SZKOLENIE ŚREDNIO ZAAWANSOWANE

Budowa aplikacji AI - LLM, RAG i agenci

AI/AGENT-IT

Czas trwania: 2 dni (16h)

Szkolenie budowa aplikacji AI to intensywne warsztaty programistyczne, podczas których uczestnicy tworzą aplikacje wykorzystujące duże modele językowe, budują systemy RAG w Haystack, projektują agentów AI z użyciem Agno oraz uruchamiają lokalne modele za pomocą Ollama, łącząc teorię z praktycznym kodowaniem w Pythonie

Cele szkolenia

- Szkolenie przygotowuje do samodzielnego projektowania i implementowania aplikacji AI wykorzystujących duże modele językowe przez API
- Szkolenie uczy budowy systemów RAG opartych na wyszukiwaniu semantycznym, embeddingach oraz frameworku Haystack
- Szkolenie pokazuje, jak projektować i wdrażać agentów AI korzystających z narzędzi, pamięci, structured output oraz zewnętrznych systemów firmowych
- Szkolenie omawia zasady bezpiecznej integracji API, walidacji danych, testowania, ewaluacji oraz wdrażania aplikacji AI w środowisku produkcyjnym
- Szkolenie rozwija umiejętność uruchamiania, konfigurowania i wykorzystywania lokalnych modeli językowych za pomocą Ollama

Zalety

- Uczestnik projektuje i implementuje aplikacje AI korzystające z modeli językowych przez API oraz wykorzystujące structured output i function calling
- Uczestnik buduje systemy RAG z wykorzystaniem frameworku Haystack, embeddingów, wyszukiwania semantycznego i baz wektorowych
- Uczestnik projektuje i wdraża agentów AI korzystających z narzędzi, pamięci, zewnętrznych API oraz mechanizmów human-in-the-loop
- Uczestnik zabezpiecza integracje API przed nieautoryzowanym i błędnym działaniem agenta, wykorzystując walidację, kontrolę dostępu i mechanizmy ochrony
- Uczestnik uruchamia i konfiguruje lokalne modele językowe za pomocą Ollama oraz integruje je z aplikacjami AI
- Uczestnik ocenia jakość, koszty, bezpieczeństwo i wydajność wdrażanej aplikacji AI, wykorzystując ewaluację, monitoring i logowanie

Dla kogo?

- Programiści i inżynierowie oprogramowania budujący aplikacje wykorzystujące duże modele językowe
- Architekci rozwiązań projektujący integrację systemów AI z istniejącą infrastrukturą IT
- Specjaliści MLOps i DevOps odpowiedzialni za wdrażanie i utrzymanie aplikacji AI
- Programiści Python chcący zdobyć praktyczne umiejętności budowy systemów RAG i agentów AI

Wymagania

- Znajomość programowania w języku Python
- Podstawowa znajomość HTTP i REST API
- Umiejętność pracy w terminalu
- Podstawowa znajomość Dockera ułatwiająca realizację ćwiczeń wdrożeniowych

Program

1. Architektura współczesnych aplikacji AI
   a. Model językowy jako komponent większego systemu
   b. Tokeny, kontekst, inferencja i embeddingi
   c. Modele zamknięte oraz modele open source
   d. Modele tekstowe, multimodalne i embeddingowe
   e. Podstawowe wzorce architektoniczne aplikacji GenAI
2. Integracja modeli językowych przez API
   a. Komunikacja z modelami za pomocą REST API i bibliotek Python
   b. Role wiadomości: system, użytkownik i asystent oraz parametry generowania
   c. Streaming odpowiedzi oraz obsługa błędów, limitów i ponowień
   d. Structured output, function calling i walidacja danych
   e. Bezpieczne zarządzanie kluczami API u wielu dostawców
3. Systemy RAG i ich budowa w Haystack
   a. Ograniczenia wiedzy modelu i zasada działania RAG
   b. Chunking, przygotowanie dokumentów oraz bazy wektorowe
   c. Wyszukiwanie semantyczne, hybrydowe i reranking wyników
   d. Komponenty i pipeline'y w Haystack, Document Store i Retriever
   e. Łączenie retrievera z modelem oraz cytowanie źródeł
   f. Ocena jakości wyszukiwania i generowanych odpowiedzi
4. Architektura agentowa
   a. Różnice między chatbotem, workflow i agentem
   b. Model, instrukcje, narzędzia, pamięć i stan agenta
   c. Pętla podejmowania decyzji oraz mechanizm human-in-the-loop
   d. Granice autonomii agenta i kontrola nad jego działaniami
5. Budowa agentów z wykorzystaniem Agno
   a. Konfiguracja agenta, instrukcje, narzędzia i structured output
   b. Integracja agenta z funkcjami Pythona i zewnętrznymi API
   c. Pamięć, historia rozmowy oraz sesje użytkownika
   d. Zespoły agentów i przekazywanie zadań między nimi
   e. Obserwowalność wykonania agenta
6. Bezpieczna integracja narzędzi i API
   a. REST API oraz specyfikacja OpenAPI jako źródło narzędzi agenta
   b. Uwierzytelnianie i kontrola dostępu do wywoływanych operacji
   c. Walidacja argumentów i wyników oraz rozdzielenie odczytu od zapisu
   d. Idempotencja, ponowienia, limity czasu i circuit breaker
7. Lokalny hosting modeli z użyciem Ollama
   a. Sytuacje uzasadniające zastosowanie modelu lokalnego
   b. Wybór modelu, wymagania sprzętowe i kwantyzacja
   c. Uruchomienie modelu w Ollama z endpointem zgodnym z API OpenAI
   d. Zarządzanie modelami, wersjami oraz plikami Modelfile
   e. Podmiana zewnętrznego API na model lokalny bez przebudowy aplikacji
8. Ewaluacja, bezpieczeństwo i wdrażanie aplikacji AI
   a. Specyfika testowania systemów niedeterministycznych oraz evals dla RAG i agentów
   b. Logowanie promptów, odpowiedzi i wykonanych operacji
   c. Monitoring jakości, kosztów i opóźnień
   d. Ochrona przed prompt injection i separacja danych użytkowników
   e. Wdrażanie aplikacji w kontenerach Docker oraz mechanizmy fallback między modelem lokalnym a zewnętrznym API