MINESWEEPER RL AGENT

Gaming the Models

Team 69

AMD Hackathon 2026 | Track 2

CHALLENGING 20B MODELS WITH HIGH-STAKES LOGIC

The primary objective of this project was to evaluate whether a 20B parameter model, specifically `gpt-oss-20b-BF16`, could master pure logic and spatial reasoning within a high-stakes environment.

Minesweeper serves as an ideal testbed because it demands strict adherence to rules and precise logical deduction. A single incorrect move results in instant failure, forcing the model to prioritize safety over simple pattern matching.

FEATURE	REQUIREMENT
Logic	Deductive reasoning based on numerical clues
Safety	High penalty for incorrect moves (mines)
Format	Strict JSON output for game interaction

AUTONOMOUS MINESWEEPER AGENT VIA RL

Our agent operates autonomously by analyzing a 6×6 grid serialized into token-efficient JSON prompts. It deduces safe moves and flags predicted mines using a custom OpenAI Gym-like environment.

BOARD ANALYSIS

Parses the 6×6 JSON grid, identifying numerical clues and current flags to capture the spatial state.

LOGICAL DEDUCTION

Deduces safe moves and mine locations using multi-step reasoning patterns instead of random guessing.

GRPO LEARNING

Refines strategy through Group Relative Policy Optimization, learning from rewards and penalties.

LEVERAGING AMD MI300X FOR TRAINING

HARDWARE FOUNDATION

190GB

VRAM CAPACITY

High-throughput processing for the 20B parameter model on a single AMD MI300X GPU.

MI300X

COMPUTE ENGINE

Next-generation AMD Instinct™ accelerator optimized for large-scale AI workloads.

SOFTWARE ECOSYSTEM

ROCm 6.3

SOFTWARE STACK

Utilizing the latest ROCm platform for high-performance execution and stability.

20B

MODEL SCALE

Fine-tuning `gpt-oss-20b-BF16` to balance reasoning capability and parameter efficiency.

LOGIC-AWARE REWARD SYSTEM

A critical innovation in our approach was the implementation of a **12-point reward system** that distinguishes between "blind guessing" and "logical deduction," incentivizing the model to prioritize mathematically proven safety.

ACTION	REWARD / PENALTY	STRATEGIC GOAL
Logical Deduction	+20	Reward proven safety and reasoning
Correct Flag	+25	Encourage accurate threat identification
Safe Guess	+10	Baseline progress for exploration
Mine Hit	-50	Enforce extreme caution and safety

MANUAL LORA FOR ROCM STABILITY

THE ROCM CHALLENGE

Standard libraries like `peft` and `unsloth` encountered "illegal instruction" errors in their C++ extensions on the specific ROCm/PyTorch environment.

THE PURE PYTORCH SOLUTION

We bypassed external dependencies by developing a custom **Manual LoRA** adapter layer. This provided 100% stability and full control over weight management.

7.96M
TRAINABLE PARAMS

BF16
PRECISION

```python
class LoRALinear(nn.Module):
  def __init__(self, original_linear, r=16):
    super().__init__()
    self.lora_A = nn.Linear(in, r, bias=False)
    self.lora_B = nn.Linear(r, out, bias=False)
    self.scaling = alpha / r

  def forward(self, x):
    original_out = self.original_linear(x)
    lora_out = self.lora_B(self.lora_A(x))
    return original_out + lora_out * self.scaling
```

