

Lenovo AAITC Technical Assignments

Assignment 1: Advisory Engineer, AI Model Evaluation

Overview

This assignment assesses your ability to design comprehensive evaluation frameworks for foundation models, create model profiling and characterization tasks, and build a "model factory" concept that enables internal operations and B2B processes to leverage appropriate models for specific use cases and deployment scenarios.

Part A: Model Evaluation Framework Design (40%)

Task 1: Comprehensive Evaluation Pipeline

Design a complete evaluation pipeline for comparing three state-of-the-art foundation models (e.g., GPT-4, Claude 3, Llama 3) for Lenovo's internal operations.

Deliverables:

- 1. Evaluation Matrix** - Create a detailed evaluation framework including:
 - Performance metrics (BLEU, ROUGE, perplexity, F1-score, custom metrics)
 - Task-specific benchmarks (text generation, summarization, code generation, reasoning)
 - Robustness testing scenarios (adversarial inputs, edge cases, noise tolerance)
 - Bias detection and mitigation strategies
 - Safety and alignment assessments
- 2. Implementation Plan** - Provide Python pseudocode or actual code demonstrating:
 - Automated evaluation framework using PyTorch
 - Data processing pipeline with Pandas/NumPy
 - Statistical significance testing for model comparisons
 - Visualization of results using appropriate libraries
- 3. Production Monitoring Strategy** - Design a system for:
 - Real-time performance tracking in production
 - Model degradation detection
 - A/B testing framework for model updates
 - Alert mechanisms for performance anomalies

Task 2: Model Profiling and Characterization

Create a detailed profiling system for foundation models that captures:

Required Components:

1. Performance Profile

- Latency measurements across different input sizes
- Token generation speed
- Memory usage patterns
- Computational requirements (FLOPs, GPU utilization)

2. Capability Matrix

- Task-specific strengths/weaknesses
- Language/domain coverage
- Context window utilization efficiency
- Few-shot vs zero-shot performance comparison

3. Deployment Readiness Assessment

- Edge device compatibility
- Scalability considerations
- Cost-per-inference calculations
- Integration complexity scoring

Part B: Model Factory Architecture (30%)

Task 3: Model Selection Framework

Design a "Model Factory" system that automatically selects the appropriate model for specific use cases.

Requirements:

1. Use Case Taxonomy - Create a classification system for:

- Internal operations (HR, IT support, documentation)
- B2B processes (customer service, sales enablement, technical support)
- Deployment scenarios (cloud, edge, mobile)

2. Model Routing Logic - Develop an algorithm that:

- Matches use case requirements to model capabilities
- Considers performance vs. cost trade-offs

- Implements fallback mechanisms
- Handles multi-model ensemble scenarios

3. **Implementation Design** - Provide:

- System architecture diagram
- API specification for model selection service
- Example routing decisions with justifications

Part C: Practical Evaluation Exercise (30%)

Task 4: Hands-on Model Evaluation

Using publicly available models, conduct a comparative evaluation focused on a specific Lenovo use case.

Scenario: Evaluate models for internal technical documentation generation

Deliverables:

1. **Experimental Design**

- Dataset preparation strategy
- Evaluation metrics selection with justification
- Experimental protocol including controls

2. **Results Analysis**

- Quantitative performance comparison
- Error analysis with specific failure patterns
- Recommendations for model selection
- Improvement strategies for identified weaknesses

3. **Report Generation**

- Executive summary for stakeholders
- Technical deep-dive for engineering teams
- Visualization dashboard mockup

Evaluation Criteria

- Technical depth and accuracy (40%)
- Practical applicability to Lenovo's ecosystem (25%)
- Code quality and documentation (20%)
- Innovation and creative problem-solving (15%)

Assignment 2: Sr. Engineer, AI Architecture

Overview

This assignment evaluates your ability to architect end-to-end AI systems, manage the complete model lifecycle including post-training optimization, design production-ready AI platforms, and communicate complex technical concepts to diverse stakeholders.

Part A: System Architecture Design (35%)

Task 1: Hybrid AI Platform Architecture

Design a comprehensive AI platform architecture for Lenovo's hybrid-AI vision that spans mobile, edge, and cloud deployments.

Deliverables:

1. Architecture Blueprint

- Complete system architecture diagram with all components
- Data flow diagrams showing information movement
- Service mesh design for microservices communication
- API gateway and service discovery patterns

2. Technical Stack Selection

- Justify technology choices for each layer:
 - Infrastructure (Kubernetes, Docker, Terraform)
 - ML Frameworks (PyTorch, LangChain, LangGraph, AutoGen)
 - Vector Databases (Pinecone, Weaviate, Chroma)
 - Monitoring (Prometheus, Grafana, LangFuse)
- Integration patterns between components
- Scalability and fault-tolerance strategies

3. Cross-Platform Orchestration

- Design for seamless operation across:
 - Moto smartphones and wearables
 - ThinkPad laptops and PCs
 - Servers and cloud infrastructure
- Edge-cloud synchronization mechanisms

- Model deployment strategies per platform

Task 2: Model Lifecycle Management

Create a comprehensive MLOps pipeline for the entire model lifecycle.

