**KARTA OPISU PRZEDMIOTU - SYLABUS**

Nazwa przedmiotu  
Uczenie głębokie       
**Przedmiot**

Kierunek studiów  
Informatyka  
Studia w zakresie (specjalność)  
Sztuczna inteligencja  
Poziom studiów  
drugiego stopnia  
Forma studiów  
stacjonarne  
Rok/semestr  
1/2  
Profil studiów  
ogólnoakademicki  
Język oferowanego przedmiotu  
polski  
Wymagalność  
obligatoryjny

**Liczba godzin**

Wykład  
30       
Ćwiczenia  
       
Laboratoria  
30       
Projekty/seminaria  
       
Inne (np. online)  
     

**Liczba punktów ECTS**5     

**Wykładowcy**

Odpowiedzialny za przedmiot/wykładowca:  
prof. dr hab. inż. Krzysztof Krawiec  
email: krzysztof.krawiec@cs.put.poznan.pl  
tel. 61 665-3061   
Wydział Informatyki i Telekomunikacji   
ul. Piotrowo 2, 60-965 Poznań     Odpowiedzialny za przedmiot/wykładowca:  
mgr inż. Jakub Bednarek  
email: jakub.bednarek@put.poznan.pl  
tel. 61 665-3063   
Instytut Robotyki i Inteligencji Maszynowej, Zakład Robotyki  
ul. Piotrowo 3A, 60-965 Poznań

**Wymagania wstępne**  
Wiadomości i umiejętności wyniesione z pierwszego stopnia studiów technicznych.

**Cel przedmiotu**

1. Przekazanie studentom podstawowej wiedzy z wybranymi zagadnieniami uczenia głębokiego, sieci neuronowych, uczenia reprezentacji i powiązanych zagadnień uczenia maszynowego.
2. Rozwijanie u studentów umiejętności rozwiązywania problemów i zadań uczenia maszynowego z wykorzystaniem modeli uczenia głębokiego, w tym głębokich sieci neuronowych, zwłaszcza klasyfikacji, regresji, uczeniu reprezentacji i inżynierii cech.
3. Nabranie doświadczenia w zakresie wykorzystywania tych umiejętności w wybranych zastosowaniach praktycznych.
4. Kształtowanie u studentów umiejętności efektywnej pracy nad małymi przedsięwzięciami projektowo-programistycznymi w zakresie uczenia głębokiego, w tym współpracy w małych grupach projektowych.

**Przedmiotowe efekty uczenia się**Wiedza  
Student:

Ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie modeli uczenia głębokiego, w tym zwłaszcza sieci neuronowych [K2st\_W2].

Ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: uczenie głębokie, sieci neuronowe, uczenie maszynowe, uczenie reprezentacji, inżynieria cech [K2st\_W3].

Ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia systemów informatycznych sprzętowych lub programowych stosowanych do rozwiązywania wybranych zadań charakterystycznych dla uczenia głębokiego [K2st\_W5].

Zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich oraz przy prowadzeniu prac badawczych typowych dla klasyfikacji, regresji, inżynierii cech z wykorzystaniem uczenia głębokiego [K2st\_W6].

Umiejętności

Student:

Potrafi formułować i testować hipotezy związane z problemami rozwiązywanymi przy pomocy modeli uczenia głębokiego i prostymi problemami badawczymi charakterystycznymi dla tego obszaru sztucznej inteligencji [K2st\_U3].

Potrafi wykorzystać metody eksperymentalne do formułowania i rozwiązywania zadań w obszarze zastosowań uczenia głębokiego, zwłaszcza głębokich sieci neuronowych, oraz prostych problemów badawczych [K2st\_U4].

Potrafi - przy formułowaniu i rozwiązywaniu typowych dla uczenia głębokiego i sieci neuronowych - integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) [K2st\_U5].

Potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć dotyczących uczenia głębokiego oraz powiązanych zagadnień i technik sztucznej inteligencji [K2st\_U6].

Potrafi zaproponować ulepszenia (usprawnienia) istniejących rozwiązań technicznych w obszarze uczenia głębokiego, w szczególności w oparciu o istniejące biblioteki i środowiska programistyczne takie jak TensorFlow i PyTorch [K2st\_U8].

Potrafi - stosując m.in. koncepcyjnie nowe metody - rozwiązywać złożone zadania typowe dla uczenia głębokiego, w tym zadania nietypowe oraz zadania zawierające komponent badawczy, np. zaawansowane uczenie reprezentacji i/lub inżynierię cech [K2st\_U10].

Kompetencje społeczne  
Student:

Rozumie, że w obszarze uczenia głębokiego wiedza i umiejętności bardzo szybko stają się przestarzałe [K2st\_K1].

Rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu uczenia głębokiego, sieci neuronowych i powiązanych podejść uczenia maszynowego i sztucznej inteligencji w rozwiązywaniu problemów badawczych i praktycznych [K2st\_K2].

**Metody weryfikacji efektów uczenia się i kryteria oceny**Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:  
Ocena formująca:

a) w zakresie wykładów:

- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,

b) w zakresie laboratoriów / ćwiczeń:

- na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności wykazanych na egzaminie pisemnym składającym się z 5-8 pytań egzaminacyjnych powiązanych bezpośrednio z treścią wykładu. W przybliżeniu połowa pytań dotyczy zagadnień teoretycznych (zdefiniuj, opisz, scharakteryzuj, etc.), a druga połowa zadań obliczeniowych. Łączna liczba punktów to 25, do uzyskania oceny 3.0 wymagane jest osiągnięcie 13 punktów.

b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę postępu prac w kilku punktach kontrolnych w trakcie semestru, na podstawie projektu i jego dokumentacji, przygotowywanych częściowo w trakcie zajęć, a częściowo po ich zakończeniu; ocena ta obejmuje także umiejętność pracy w zespole,

- ocenę i obronę przez studenta finalnego sprawozdania z realizacji projektu, połączoną z prezentacją przed pozostałymi uczestnikami kursu.