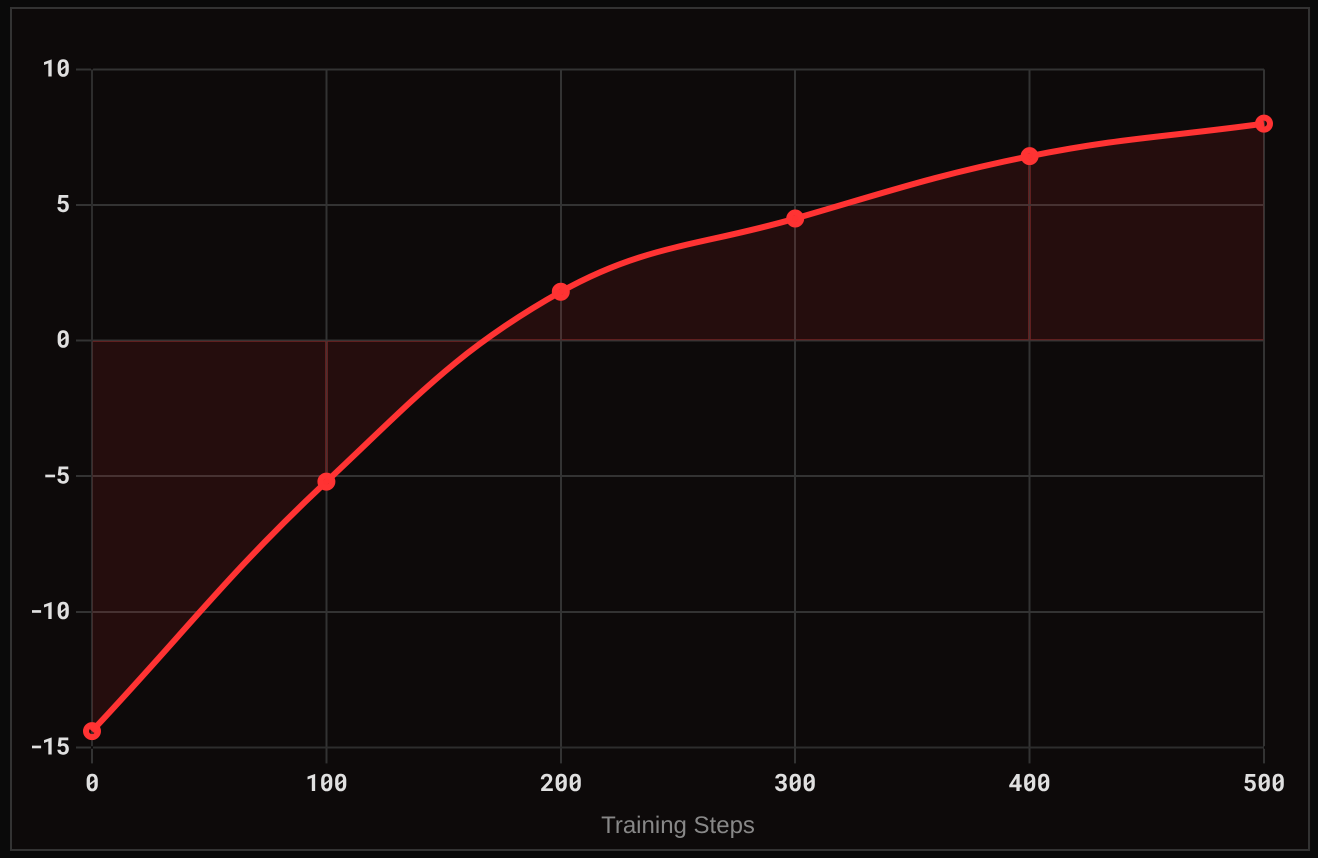
**Custom implementation manually managing matrix multiplications to ensure ROCm compatibility.*

CURRICULUM LEARNING STRATEGY

Our training strategy was divided into two distinct phases to maximize learning efficiency within a limited timeframe. This Curriculum Learning approach allowed the model to first master the "language" of the game before tackling complex strategic scenarios.

TRAINING PHASE	FOCUS AREA	STATE DISTRIBUTION
Phase 1	**Format Alignment**: Ensuring the model consistently produces valid JSON output and adheres to game rules.	100% Fresh Games
Phase 2	**Strategic Depth**: Teaching the model to solve complex boards using existing numerical clues and spatial patterns.	80% Mid-Game / 20% Fresh

PERFORMANCE METRICS & RESULTS



* Training progress over 500 steps on AMD MI300X

100%

JSON VALIDITY

Step	Mean Reward	Validity
0	-14.4	62%
250	+3.4	98%
500	+8.0	100%

The model successfully transitioned from random output to logic-based gameplay, learning to flag mines in simple patterns.

CONCLUSION & FUTURE WORK

KEY ACHIEVEMENTS

- Successfully trained a **20B model** to master logic-heavy tasks on AMD hardware.
- Implemented a **Logic-Aware Reward System** that steers LLMs toward deductive reasoning.
- Developed a custom **Manual LoRA** architecture for 100% stability in ROCm environments.

FUTURE DIRECTIONS

- → Scaling training steps to improve win rates and strategic depth.
- → Extending logic capabilities to larger grids (9×9, 16×16).
- → Exploring multi-agent collaborative Minesweeper logic.