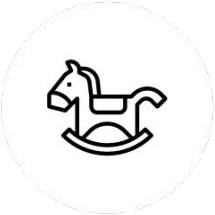
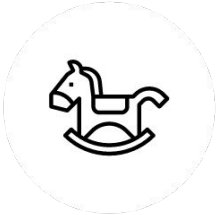


TEB

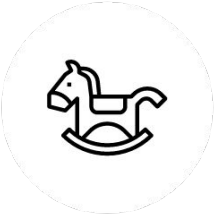
andrzey.jankowski@gmail.com

Anatomia dużych modeli językowych (LLM)



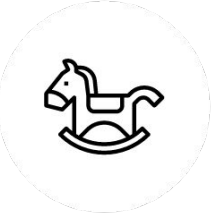
- Jak model „widzi” tekst? — tokenizacja i embedding.
- Base Model vs. Instruct Model — różnice i zastosowania.
- Jak powstają modele? — pre-trening oraz RLHF w praktyce.
- Ekonomia LLM — koszty, latency, throughput, batch vs. stream

Inżynieria interakcji (Prompt Engineering)



- Parametry generowania: temperature, top-p, top-k, max tokens – jak wpływają na kreatywność, precyzję i halucynacje.
- Wymuszenie struktury odpowiedzi (np. JSON) i projektowanie przewidywalnych interakcji.
- Scenariusze pracy: chat online, lokalny LLM, API.

Co LLM powinien wiedzieć? Kontekst, RAG i fine-tuning

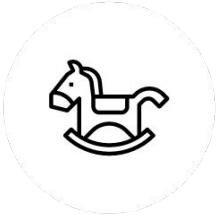


- Kontekst: jak dużo informacji model może przetworzyć?
- Skuteczne przekazywanie informacji w promcie.
- Retrieval-Augmented Generation (RAG) – łączenie LLM z zewnętrzną bazą wiedzy.
- Fine-tuning — kiedy warto i jakie daje efekty.

Bezpieczeństwo i ryzyka w pracy z LLM

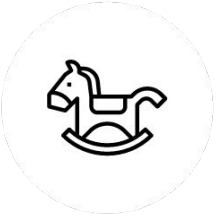
- Główne zagrożenia: halucynacje, prompt injection, przetwarzanie danych wrażliwych.
- Guardraile – jak budować bezpieczne i niezawodne aplikacje.

LLM przez API



- Struktura zapytań chat/completions, role i znaczenie system promptu.
- Limity, autoryzacja i najlepsze praktyki pracy z API.
- Function/Tool Calling – automatyzacja działań modelu.

Wprowadzenie do Agentów AI



- Czym jest Agentic AI i jak różni się od klasycznego chatbota?
- Cykl działania agenta: planowanie → wybór narzędzia → wykonanie → refleksja.
- Elementy agenta: planer, pamięć, narzędzia (API, bazy, pliki).
- Przykłady zastosowań agentów na komputerze użytkownika (np. Claude Code, Gemini CLI).