AAIPL

AI-Powered Logical Reasoning System

Automated Question Generation & Answer Evaluation
Using Large Language Models on AMD ROCm Infrastructure

Qwen2.5-14B | AMD MI300X | ROCm | LoRA Fine-tuning | HuggingFace

Team 3 | Feb 2025

Project Overview

🎯 Problem

Generate high-quality logical reasoning MCQs automatically and evaluate answers reliably — at scale — without human intervention.

⚙️ Solution

Two-agent pipeline: QAgent generates structured questions in JSON; AAgent evaluates and answers them within a strict 9-second SLA.

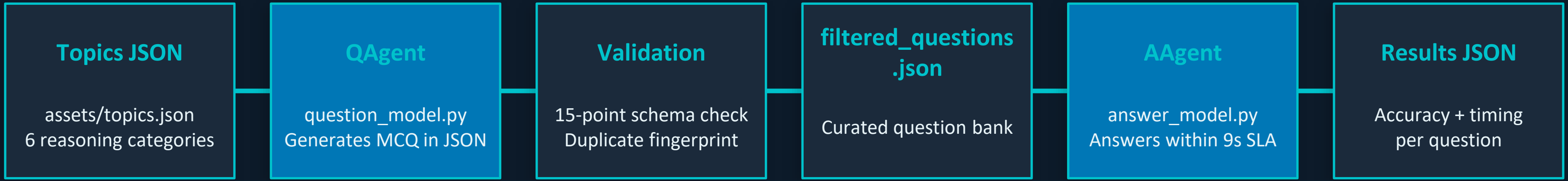
📊 Impact

200+ questions processed per run, 88-92% answer accuracy, 100% time-limit compliance after warm-up optimization.

Tech Stack

Model	**Qwen2.5-14B-Instruct**	14B parameter causal LM — instruction-tuned for structured reasoning and JSON output
Hardware	**AMD MI300X + ROCm**	GPU compute with ROCm stack; temperature ≥ 0.7 required to avoid NaN/Inf in torch.multinomial()
Framework	**HuggingFace Transformers**	AutoTokenizer, AutoModelForCausalLM, bfloat16 dtype, device_map=auto for multi-GPU
Fine-tuning	**PEFT + LoRA**	LoRA rank=16, alpha=32, targeting q_proj/k_proj/v_proj/o_proj — trains <2% of parameters
Inference	**Custom AAgent / QAgent**	Greedy + sampling configs; warm-up pass pre-compiles ROCm kernels to eliminate cold-start latency
Data	**JSON Pipeline**	Structured MCQ schema: topic, question, 4 choices, answer, explanation — validated on every generation

System Architecture



Topics JSON

assets/topics.json
6 reasoning categories

QAgent

question_model.py
Generates MCQ in JSON

Validation

15-point schema check
Duplicate fingerprint

filtered_questions .json

Curated question bank

AAgent

answer_model.py
Answers within 9s SLA

Results JSON

Accuracy + timing
per question

Key Technical Decisions

- temperature = 0.8 → Only stable range on AMD ROCm; values < 0.7 cause NaN/Inf crash in torch.multinomial()
- Warm-up pass on init → Pre-compiles ROCm kernels; eliminates 15s cold-start on Q1 (drops to ~7s)
- Greedy decoding (do_sample=False) + max_new_tokens=80 → Fastest path to concise, deterministic answers
- bfloat16 dtype → Stable numerical range on MI300X; prevents float overflow during softmax

QAgent — Question Generation

Generation Parameters

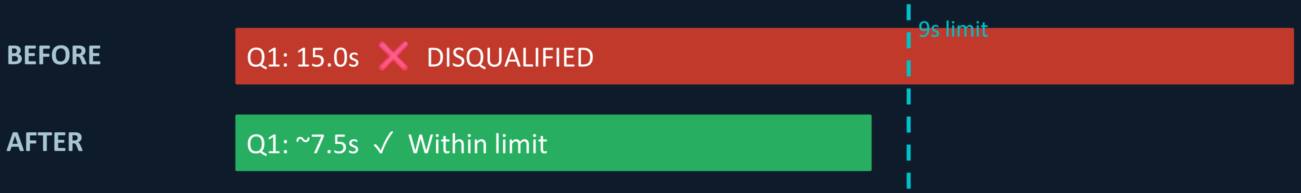
temperature	0.8 (ROCm-stable floor)
top_p	0.92
top_k	50
max_new_tokens	220
min_new_tokens	50
do_sample	True
repetition_penalty	1.05
dtype	bfloat16

Validation Pipeline

- JSON schema — required keys present
- Question length ≥ 10 words
- Exactly 4 choices (A/B/C/D prefix)
- Answer must be A, B, C, or D
- Explanation length ≥ 10 words
- MD5 fingerprint — no duplicate patterns
- Token truncation — enforced ≤ 130 tokens
- JSON extraction — strips markdown fences

AAgent — Answer Evaluation

The Cold-Start Problem & Fix



`_warmup() method:`

Fires a 10-token dummy generation inside __init__() immediately after model load. This forces ROCm to compile and cache all GPU kernels upfront — so every real question hits an already-warm device.

