**Nazwa przedmiotu:**

Sztuczne sieci neuronowe

**Koordynator przedmiotu:**

Rajmund Kożuszek

**Status przedmiotu:**

Fakultatywny ograniczonego wyboru

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Sztuczna inteligencja

**Kod przedmiotu:**

SSNE

**Semestr nominalny:**

6 / rok ak. 2021/2022

**Liczba punktów ECTS:**

4

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

1. liczba godzin kontaktowych – 62 godz., w tym
− obecność na wykładach: 30 godz.,
− obecność na ćwiczeniach: 30 godz.,
− udział w konsultacjach związanych z realizacją przedmiotu: 2 godz.
2. praca własna studenta – 50 godz., w tym
− analiza literatury i materiałów wykładowych związana z przygotowaniem do kolejnych wykładów: 15 godz.
− dokończenie zadań rozpoczętych na ćwiczeniach: 20 godz.
− przygotowanie do kolokwiów: 15 godz.
Łączny nakład pracy studenta wynosi 112 godz., co odpowiada 4pkt. ECTS.

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

2,25 pkt. ECTS, co odpowiada 62 godz. kontaktowym

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

2 pkt. ECTS, co odpowiada 30 + 20 = 50 godz. realizacji ćwiczeń

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 30h |
| Ćwiczenia: | 30h |
| Laboratorium: | 0h |
| Projekt: | 0h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Wprowadzenie do sztucznej inteligencji (WSI),
Podstawy informatyki i programowania (PIP)

**Limit liczby studentów:**

60

**Cel przedmiotu:**

Przedmiot ten ma na celu zapoznanie studentów z sieciami neuronowymi, t.j. ich strukturami, metodami ich uczenia i potencjalnymi zastosowaniami. Przedstawia praktycznie wszystkie używane struktury sieci, w tym perceptron wielowarstwowy, sieci splotowe, generatywne, autokodery, sieci rekurencyjne i inne. Prezentuje algorytmy ich uczenia takie jak CM, NAG, Adam i inne. Przedstawia narzędzia programistyczne służące do implementowania sieci neuronowych, takie jak Tensorflow.
Ćwiczenia związane z przedmiotem mają na celu implementację poznanych metod oraz ich zastosowaniu do rozwiązywania praktycznych problemów. Dodatkowo na ćwiczeniach poruszane są aspekty techniczne oraz omawiane są dobre praktyki w uczeniu i zastosowaniu sieci neuronowych.

**Treści kształcenia:**

WYKŁADY:
1. Wprowadzenie (2 godz.)
Model perceptronu, sieci jednokierunkowe wielowarstwowe, algorytm wstecznej propagacji błędu. Podstawowe metody uczenia sieci: algorytm gradientu prostego, algorytm stochastycznego najszybszego spadku.
2. Wprowadzenie do Tensorflow (2 godz.)
Zaprezentowanie zasady działania biblioteki Tensoflow (TF). Graf obliczeń (dynamiczny/statyczny), automatyczne różniczkowanie, obliczenia na CPU/GPU/TPU. Modelowanie sieci, uczenie oraz ewaluacja w TF. Wizualizacja oraz diagnostyka sieci z wykorzystaniem Tensorboard. Porównanie TF z innymi dostępnymi rozwiązaniami (MXNET, PyTorch, Teano) oraz bibliotekami wyższego poziomu (na przykładzie biblioteki Keras/API TF 2.0).
3. Algorytmy uczenia sieci (3 godz.)
Podejścia do estymacji gradinetu; minipakiety. Algorytmy wykorzystujące inercję: CM, NAG. Algorytmy adaptacyjne: AdaGrad, RMSprop, Adam.
4. Architektury sieci jednokierunkowych (6 godz.)
Sieci splotowe (CNN), autokodery (AE), sieci kapsułkowe (CapsNet) i algorytm uczenia dynamic routing.
5. Sieci Generacyjne (2 godz.)
Model sieci dla ograniczonych maszyn Bolzmanna (RBMs) oraz głębokich sieci przekonań (DBN). Algorytm wake-sleep. Generatywne sieci przeciwstawne (GAN), autokodery wariacyjne (VAE).
6. Sieci rekurencyjne (4 godz.)
Sieć neuronowa jako maszyna stanowa. Sieci Jordana i Elmana. Algorytm wstecznej propagacji przez czas (BPTT). Algorytm rekurencyjnego uczenia w czasie rzeczywistym (RTRL). Problem zanikania i eksplozji gradientu. Sieci LSTM i GRU. Zastosowanie sieci rekurencyjnych w przetwarzaniu języka naturalnego (NLP).
7. Dobre praktyki oraz podejścia pomocnicze przy uczeniu sieci neuronowych (4 godz.)
Trenowanie wstępne sieci. Zagnieżdżenie (embedding): word2vec (skip-gram, CBOW), GloVe. Przeciwdziałanie przeuczeniu: opuszczanie (drop-out), wczesne zatrzymanie. Wspomaganie uczenia: normalizacja wsadów (batch normalization), odpowiednia inicjalizacja wag. Ataki na sieci neuronowe i metody obrony przed nimi.
8. Architektury specjalizowane (2 godz.)
Sieci Hopfielda, Kohonena, model Wide&Deep.
9. Tensorflow - zagadnienia zaawansowane (2 godz)
Rozproszenie obliczeń, przygotowanie danych z wykorzystaniem tf.data, wdrożenie produkcyjne TF (TF Serving), Projekt TF Hub.
10. Sieci neuronowe w uczeniu z wzmocnieniem (1 godz)
Algorytmy Deep Q-Learning i REINFORCE epizodyczny z użyciem sieci neuronowych.
11. Sprawdzenie efektów uczenia się (2 godz.)
Pierwsze kolokwium po zrealizowaniu piątego tematu (1 godz.). Drugie kolokwium kończące przedmiot (1 godz.).
 
