8 (tutoring) ; r=8, α=16 (grading)	Lower rank = lower VRAM; higher rank when we need classifier muscle.
Grad accum 32	True batch 64 on two T4s.	
bf16 + gradient-check pointing	Faster math, fewer out-of-memory errors.	

End-to-end:

- **Math tutor SFT**: ~3 h on two T4s
- **Math classifier**: 45 min on an A100
- **Essay classifier**: 25 min on an A100

5. Does It Actually Work?

Task	Metric	Result
Solve-and-Explain	Human spot-check	Solutions are correct 9/10 times; LaTeX renders cleanly.
Math grading	Accuracy	**92 %** on a held-out 1 k set.

Essay grading	Macro F1	**0.78** across six score bands.

6. What We Learned

- **Small can be mighty.** A 2-billion-parameter model—when pruned, quantized, and LoRA-patched—is plenty for K-12 tutoring.
- **Prompt rigidity beats prompt magic.** By *manually* concatenating the prompt (no fancy .apply_chat_template()), we kept the model from drifting into off-topic tangents.
- **Synthetic labels get you 80 % there.** Auto-graded datasets let us stand up viable classifiers in days, not weeks.

7. Roadmap

1. **Hand-written math input** – pair Gemma with a lightweight OCR so students can snap a photo of their scratch work.
2. **Rubric-based essays** – break the 0–5 score into sub-scores (logic, grammar, style).
3. **Pairwise ranking & RLHF** – teach the model *which* feedback is most helpful.
4. **One-click offline bundle** – export to GGUF for llama.cpp; aim for < 1 GB so a Raspberry Pi can host ThinkGrade at the edge.