AI-101: Intro to Large Language Models (LLMs) and LLM-based apps.

Course Length: 8 training hours

Course Description:

The fast pace of development in LLMs and related technologies, such as agentic AI, made it possible to use them even in enterprise grade applications. There are already a few areas where a new generation of LLM-based applications totally redefined applications' capabilities and users' expectations while AI technologies are going to radically change all kinds of other software areas as well.

That's why IT and other technical managers and professionals need to understand the technologies used in AI applications such as LLMs, RAG and agents.

Training objectives: At the end of the training participants:

- Recognize common LLM-based applications and understand their main building blocks.
- Get a high-level understanding of how modern large language models (LLMs) work and how they are trained in multiple steps.
- Explain the pros and cons of using LLMs via their APIs and via frameworks and be familiar with some popular open-source options.
- Understand the main ideas behind prompt engineering, including practical tips and best practices for working effectively with modern LLMs.
- Know the basics of RAG (Retrieval-Augmented Generation) systems, including their main parts, ways to improve their performance, and some new alternative solutions.
- Understand the motivations for and the two main types of LLM-based agentic systems (workflows and autonomous agents) as well as the key components and the way of working of autonomous agents.
- Recognize the importance and services of monitoring and evaluating LLM applications throughout their lifecycle as well as learning about some leading tools in this field (optional).

Main topics:

- Introduction to LLM based applications: current types, building blocks, challenges
- Why and how LLMs work and are trained?
- Using closed- and open-source LLMs via APIs and app. development frameworks
- Prompt engineering
- "Talk with your documents": Retrieval Augmented Generation (RAG)
- "AI that thinks and acts": LLM Agents
- Quality Assurance at LLM apps: Tracing and Evaluation (optional)

Besides gaining a basic understanding of Large Language Models (LLMs) and other technologies used in LLM-based applications, students will be able to examine their features and play with them during instructor's demonstrations and lab exercises.

This training is part of the AI portfolio of Component Soft which explores essential AI topics, such as:

- AI-101: Intro to GenAI with Large Language Model (LLMs) and LLM-based apps.
- AI-141: Using GitHub Kiro as coding assistant
- AI-161: Using Amazon Q as coding assistant
- AI-242: Using Github Copilot and spec-kit for agentic coding and spec-driven development (SDD)
- AI-262: Using Amazon Kiro for agentic coding and spec-driven development (SDD)
- AI-452: Agentic AI Application Development with LLMs

Structure: 50% lecture, 50% demonstration by the instructor, students can do hands-on lab exercises outside of training hours

Target audience: Technical managers and professionals who want to familiarize themselves with Large Language Models (LLMs) and LLM based applications.

Prerequisites: Basic understanding of IT concepts, User experience with ChatGPT or similar chatbots.

Detailed Course Outline

Module 1. Introduction to LLM based applications: current types, building blocks, challenges

- Main usage areas of LLM-based applications
- Main types of LLM-based applications
- Building blocks of LLM-based applications
- Demo: Popular LLM-based application types

Module 2. Why and how LLMs work and are trained?

- Main elements and operation of LLMs (tokenizer, embeddings, transformer, transformer head, next token selector)
- The 4+1 training phase of LLMs
- Most important LLM vendors and models
- Demo: Early and new LLM generations (GPT models before and after ChatGPT)

Module 3. Using closed- and open-source LLMs via APIs and frameworks

- Using LLMs through APIs
- Typical LLM parameters
- Using LLMs via Langchain
- Creating simple chatbot agents with Langchain
- Demo: Using a closed-source and an open-source LLM via API and the Langchain framework

Module 4. Prompt engineering

- What is prompt engineering?
- The 4 golden rules of prompt engineering

- Some important specific prompt engineering rules
- Demo: Demonstrating basic prompt techniques

Module 5. **"Talk with your documents": Retrieval Augmented Generation (RAG)**

- What is Retriever Augmented Generation (RAG)?
- How does RAG work?
- Main building blocks of an RAG pipeline
- Advanced RAG techniques
- **Demo**: Demonstration of Retrieval Augmented Generation (RAG) in an LLM app

Module 6. **"AI that thinks and acts": LLM Agents**

- Motivations for LLM-based Agentic Systems
- Main Features of and differences between Workflows and Agents
- Main Building Blocks: Functions, Tools, Agents
- The ReAct autonomous agent execution logic
- Multi-agent systems
- Demo: Agentic workflow and agent

Module 7. **Quality Assurance at LLM apps: Tracing and Evaluation (optional)**

- Why do we need them during development and operation?
- Tracing and evaluation tools for LLM-based apps
- Tracing basics
- Evaluation basics
- **Demo:** Langsmith Tracing and Evaluation