



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Przetwarzanie danych z wykorzystaniem GPU [S2Inf1-PB>PDGPU]

Przedmiot

Kierunek studiów
Informatyka

Rok/Semestr
1/2

Studia w zakresie (specjalność)
Przetwarzanie brzegowe

Profil studiów
ogólnoakademicki

Poziom studiów
drugiego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
20

Laboratorium
15

Inne (np. online)
0

Ćwiczenia
0

Projekty/seminaria
15

Liczba punktów ECTS

4,00

Koordynatorzy

mgr inż. Paweł Pietrzak
pawel.pietrzak@doctorate.put.poznan.pl

dr inż. Adam Turkot
adam.turkot@put.poznan.pl

Wykładowcy

Wymagania wstępne

Przed przystąpieniem do kursu studenci powinni być zaznajomieni z podstawami sieci neuronowych oraz systemów brzegowych. Oprócz tego studenci powinni umieć pisać programy w języku programowania Python z wykorzystaniem frameworka Tensorflow. Powinni również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

Cel przedmiotu

Celem kursu jest nauczenie studentów akceleracji obliczeń przy użyciu procesorów graficznych. Obliczenia, które mają zostać przyspieszone to wnioskowanie i trening sieci neuronowych. Dane, które mają być przetwarzane to obrazy i sekwencje wideo. Studenci zapoznają się z optymalizacjami dla sieci neuronowych opartymi na procesorach graficznych oraz ich implementacją, zaawansowanymi architekturami konwencjonalnych sieci neuronowych, przygotowaniem danych obrazowych oraz ugruntują swoją wiedzę na temat metodologii głębokiego uczenia. Kształtowanie u studentów umiejętności pracy zespołowej poprzez realizację elementów projektu i połączenie ich w całość.

Przedmiotowe efekty uczenia się

Wiedza

1. Ma zaawansowaną i pogłębioną wiedzę z zakresu szeroko rozumianych systemów informatycznych, podstaw teoretycznych ich budowania oraz metod, narzędzi i środowisk.
2. Ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną związaną z kluczowymi zagadnieniami z zakresu informatyki.
3. Zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich i prowadzeniu prac badawczych w wybranym obszarze informatyki.

Umiejętności

1. Potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku polskim i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie.
2. Potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne
3. Potrafi — przy formułowaniu i rozwiązywaniu zadań inżynierskich — integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne.
4. Potrafi dokonać krytycznej analizy istniejących rozwiązań technicznych oraz zaproponować ich ulepszenia (usprawnienia).
5. Potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania informatyczne, w tym zadania nietypowe oraz zadania zawierające komponent badawczy.
6. Potrafi — zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne — zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt — co najmniej w części — używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia.

Kompetencje społeczne

1. Rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe.
2. Rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca: (a) w przypadku wykładów: na podstawie odpowiedzi na pytania dotyczące materiału omawianego na poprzednich wykładach (b) dla laboratoriów/projektów: na podstawie oceny bieżącego postępu w realizacji zadań, Ocena podsumowująca: a) w zakresie wykładów weryfikacja założonych efektów kształcenia odbywa się poprzez test elektroniczny na platformie Moodle; b) w zakresie laboratoriów/projektów ustna weryfikacja wiedzy połączona z obroną projektu przygotowanego przez studentów w grupie

Treści programowe

Akceleracja z wykorzystaniem GPU, architektury konwolucyjnych sieci neuronowych, przygotowanie danych, implementacja modeli sieci neuronowych, optymalizacja modeli sieci neuronowych, framework Tensorflow, optymalizator modeli TensorRT, urządzenia brzegowe oparte na GPU, zastosowania głębokiego uczenia w dziedzinie widzenia komputerowego

Metody dydaktyczne

wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, prezentacje wybranych rozwiązań studenckich.
laboratoria/projekty: wykonywanie zadań i doświadczeń na sprzęcie i oprogramowaniu przewidzianym dla laboratorium, praca w grupach.

Literatura

Podstawowa 1. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Géron, Aurélien, O'Reilly UK Ltd., 2019

Uzupełniająca 1. Mastering Computer Vision with TensorFlow 2.x: Build advanced computer vision applications using machine learning and deep learning techniques, Krishnendu Kar, Packt Publishing, 2020

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	50	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	50	2,00