



Applied AI & LLM Engineer

Duration: 12 Weeks | **Total Hours:** 72, **Schedule:** 3 Days/Week | 2 Hours/Day

Learning Outcomes

By the end of this course, learners will:

- Build and deploy production-ready AI applications using FastAPI and modern backend frameworks.
- Train and deploy machine learning models to solve real-world business problems.
- Develop deep learning solutions for image, text, and sequence-based tasks.
- Create advanced NLP systems, including text classification and semantic search applications.
- Build intelligent AI assistants using Retrieval Augmented Generation and vector databases.
- Develop autonomous AI agents capable of performing complex multi-step workflows.
- Deliver complete, portfolio-ready AI projects suitable for jobs, freelancing, or product development.

PHASE 1

WEEK 1: FastAPI & AI Application Backends

Goal: Build AI-ready REST APIs that serve ML/DL/NLP models.

Day	Topic	Theory	Practical
1	FastAPI Foundations	FastAPI vs Flask vs Django; async/await; Pydantic schemas; Swagger UI; CORS middleware	Build typed REST API with 4 endpoints (GET, POST, PUT, DELETE) + Swagger documentation
2	Authentication & Middleware	API key auth, OAuth2, rate limiting, request logging, dependency injection, error handling	Secure API with key auth, rate limiter, structured error responses, test with Postman

3	Serving ML Models via FastAPI	Load scikit-learn / PyTorch models at startup; background tasks; async endpoints; deployment options (Railway/Render)	Wrap pretrained sentiment model into FastAPI; deploy live and share URL
---	-------------------------------	---	---

PHASE 2 - Classical Machine Learning

Goal: Train, evaluate, and deploy classical ML models as services. Communicate results in client-friendly language.

WEEK 2: Regression & Classification

Day	Topic	Theory	Practical
1	ML Landscape & Regression	Supervised vs unsupervised; ML vs LLM vs rule-based; bias-variance; Linear & Logistic Regression	Train price prediction (linear) and churn model (logistic) on real datasets
2	Tree-Based Models & Ensembles	Decision Trees, Random Forest, XGBoost, hyperparameter tuning, feature importance	Train all three models; compare performance; analyze important features
3	Model Evaluation & Client Reporting	Precision, Recall, F1, ROC-AUC, confusion matrix; cross-validation; model cards	Produce client-ready report for one-page evaluation — billable format



WEEK 3: Feature Engineering, NLP-Lite & ML as a Service

Day	Topic	Theory	Practical
1	Feature Engineering	Encoding, scaling, feature selection, SMOTE	Improve model accuracy by 10%+; document changes
2	Text Features & Classical NLP	Bag-of-Words, TF-IDF, n-grams, text classifier	Train spam/intent classifier; evaluate F1
3	ML API Deployment	Save/load models; async batch prediction; health check; versioning	Deploy best model as FastAPI API with /predict, /batch endpoints

PHASE 3 - Deep Learning

Goal: Build and deploy CNN and RNN/LSTM models for images and sequences.

WEEK 4: Neural Networks & CNNs

Day	Topic	Theory	Practical
1	NN Fundamentals	MLP, forward/backward pass, activation functions, loss functions, PyTorch tensors	Build 3-layer MLP on tabular data; visualize loss & weights
2	CNNs & Transfer Learning	Conv layers, pooling, ResNet/EfficientNet, transfer learning	Fine-tune ResNet-18 on 5-class dataset, >85% accuracy
3	Training Best Practices & CNN API	Dropout, batch norm, augmentation, checkpoints; serve via FastAPI	Deploy CNN as REST API with image upload and prediction

WEEK 5: RNNs, LSTMs & Sequence Modeling

Day	Topic	Theory	Practical
1	RNNs & Vanishing Gradients	Recurrent unrolling, vanishing/exploding gradients, GRU vs LSTM	Implement RNN and LSTM; compare stability on char prediction
2	LSTMs for Text & Time Series	Sentiment analysis, forecasting, teacher forcing	Train LSTM for text and time series; compare with Week 3 TF-IDF classifier
3	Sequence Model Deployment	FastAPI endpoints, preprocessing, bridge to Transformers	Deploy LSTM sentiment API; load-test 100 concurrent requests

PHASE 4

WEEK 6: Natural Language Processing

Goal: Build NLP APIs for preprocessing, embeddings, and classification.