Required Components:

1. Post-Training Optimization Pipeline

- Supervised Fine-Tuning (SFT) implementation strategy
- LoRA and QLoRA integration for parameter-efficient training
- Prompt tuning and optimization framework
- Model quantization and compression techniques

2. CI/CD for AI Models

- Version control strategy for models and datasets
- Automated testing pipeline for model updates
- Staging environments and progressive rollout
- Rollback mechanisms and safety checks

3. Observability and Monitoring

- Model performance tracking across deployments
- Drift detection and alerting systems
- Resource utilization monitoring
- Business metric correlation

Part B: Intelligent Agent System (30%)

Task 3: Agentic Computing Framework

Design an advanced agent system leveraging LLMs for complex task automation.

Deliverables:

1. Agent Architecture

- Intent understanding and classification system
- Task decomposition and planning algorithms
- Tool calling framework (using MCP - Model Context Protocol)
- Memory management and context retention

2. Implementation Design

- Detailed sequence diagrams for agent workflows
- State management and persistence strategies
- Error handling and recovery mechanisms
- Multi-agent collaboration patterns

3. Code Sample

- Provide working Python code demonstrating:
 - Basic agent implementation using LangGraph or AutoGen
 - Tool integration example
 - Reasoning chain visualization

Part C: Knowledge Management & RAG System (20%)

Task 4: Enterprise Knowledge Platform

Design a production-ready RAG system with advanced retrieval capabilities.

Requirements:

1. Knowledge Architecture

- Vector database design and embedding strategy
- Knowledge graph integration for structured data
- Hybrid search implementation (semantic + keyword)
- Reranking models and algorithms

2. Context Engineering

- External data integration patterns
- Context window optimization strategies
- Dynamic context selection based on query type
- Memory-efficient processing techniques

3. Quality Assurance

- Retrieval accuracy metrics and benchmarks
- Hallucination detection and mitigation
- Source attribution and citation system
- Feedback loop for continuous improvement

Part D: Stakeholder Communication (15%)

Task 5: Executive Presentation

Create presentation materials for different audiences demonstrating your architectural decisions.

Deliverables:

1. **Board-Level Presentation** (5 slides max)
 - Business value proposition
 - ROI projections and KPIs
 - Risk assessment and mitigation
 - Competitive advantage analysis
2. **Technical Documentation**
 - Comprehensive architecture decision records (ADRs)
 - API documentation with OpenAPI/Swagger specs
 - Deployment runbooks
 - Troubleshooting guides
3. **SME Collaboration Framework**
 - Guardrail design template for domain experts
 - Feedback collection and integration process
 - Knowledge transfer protocols
 - Training materials for non-technical stakeholders

Bonus Challenge: Innovation Showcase

Propose an innovative AI capability that leverages Lenovo's unique ecosystem advantage.

Suggestions:

- Cross-device AI orchestration system
- Federated learning across Lenovo devices
- Edge-cloud hybrid inference optimization
- Novel multimodal interaction paradigm

Evaluation Criteria

- Architectural sophistication and scalability (35%)

- Technical depth and implementation feasibility (30%)
 - Innovation and forward-thinking approach (20%)
 - Communication clarity and documentation quality (15%)
-

Submission Guidelines

Format Requirements

- All code should be production-quality with proper error handling
- Include README files with setup instructions
- Provide both technical and executive summaries
- Use appropriate visualization tools for complex concepts

Time Allocation Suggestions

- **Model Evaluation Assignment:** 6-8 hours
- **AI Architecture Assignment:** 8-10 hours

Assessment Focus Areas

For Model Evaluation Role:

- Deep understanding of evaluation metrics and methodologies
- Practical experience with model benchmarking
- Ability to identify and mitigate model weaknesses
- Strong analytical and experimental design skills

For AI Architecture Role:

- System-level thinking and design capabilities
- End-to-end ML lifecycle expertise
- Production deployment experience
- Stakeholder communication skills
- Innovation in applying AI to real-world problems

Additional Notes

- Feel free to make reasonable assumptions where details are not specified
- Document all assumptions clearly

- Focus on practical, implementable solutions
 - Consider Lenovo's specific ecosystem and business context
 - Demonstrate understanding of enterprise-scale challenges
-

Resources Referenced

The assignments incorporate concepts from:

- LLM Development Principles (8 Core Concepts)
- Context Engineering Framework
- Prompt Engineering Mastery Guide
- Advanced AI Architecture patterns
- Production MLOps best practices

These assignments are designed to thoroughly assess both theoretical knowledge and practical implementation skills required for success in the respective roles at Lenovo's AAITC.