

Inżynieria Generatywnej AI – zakres materiału

Specjalizacja rozwija praktyczne kompetencje z obszaru sztucznej inteligencji, uczenia maszynowego, uczenia głębokiego i systemów generatywnych. Zakres przedstawia planowany kierunek specjalizacji i może zostać doprecyzowany przed rozpoczęciem zajęć lub w trakcie semestru. Szczegółowy zakres, poziom trudności i dobór projektów mogą zostać dostosowane do liczebności, przygotowania i postępów grupy, a program może ulec modyfikacjom wynikającym z rozwoju technologii oraz organizacji zajęć.

Nacisk położony jest na rozumienie metod, implementację w Pythonie oraz umiejętność budowania rozwiązań możliwych do pokazania, przetestowania i rozwijania w projekcie inżynierskim. Program ma charakter inżynierski i projektowy. Studenci będą pracować z narzędziami znanymi z praktyki AI/ML, takimi jak NumPy, PyTorch, TensorFlow/Keras, FastAPI, Docker, MLflow, CI/CD i Kubernetes. Celem specjalizacji jest nie tylko poznanie modeli, lecz także nauczenie się budowania działających, testowalnych i odpowiedzialnie wdrażanych systemów AI. Przygotowujemy Studentów do wymagań rynku IT.

UGP – Uczenie Głębokie w Pythonie

Uporządkowanie podstaw DL i przygotowanie do bardziej zaawansowanych projektów AI, w tym RAG, RL i systemów opartych na modelach głębokich.

Zakres	Temat	Najważniejsze elementy
1	Podstawy sieci neuronowych	Perceptron, wektory i macierze w NumPy, klasyfikacja, podstawowe metryki.
2	Sieci wielowarstwowe	MLP, propagacja w przód i wstecz, Softmax, TensorFlow/Keras i PyTorch.
3	Optymalizacja i regularyzacja	Funkcje straty, SGD/Adam, overfitting, callbacks, krzywe uczenia.
4	CNN i obrazy	Konwolucja, pooling, klasyfikacja MNIST/Fashion-MNIST/CIFAR, transfer learning.
5	Detekcja i modele czasowe	Detekcja obiektów, śledzenie, proste przetwarzanie wideo, optymalizacja pipeline'u.
6-7	Autoenkodery, VAE i GAN	Przestrzeń latentna, generowanie danych, stabilność uczenia, porównanie VAE i GAN.
8	RNN, LSTM i GRU	Sekwencje, problem zanikającego gradientu, predykcja szeregów czasowych, dane Stooq.
9-11	RL i Deep RL	Q-learning, MDP, eksploracja/eksploatacja, Deep Q-learning, przykłady z robotyki.
12-13	Attention, Transformery i RAG	Embeddingi, wyszukiwanie wektorowe, integracja LLM przez API, budowa prostego pipeline'u RAG.
14-15	Etyka, prawo ograniczenia AI	Prywatność, prawa autorskie, bias, ograniczenia modeli, koszt obliczeń, kierunki rozwoju.

Projekty: klasyfikacja obrazów CIFAR-100, implementacja RL/Deep RL oraz prosty system LLM+RAG z embeddingami i wyszukiwaniem wektorowym.

PRO1 – Projekt: inżynieria systemów AI i MLOps

Przejście od modelu ML/DL do działającej usługi, którą można uruchomić lokalnie, przetestować, udokumentować i zaprezentować.