Day	Topic	Theory	Practical
1	NLP Preprocessing	Tokenization, lemmatization, POS tagging, NER, reusable FastAPI pipeline	NLP preprocessing API: input raw text, output tokens + entities + POS
2	Sentence Embeddings & Semantic Similarity	Word2Vec intuition, Sentence Transformers, FAISS similarity search	Build semantic FAQ matcher API (top-3 answers)
3	Text Classification & NLP API Portfolio	Fine-tune sentence transformer; zero-shot LLM classifier	Deploy unified NLP API: /preprocess, /classify, /similarity

PHASE 5

WEEK 7: Vision & Multimodal AI

Goal: Build multimodal AI products using GPT-4o Vision and LLaVA.

Day	Topic	Theory	Practical
1	Vision LLMs	GPT-4o Vision API; image encoding; prompt design	Visual Q&A API for images/charts/forms
2	Document & Screenshot Intelligence	Parsing invoices, receipts, forms; prompt chaining; JSON output	Invoice parser API: structured output JSON
3	LLaVA & Local Vision Models	Multimodal chatbots; LLaVA via Ollama; local vs API considerations	Multimodal chatbot API: text + optional image, conversation history maintained

PHASE 6 - RAG Systems & Multi-Platform Deployment

Goal: Build enterprise-grade knowledge assistants.

WEEK 8: RAG Pipeline

Day	Topic	Theory	Practical
1	RAG Architecture & Chunking	Chunking strategies, embeddings, retriever	RAG over 60-page PDF; index, query, verify retrieval
2	Vector DBs & Hybrid Search	ChromaDB vs Pinecone; BM25 + embeddings; re-ranking	Index 20-document corpus; compare semantic vs hybrid retrieval
3	Conversational RAG & Evaluation	Query rewriting, RAGAs framework	Deploy conversational RAG chatbot; evaluation report

WEEK 9: Multi-Platform Deployment

Day	Topic	Theory	Practical
1	WhatsApp & Telegram	Twilio/Telegram bot APIs, session management	RAG chatbot live on WhatsApp & Telegram
2	Slack & MS Teams	Slack Bolt framework, Teams webhooks	RAG bot integrated in Slack workspace
3	REST API & Client Handoff	Versioning, pagination, cost monitoring	Finalize multi-platform RAG API; export Postman collection

PHASE 7

WEEK 10 & 11: Agentic AI & Autonomous Systems

Goal: Build autonomous AI agents with LangGraph.

Day	Topic	Theory	Practical
1	Prompt Engineering & LLM Ops	Few-shot, CoT, structured outputs, cost & latency	Cost-aware LLM router API
2	AI Agent Fundamentals	ReAct pattern, tool-calling, human-in-the-loop	Research agent with 3 tools, 5-step task, approval checkpoint
3	LangGraph & Multi-Agent Systems	State machines, multi-agent orchestration, workflow logging	Multi-agent pipeline: Researcher + Writer + Reviewer; deploy async FastAPI

PHASE 8

WEEK 12: Capstone, Freelancing & Security

Goal: Build portfolio-ready, client-deployable AI project. Learn freelancing essentials.

Day	Topic	Theory	Practical
1	AI Security & Responsible AI	Prompt injection, API key security, GDPR, hallucination disclosure	Security audit of two portfolio projects
2	Freelancing Business	Productized packages, pricing, proposals, scope creep	Create 3 productized service packages; submit proposals on Upwork/Fiverr
3	Capstone Demo Day	Technical Q&A, portfolio presentation	Submit GitHub repo, live deployment, architecture diagram, 3-min Loom pitch

Assessment & Certification

- Weekly quizzes and lab assessments
- Midterm practical evaluation
- Final capstone project
- Certificate of completion

Tools Covered

1. Python
2. SQL
3. Git
4. GitHub
5. Virtual Environments (venv, conda)
6. Pandas
7. NumPy
8. scikit-learn
9. PyTorch
10. OpenAI API
11. Hugging Face
12. LangChain



13. LangGraph
14. FAISS
15. Pinecone
16. ChromaDB
17. FastAPI
18. Docker
19. MLflow
20. REST APIs