Meet the DCAP Team

Driving innovation

- Ayush
- Daksh
- Deepans
- h Pallavi

Together, our team, DCAP, is dedicated to developing and refining the Fine-Tuned Question-Answer Agent System, ensuring its robustness and efficiency.

Fine-Tuned Question-Answer Agent System

Buildingan efficientandreliableAI forgenerating and solvingMultiple-ChoiceQuestions(MCQs)understrictconstraints. This report details the architecture, methodology, and key innovations of our system.

Project Overview: Q-Agent & A-Agent

Thisproject developed a sophisticatedQuestion-Answer(QA)agentsystemdesigned for specialized tasks:

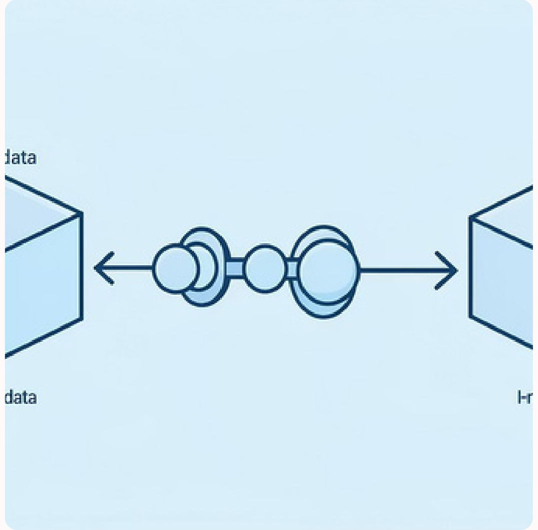
- **Q-Agent:** Generates high-quality multiple-choice questions.
- **A-Agent:** Accurately solves those generated questions.

The system operates under stringent formatting, topic, and latency constraints, leveraging a self-play synthetic data generation pipeline and Supervised Fine-Tuning (SFT) with LoRA adapters for optimal efficiency and performance.

Core Technologies: Foundation & Efficiency

Base Model: Qwen3-4B

Selected asthe foundational model for both Q-Agent (question generation) and A-Agent (question answering). Its robust reasoning capabilities make it ideal for logical and puzzle-based domains, ensuring complex problem structures can be effectively processed.



Fine-Tuning: Unsloth (LoRA)

Utilized for parameter-efficient fine-tuning, significantly reducing memory usage and training time. The LoRA configuration (Rank=64, lora_alpha=128, Dropout=0) targeted key modules to maintain model performance with minimal computational overhead.

Training Methodology: SFT & Configuration

Supervised Fine-Tuning (SFT)

Separate SFT was applied to optimize each agent:

- **Q-Agent:** Fine-tuned for precise MCQ generation.
- **A-Agent:** Fine-tuned for accurate MCQ solving.

Loss computation was strictly on assistant responses. The training configuration included:

- `bf16` precision
- `max_seq_length = 1024`
- `batch_size = 32`
- `gradient_accumulation = 2`
- `2 epochs`
- `learning_rate = 2e-4`

Data Engineering: Precision & Quality Control

Python Pipeline

Robust preprocessing scripts.

JSON Schema

Strictenforcementand auto-correction.

Deduplication

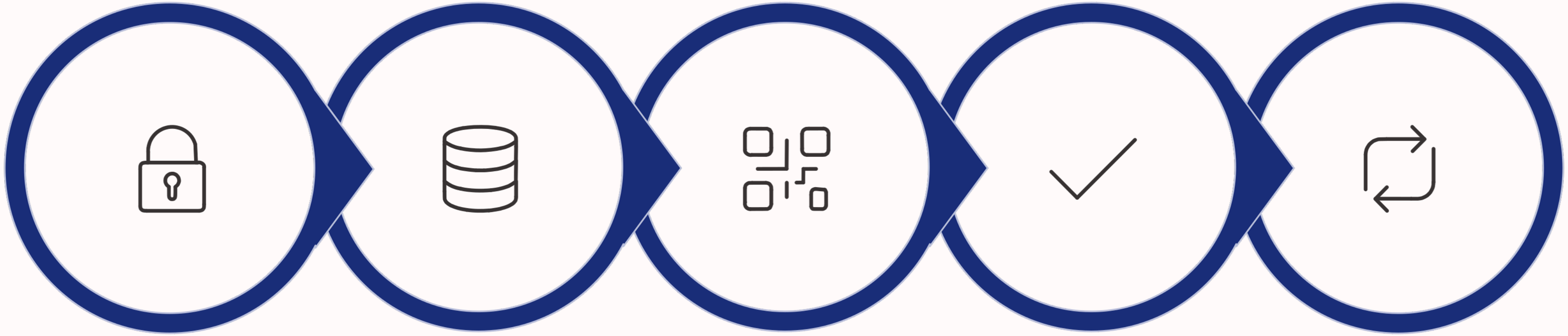
Jaccard-stylesimilarity for diversity.

Answer Balancing

Controlled A/B/C/D distribution.

Our comprehensive data engineering pipeline ensures high data quality, format integrity, and balanced answer distributions, critical for unbiased model training and optimal performance.

Work Flow: From Topic to Fine-Tuning



Topic Restriction

Synthetic Data

JSON Extraction

Validation

Deduplication

The project followed a meticulous multi-step process, beginning with strict topic adherence and culminating in specialized fine-tuning for both agents.

Adaptive Verification & Dataset Construction

Adaptive Answer Verification

To ensure accuracy in logic-heavy domains (Seating, Blood Relations, Series), the teacher model solved questions twice, with majority voting. Syllogism verification was disabled due to low reliability, optimizing compute.

Dataset Construction

Two specialized datasets were created:

- **A-Agent Dataset:** Input (Question + Choices), Output (JSON answer).
- **Q-Agent Dataset:** Input (Topic prompt), Output (JSON MCQ).

Both were saved in structured JSON format, ready for fine-tuning.

Fine-Tuning and End-to-End Testing

1

Separate LoRA Adapters

DedicatedLoRAadaptersweretrained for each agent, allowing for specialized optimization without retraining the entire base model. This modular approach enhances efficiency and performance.

2

Comprehensive Testing

Thesystemunderwentrigorousend-to-end testing, covering question generation, filter pass rates, answer prediction, and accuracy computation. Additional checks included conversation structure, JSON schema, and answer distribution validation.

Key Design Strengths

→ **Robust JSON Reliability Engineering**

Ensuring consistent and accurate JSON output.

→ **Strict Schema Validation**

Guardingagainstmalformedoutputs and data inconsistencies.

→ **Balanced Answer Distribution**

Preventingmodelbiastowardsspecificanswer choices.

→ **Deduplication for Generalization**

Enhancingdatasetdiversityandmodel'sability to generalize.

→ **Adaptive Correctness Verification**

Intelligently verifying answers for logical consistency.

→ **Parameter-Efficient Fine-Tuning**

Optimizingcomputationalresourceswithoutsacrificing performance.

→ **Latency-Aware Inference Setup**

Ensuring quick response times for real-time applications.

Conclusion: A Robust QA System

Thisprojectsuccessfullydevelopedacomplete QAsystempipeline,integratingsynthetic dataset generation, stringent data validation, parameter-efficient fine-tuning, and robust reliability engineering. The result is a high-performing Question-Answer agent system that meets strict evaluation constraints and sets a new standard for efficiency and accuracy.