Zakres	Temat	Najważniejsze elementy
1-3	API dla modelu	FastAPI, REST, Pydantic, endpoint /predict, walidacja wejścia/wyjścia, obsługa błędów, logi i metryki.
4-5	Konteneryzacja	Docker, Dockerfile, konfiguracja środowiska, Docker Compose, usługi pomocnicze, wolumeny i podstawy ruchu w API.
6-8	Dane preprocessing	Wersjonowanie danych, schematy, train/val/test, walidacja jakości, odporność na błędne lub uszkodzone wejście.
9-10	Cykl życia modelu	MLflow Tracking, artefakty, rejestr modeli, porównanie eksperymentów, podstawowe testy jakości i smoke tests.
11	CI/CD i monitoring	GitHub Actions, lint/test/build, podstawowa detekcja driftu i monitorowanie statystyk wejścia lub embeddingów.
12	Wydajność inferencji	Latency, throughput, cache, batching, warmup, podstawy kwantyzacji i pomiar kosztu działania usługi.
13	Kubernetes	Pod, Deployment, Service, ConfigMap, readiness/liveness probes, lokalne wdrożenie w kind/minikube.
14	RAG jako usługa	API + baza wektorowa, inicjalizacja indeksu, endpoint /ask, testy zapytań i ograniczenia systemu.
15	Finalizacja systemu	Swagger/OpenAPI, standaryzacja błędów, request-id, checklist wdrożeniowy, demo projektu.

Projekty: lokalny system AI, np. API dla modelu CIFAR albo RAG, z preprocessingiem, walidacją danych, Docker/Docker Compose, MLflow, metrykami, testami i dokumentacją.

GAI – Generatywne AI

Poznanie współczesnych systemów generatywnych i multimodalnych od strony architektury, implementacji, optymalizacji, ewaluacji i wdrożeń.

Zakres	Temat	Najważniejsze elementy
1	Vision Transformers	Patche obrazu, blok ViT, porównanie CNN vs ViT, wpływ rozmiaru patchy.
2	Modele dyfuzyjne	Dodawanie szumu, odszumianie, przykłady 1D, liczba kroków próbkowania vs jakość.
3-5	Multimodalność i CLIP	Embeddingi tekst-obraz, cross-attention, uczenie kontrastowe, wyszukiwanie obraz-tekst, wrażliwość na prompt.
6	GNN	Grafy, warstwa GCN, message passing, klasyfikacja węzłów, porównanie z MLP.
7	RAG	Baza wektorowa, retrieval, kontekst dla LLM, pomiar faktograficzności i analiza błędów.
8	Alignment preferencji	RLHF, DPO, dane preferencji, model nagrody, przeuczenie modelu nagrody.
9	Efektywne dostrajanie	LoRA, adaptory, zamrażanie wag, liczba parametrów, zużycie pamięci GPU.
10	Kompresja modeli	Dystylacja, kwantyzacja, pruning, opóźnienie inferencji, spadek jakości.
11-12	Skalowanie modeli	MoE, routing ekspertów, balansowanie obciążenia, trening rozproszony, batch size, przepustowość.
13-15	Wdrożenia, agenci i dane ustrukturyzowane	Guardrails, odporność, red-teaming, agenci narzędziowi, interpretowalność, strukturalne odpowiedzi i zastosowania w robotyce.

Projekty: multimodalne wyszukiwanie obraz-tekst w stylu CLIP oraz efektywny system LLM/RAG z LoRA, kwantyzacją i analizą kompromisu jakość–czas–pamięć.

PRO2 i PRO3 – zajęcia projektowo-prezentacyjne

PRO2. Zajęcia mają charakter wykładowo-prezentacyjny i służą uporządkowaniu szerszego kontekstu rozwoju generatywnej AI. Omawiane są m.in. prawa skalowania, koszt obliczeń, modele graniczne, modele multimodalne, AI w nauce, medycynie, robotyce, przemyśle kreatywnym i pracy zawodowej, a także problemy halucynacji, niezawodności, danych syntetycznych, bezpieczeństwa, alignmentu oraz możliwe kierunki po Transformerach. Celem jest umiejętność krytycznej oceny trendów, a nie tylko znajomość pojedynczego narzędzia.

PRO3. Zajęcia w semestrze dyplomowym wspierają przygotowanie pracy inżynierskiej i prezentacji końcowej. Zakres obejmuje strukturę pracy, terminy i formalności, LaTeX/Overleaf lub narzędzia równoważne, formatowanie tabel, rysunków, kodu i bibliografii, styl pisanie akademickiego, wymagania PJATK, kwestie plagiatu, konsultacje, przegląd fragmentów pracy, prezentacje ustne oraz próbne przygotowanie do obrony.