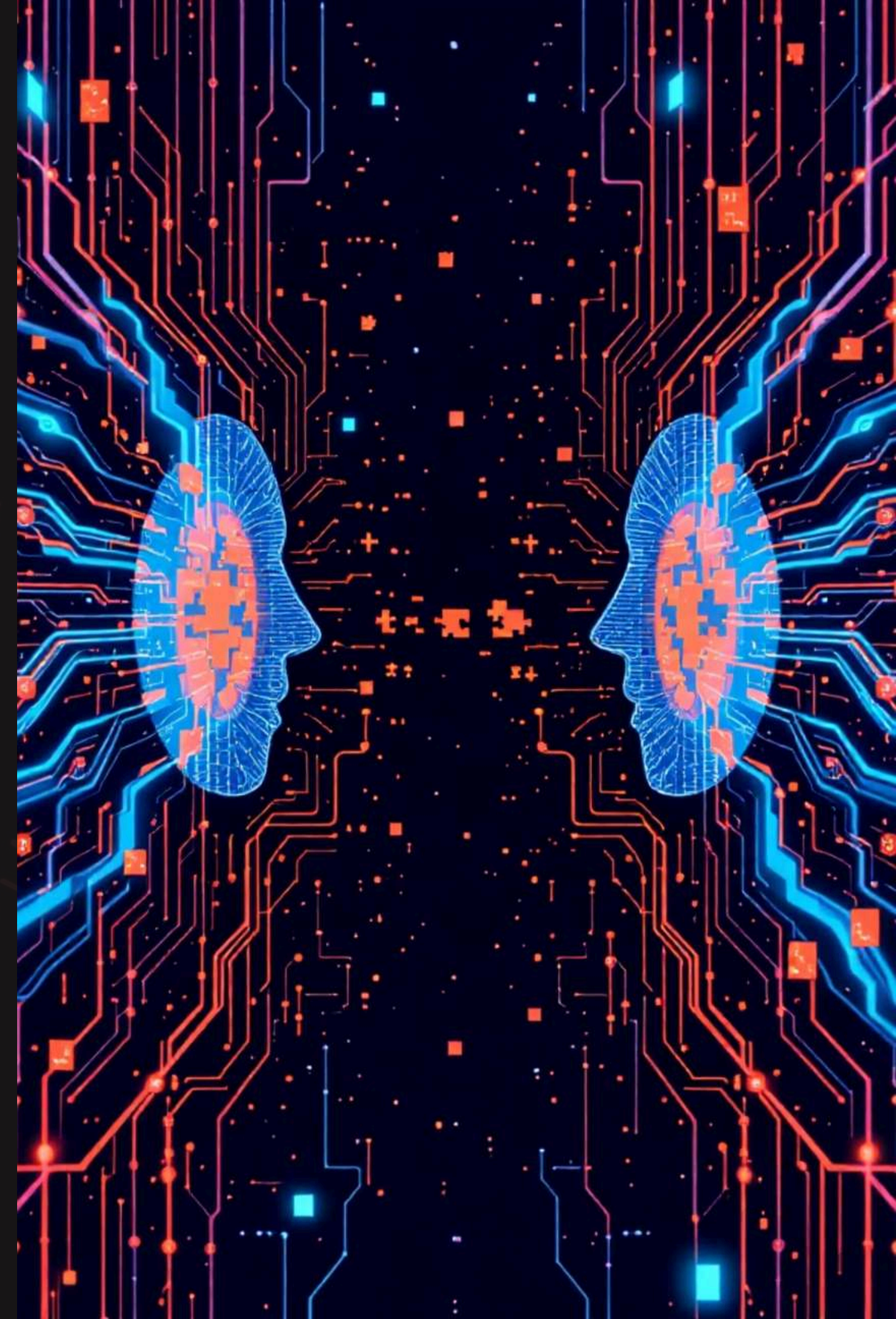


Engineering Competitive Logic Agents

A synthetic data generation and reverse-supervised fine-tuning approach for autonomous question generation and complex puzzle-solving in competitive logic tournaments.

Team Name: Insync
TRACK-1



Data Strategy: High-Fidelity Synthetic Generation

Traditional curated datasets proved insufficient for tournament-level complexity. We leveraged **gptoss-120b** to generate approximately 4,000 logic-dense samples spanning four critical domains, with roughly 1,000 samples per topic.