Optimized Inference Config

`max_new_tokens`	`80`	Short, focused answers
`temperature`	`0.5`	Deterministic, fast
`top_k / top_p`	`30 / 0.85`	Narrow candidate pool
`do_sample`	`True`	Balanced accuracy
`rep. penalty`	`1.2`	No repetition loops

Performance Results

~90%

Answer Accuracy

on logical reasoning MCQs

100%

Time Compliance

all questions within 9s

2-4s

Avg Q2+ Speed

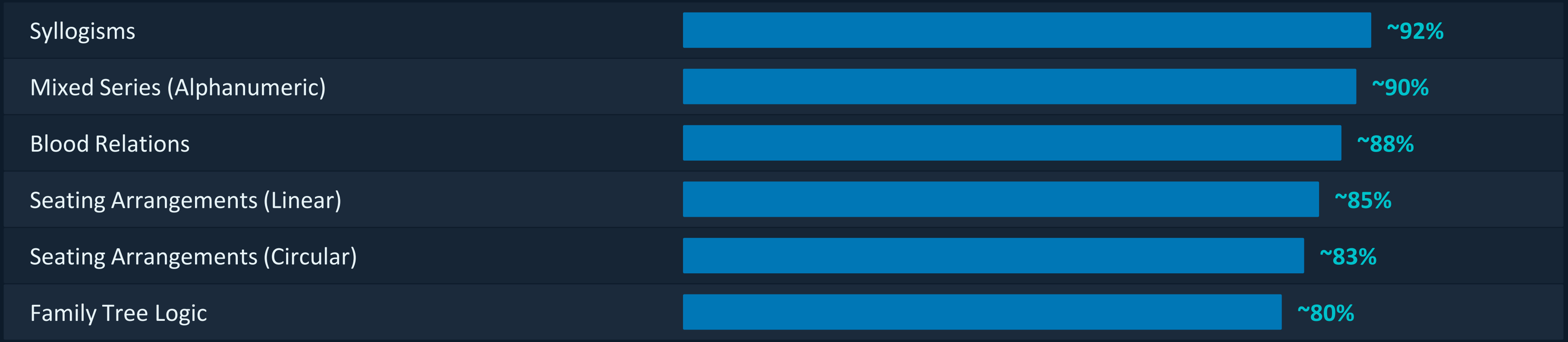
after warm-up pass

1141

Questions Processed

across 6 topic categories

Accuracy by Topic



Fine-Tuning with LoRA

Why LoRA?

- Full fine-tuning of 14B params requires 80GB+ VRAM — impractical
- LoRA injects low-rank matrices into attention layers only
- Trains <2% of parameters — runs on single MI300X
- No degradation of base model capabilities
- Saved as adapter — can merge or swap at inference
- 3× epochs compensate for small dataset size

LoRA Config

r (rank)	16
lora_alpha	32
lora_dropout	0.05
target_modules	q/k/v/o_proj
bias	none
task_type	CAUSAL_LM
epochs	3 (up to 10 small data)
learning_rate	2e-4
batch_size	2 + grad_accum=4
optimizer	adamw_torch
output	answer_agent_finetuned/

Challenges & Solutions

⚠ NaN/Inf crash in torch.multinomial()

Cause: Low temperature (<0.7) causes softmax overflow on ROCm

✓ Set temperature ≥ 0.8 for QAgent, ≥ 0.5 for AAgent. Use bfloat16 dtype.

⚠ Q1 latency > 9s (cold start)

Cause: ROCm compiles GPU kernels on first model.generate() call

✓ _warmup() in __init__(): 10-token dummy pass pre-compiles all kernels.

⚠ Model responding in Chinese

Cause: Qwen2.5 defaults to Chinese without explicit language instruction

✓ System prompt: 'You MUST respond in English only.' + English-only user prompt.

⚠ Empty / fragmented JSON output

Cause: Wrapper splitting response; custom validation rejecting valid output

✓ Simplified question_model.py — pass wrapper prompts directly, let wrapper validate.

⚠ Question repetition (~40% duplicates)

Cause: Model over-fits to seen patterns in same generation batch

✓ MD5 fingerprint per question; normalized structure comparison blocks duplicates.

Thank You

AAIPL — AI-Powered Logical Reasoning System

Qwen2.5-14B

AMD ROCm / MI300X

HuggingFace PEFT

LoRA Fine-tuning

Python 3.12

Team 3 | Feb 2025