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,

- uwagi związane z udoskonaleniem materiałów dydaktycznych,

- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

**Treści programowe**

Program **wykładu** obejmuje następujące zagadnienia:

Ogólnym celem jest poznanie przez studentów teoretycznych i praktycznych aspektów szeroko rozumianego uczenia głębokiego i sztucznych sieci neuronowych, a w szczególności:

1. Opanowanie podstaw głębokiego uczenia i sztucznych sieci neuronowych jako metod uczenia maszynowego i optymalizacji.

2. Poznanie algorytmów i metod uczenia modeli głębokich, ze szczególnym naciskiem na uczenie reprezentacji i inżynierię cech.

3. Nabycie umiejętności stosowania modeli uczenia głębokiego i sztucznych sieci neuronowych do rozwiązywania problemów klasyfikacji, detekcji, regresji, widzenia komputerowego, oraz interpretacji sekwencji i analizy szeregów czasowych.

Dla realizacji powyższych celów program wykładu obejmuje: Wprowadzenie. Definicja uczenia głębokiego jako specyficznego paradygmatu uczenia maszynowego, optymalizacji i modelowania. Definicja parametrów i hiperparametrów modeli. Omówienie modularnej charakterystyki modeli uczenia głębokiego. Opis najważniejszych i najczęściej wykorzystywanych komponentów uczenia głębokiego, w tym warstw gęstych, splotowych, agregujących, zwijających, redukujących, residualnych. Komponenty nieliniowe i normalizacyjne. Powiązanie z wybranymi pojęciami programowania funkcyjnego. Taksonomia funkcji straty i charakterystyka najczęściej wykorzystywanych funkcji straty. Uczenie przez hetero- i autoasocjację. Realizacja algorytmów uczenia głębokiego (autodiff). Architektury głębokie dla analizy struktur kombinatorycznych o zmiennym rozmiarze, zwłaszcza grafów. Modele głębokie dla uczenia nienadzorowanego, w szczególności dla analizy i konstrukcji skupień. Modele generatywne (GAN). Przegląd architektur stanowiących kamienie milowe w tym obszarze badań na bazie publikacji naukowych, ze szczególnym naciskiem na architektury splotowe i rekurencyjne, w tym: AlexNet, VGG, ResNet, GoogleLeNet, UNet, LSTM, GRU, GAN, PatchGAN, DCGAN, oraz wybrane architektury typu autoenkoder. Ilustracja efektów działania wybranych modeli.

Zajęcia **laboratoryjne** prowadzone są w formie piętnastu 2-godzinnych ćwiczeń, odbywających się w laboratorium, poprzedzonych 6-godzinną sesją instruktażową (trzy spotkania) na początku semestru. Ćwiczenia realizowane są przez 2-osobowe zespoły studentów.

Program laboratorium obejmuje następujące zagadnienia:

- Wprowadzenie (2h): Prezentacja założeń części laboratoryjnej przedmiotu.

- Prezentacja narzędzi informatycznych wykorzystywanych w części laboratoryjnej (biblioteki programistyczne, środowiska programistyczne).

- Sesja instruktażowa (6h): Ćwiczenia polegające na implementacji wybranych modeli sieci neuronowych w popularnych środowiskach (Python, Keras, Tensorflow, Pytorch).

- Testowanie zaimplementowanych algorytmów na danych rzeczywistych i sztucznych.

- Ocena poprawności i skuteczności algorytmów (w szczególności złożoność czasowa).

- Dobre praktyki projektowania i implementacji sieci neuronowych; typowe błędy i sposoby ich unikania.

- Realizacja projektów w grupach (22h): Realizacja, w grupach dwuosobowych, projektów programistycznych mających na celu realizację konkretnych zadań.

**Metody dydaktyczne**

**Wykład:** prezentacja multimedialna, ilustrowana przykładami podawanymi na tablicy, demonstracja.

**Ćwiczenia laboratoryjne**: rozwiązywanie zadań, projektowanie systemów indywidualnie i w małych grupach (typowo dwuosobowych), implementacja modeli sieci neuronowych, przeprowadzanie eksperymentów obliczeniowych, dyskusja, prezentacja wyników eksperymentów obliczeniowych i działania zaimplementowanych metod.

**Literatura**

**Podstawowa**  
1. Josh Patterson, Adam Gibson, *Deep learning : praktyczne wprowadzenie*. Grupa Wydawnicza Helion. 2018.

2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep learning: systemy uczące się*. Wydawnictwo Naukowe PWN, 2018.

3. Francois Chollet, *Deep Learning: Praca z językiem Python i biblioteką Keras*, Helion 2019 (oryginał: Deep Learning with Python).

**Uzupełniająca**  
1. Valentino Zocca, Gianmario Spacagna, *Deep learning: uczenie głębokie z językiem Python: sztuczna inteligencja i sieci neuronowe*, Grupa Wydawnicza Helion, 2018.

2. Krzysztof Krawiec, Jerzy Stefanowski. *Uczenie maszynowe i sieci neuronowe*. Politechnika Poznańska. Wydawnictwo, 2004.

**Bilans nakładu pracy przeciętnego studenta**

|  |  |  |
| --- | --- | --- |
|  | Godzin | ECTS |
| Łączny nakład pracy | 125 | 5 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem | 60 | 3 |
| Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych, przygotowanie do egzaminu, wykonanie projektu)[[1]](#footnote-1) | 65 | 2 |

1. niepotrzebne skreślić lub dopisać inne czynności [↑](#footnote-ref-1)