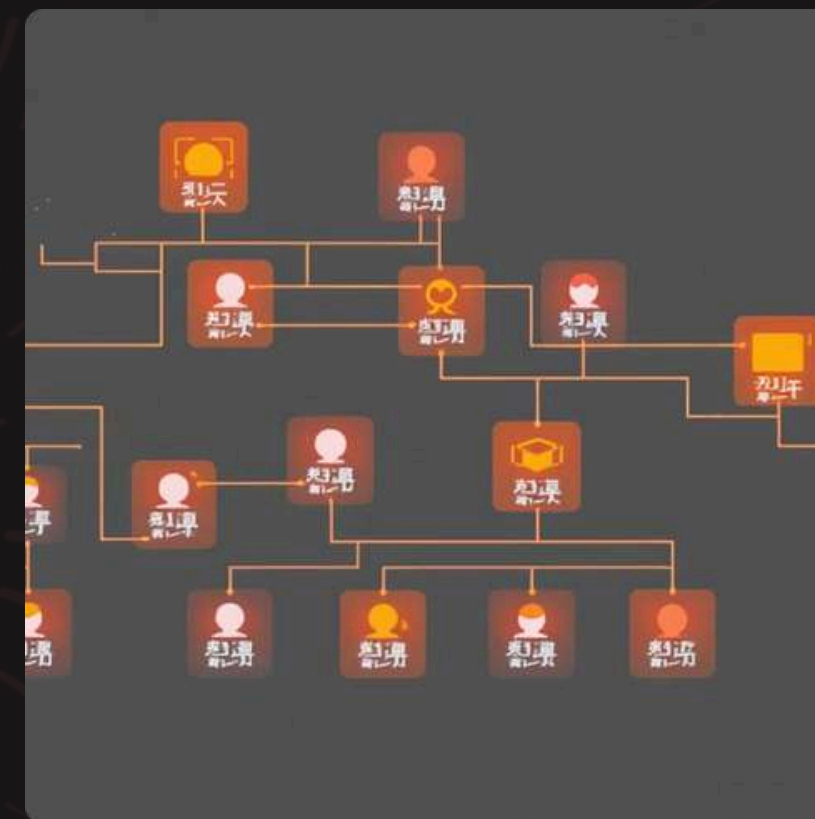
Syllogisms

Classical logical reasoning with categorical propositions and valid inference patterns



Seating Arrangements

Circular and linear ordering with positional constraints and relative relationships



Blood Relations

Complex kinship networks including multi-generational and in-law relationships



Alphanumeric Series

Mathematical patterns combining letters, numbers, and arithmetic operations

Why GPTOSS-120B? This model demonstrated superior performance in STEM and logical reasoning benchmarks compared to general-purpose chat models. Its specialized training on mathematical problem-solving yielded higher-quality samples that closely matched tournament requirements.



Training Architecture: Efficiency Through Specialization



LoRA Adaptation

Applied Low-Rank Adaptation to preserve base model knowledge while specializing for tournament JSON schema



AMD ROCm Stack

Optimized training infrastructure using ROCm for AMD MI300X hardware with memory-efficient operations



Unsloth Framework

Deployed memory-efficient fine-tuning enabling full training cycles in just 1 hour per agent

This architecture balanced the need for rapid iteration (**1-hour training cycles**) with production-grade inference performance. The LoRA approach allowed us to maintain the base model's broad linguistic capabilities while specializing for the tournament's precise JSON requirements and logical reasoning patterns.

The Q-Agent: Engineering "The Perfect Trap"

Maximizing Opponent Confusion Through Strategic Complexity

The Question Agent serves as the offensive component, designed specifically to generate puzzles that maximize the "Q-agent score"—the metric measuring incorrect opponent responses. This required careful calibration between solvability and deceptive complexity.

1 Domain-Specific Training

Fine-tuned exclusively on puzzle "Backstories" and "Conditions" to master tournament-specific framing

2 Symbolic Language Optimization

Prioritized single-letter identifiers (A, B, C) and direct condition phrasing to minimize token count

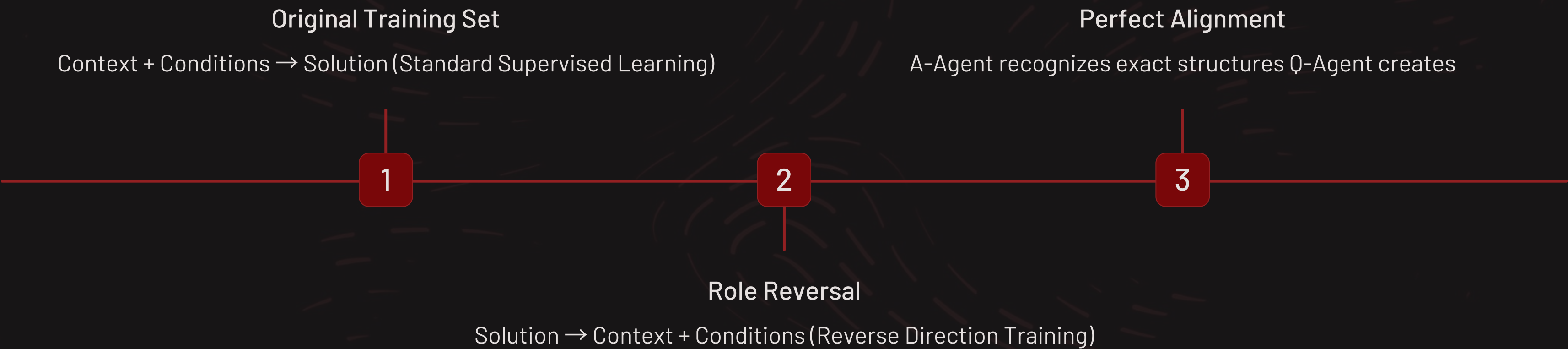
3 Complexity Engineering

Balanced multiple constraint layers to increase solving difficulty while staying within 130-token limit



The A-Agent: The "Reverse-Training" Innovation

Our most significant methodological breakthrough involved a strategic role reversal during training. Rather than treating question generation and solving as independent processes, we leveraged the same 4,000 logic samples to create symmetric training data.



This approach ensured the Answer Agent was specifically trained to decode the logical patterns generated by our own Q-Agent, rather than generic puzzle formats. The model learned to expect symbolic names (A, B, C), specific constraint phrasing patterns, and tournament-standard JSON structures.

Additionally, we forced explicit Chain-of-Thought (CoT) reasoning within the JSON output format, which proved critical for complex cases like 8-person circular seating arrangements and 4-generation family trees with multiple relationship types.

Results: Tournament-Ready Performance

Key Capabilities Validated

Family Tree Complexity

Successfully identified first-cousin-once-removed relationships across 4-generation trees with in-law connections

Multi-Variable Seating Arrangements

Solved circular arrangements with 8+ variables, handling relative positional constraints

Alphanumeric Series Precision

Generated and solved patterns with consistent mathematical increments, maintaining clean JSON output

Each capability was validated through tournament-simulation testing with 100+ questions per category. Agents demonstrated consistent performance within token and time constraints across all four puzzle types.

