Mastering AI and Large Language Model (LLM) Testing

Course Description

This course prepares learners to become experts in quality assurance and testing for AI and Large Language Model (LLM) applications. Starting from fundamental AI concepts, learners will master various test types, evaluation metrics, modern tools, and automation frameworks, culminating in the ability to design, implement, and maintain comprehensive AI/LLM testing pipelines.

Target Audience

- Beginners in software testing interested in AI quality assurance
- QA professionals transitioning to AI/LLM domains
- Developers and data scientists seeking test automation techniques
- AI product managers and tech leads focused on model reliability and safety

---

Learning Objectives

By the end of this course, learners will be able to:

- Explain key AI and LLM concepts and architectures
- Identify unique challenges in AI/LLM testing
- Perform functional, bias, safety, performance, and usability testing for AI systems
- Use popular automation frameworks like Playwright, DeepEval, RAGAs, LangSmith, OpenAI Evals, and Promptfoo
- Develop automated test suites with comprehensive evaluation metrics
- Integrate AI testing into CI/CD workflows for continuous quality assurance
- Evaluate AI model outputs for fairness, toxicity, and accuracy
- Build trustable, safe, and user-friendly AI applications

Course Modules

Module 0: Introduction to AI and LLMs

- Introduction to Python and installation setup
- Understanding variables, data types, and operators
- Control flow: conditional statements and loops
- Working with functions and reusable modules
- Lists, tuples, sets, and dictionaries in test data handling
- String manipulation and regular expressions
- File handling and JSON data parsing
- Exception handling, debugging, and logging best practices
- Virtual environments and dependency management using venv and pip
- Writing clean, modular, and maintainable Python scripts
- Mini hands-on exercises for automating small QA tasks

Module 1: Introduction to AI and LLMs

- What is Artificial Intelligence?
- Evolution and current trends in AI
- Foundations of Natural Language Processing (NLP)
- Introduction to Large Language Models (GPT, PaLM, LLaMA)
- Real-world applications and case studies of LLM-powered apps

---

Module 2: Unique Testing Challenges in AI/LLM

- Differences between traditional software and AI systems testing
- Probabilistic, non-deterministic outputs
- Risks: hallucinations, bias, toxicity, privacy concerns
- Ethical and legal implications
- Regulatory landscape and compliance basics

Module 3: Types of Testing in AI/LLM Applications

- Functional testing principles in AI contexts
- Bias and fairness testing: detecting and mitigating unwanted bias
- Safety and ethical testing: toxicity, refusal, harmful content prevention
- Performance and scalability testing for AI inference services
- Usability and accessibility testing for AI user interfaces

- Advanced types: explainability, regression, localization, logging/auditing, disaster recovery

---

Module 4: Evaluation Metrics for AI/LLM Outputs

- Understanding evaluation metrics: what to measure and why
- Core metrics: faithfulness, relevancy, completeness
- Fairness, bias scores, stereotype detection
- Safety metrics: toxicity, refusal rate
- Robustness and consistency
- Latency and scalability measures
- Sentiment, readability, coherence, and fluency
- Privacy and compliance metrics

Module 5: AI/LLM Testing Frameworks and Tools

- Overview of popular testing frameworks
- Playwright for frontend UI automation
- DeepEval for multi-metric LLM evaluation
- RAGAs for retrieval-augmented generation pipelines
- LangSmith for monitoring and evaluation dashboards
- OpenAI Evals for configurable end-to-end testing
- Promptfoo for prompt-output assertions

Module 6: Hands-On Setup and Test Automation

- Environment setup: Node.js, Python virtualenv, API keys
- Writing and running Playwright UI tests
- Implementing DeepEval Python test scripts
- Creating RAGAs evaluation pipelines
- Using LangSmith SDK and dashboards
- Building OpenAI Evals configs and running CLI tests
- Defining Promptfoo YAML test scenarios
- Automating test suites for continuous integration

Module 7: Designing Test Suites and Datasets

- Curating effective test inputs and output references
- Creating diverse prompt sets for bias and safety analysis
- Building benchmark datasets for regression and model updates
- Incorporating edge case and adversarial inputs
- Dataset versioning and maintenance best practices

Module 8: Integrating AI/LLM Testing into CI/CD Pipelines

- Overview of CI/CD concepts for AI apps
- Connecting tests to GitHub Actions, Jenkins, or other platforms
- Automating evaluations on model updates or frontend changes
- Monitoring test results and alerting mechanisms
- Managing flaky tests and false positives

Module 9: Fine-Tuning and Red Teaming in AI Systems

- Understanding model fine-tuning: purpose and benefits
- When to fine-tune vs. use pre-trained models
- Types of fine-tuning: instruction tuning, domain adaptation, RLHF
- Steps in fine-tuning: data preparation, labeling, and evaluation
- Testing fine-tuned models for performance, bias, and hallucination reduction
- Continuous evaluation and version tracking after fine-tuning
- Concept and importance of red teaming in AI testing
- Adversarial prompt testing and jailbreak scenarios
- Safety, compliance, and ethical vulnerability assessments
- Detecting bias, toxicity, and harmful content generation
- Simulating attacks for privacy and data leakage
- Red teaming tools and frameworks overview
- Integrating red team results into QA pipelines
- Real-world examples and lessons learned from red teaming exercises

---

Module 10: Case Studies and Industry Practices

- Analysis of real AI system failures and lessons learned
- Successful AI testing implementations
- Ethical AI and responsible innovation in testing context
- Future trends: multimodal AI, self-supervised evaluation

Module 11: Capstone Project and Assessment

- Learners design and implement a full testing pipeline
- Test automation covering UI, backend, and model outputs
- Evaluate a publicly available LLM or AI chatbot system
- Submit report and demo with lessons learned
- Peer review and instructor feedback