ĆWICZENIA:
1. Organizacja ćwiczeń. Zapoznanie z środowiskiem Jupyter Notebook. Wprowadzenie do Tensorflow oraz podstawowych obiektów: graf obliczeń, sesja oraz operacje. Automatyczne różniczkowanie w grafie obliczeń. “Hello world” w TF - implementacja regresji logistycznej.
2. Budowanie modelu sieci neuronowej dla zadania regresji oraz klasyfikacji. Zapisywanie modelu wraz z wagami do pliku. Odczytanie zapisanego modelu oraz ewaluacja wyników na zbiorze testowym.
3. Klasyfikacja na zbiorze MNIST - z sieciami jednokierunkowymi a następnie z warstwami wykorzystującymi operację splotu (CNN). Porównanie wyników dla obu sieci.
4. Uczenie modelu z wykorzystaniem różnych metod dostępnych w TF. Dostrajanie hyper-parametrów oraz diagnostyka uczenia modelu z wykorzystaniem Tensorboard. Implementacja metody wczesnego stopu w uczeniu sieci. Różne architektury sieci CNN. Zapoznanie z warstwami max-pooling, sieci rezydualne. Wykorzystanie pre-trenowanych sieci (VGG, ResNet) do detekcji obrazów ze zbioru TinyImageNet-200. Dostrajanie parametrów już wytrenowanej sieci.
5. Sieci Kapsułkowe - algorytm dynamicznego routingu. Utworzenie klasyfikatora dla zbioru MNIST.
6. Autokodery oraz VAE: kompresja obrazów MNIST (AE) oraz ich generacja (VAE). Implementacja prostej sieci GAN
7. Przewidywanie szeregów czasowych z sieciami neuronowymi - RNN, CNN.
8. Word embedding - implementacja podejść CBOW oraz skip-gram. Podejście globalnych wektorów (GloVe) do zagnieżdżenia. Wykorzystanie istniejących zagnieżdżeń. Porównanie metod i wizualizacja w Tensorboard.
9. Sieci rekurencyjne w NLP: wykorzystanie podejścia bag-of-words oraz sieci jednokierunkowej vs sieci rekurencyjnej. Utworzenie klasyfikatora filtru spamowego.
Uczenie sekwencja-sekwencja: zastosowanie rekurencyjnych sieci neuronowych do automatycznej translacji tekstów.
10. Implementacja modeli ograniczonych maszyny Bolzmanna (RBM) oraz głębokie sieci przekonań (DBN) z algorytmem wake-sleep.
11. Sieci Neuronowe w uczeniu z wzmocnieniem. Implementacja wybranych metod z wykorzystaniem środowiska OpenGym oraz TF.
12. Rozpraszanie obliczeń przy implementacji sieci neuronowych.

**Metody oceny:**

Realizacja przedmiotu obejmuje następujące formy zajęć:
- wykład prowadzony w wymiarze 2 godz. tygodniowo; w wybranych zagadnieniach przewidziana jest aktywizacja studentów na wykładzie,
- ćwiczenia przy komputerach w wymiarze 2 godz. tygodniowo; w ramach tych zajęć student, korzystając z oprogramowania i sprzętu komputerowego, będąc pod opieką prowadzącego zajęcia, będzie realizował wskazane ćwiczenia dotyczące modelowania/uczenia/ewaluacji sieci neuronowych z wykorzystaniem biblioteki Tensorflow na CPU/GPU; studenci podzieleni na grupy zadaniowe będą zobowiązani do wspólnego rozwiązania zadań.
Sprawdzanie założonych efektów kształcenia realizowane jest przez:
− ocenę wiedzy i umiejętności związanych z realizacją zadań przydzielanych na ćwiczeniach – ocena z realizacji zadań;
− ocenę wiedzy wykazanej na dwóch kolokwiach pisemnych.

**Egzamin:**

nie

**Literatura:**

● Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning,” WNT, 2015.
● Dokumentacja biblioteki Tensorflow (https://www.tensorflow.org).
● Geoffrey E. Hinton, 2012 Coursera course lectures: Neural Networks for Machine Learning (http://www.cs.toronto.edu/~hinton/coursera\_lectures.html).
● Wybrane publikacje (dostępne na platformie arXiv.org).

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103A-INSZI-ISP-SSNE

**Uwagi:**

(-)

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka W01:**

ma wiedzę dotyczącą matematycznych podstaw działania sieci neuronowych

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** W01, W06

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o

**Charakterystyka W02:**

ma wiedzę dotyczącą struktur sieci neuronowych dostosowanych do odpowiednich klas zastosowań

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** W06

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o

**Charakterystyka W03:**

ma wiedzę dotyczącą algorytmów uczenia odpowiednich sieci neuronowych

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** W06

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o

**Charakterystyka W04:**

ma wiedzę na temat narzędzi programistycznych służących do implementacji sieci neuronowych i posługiwania się nimi

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** W06, W08

**Powiązane charakterystyki obszarowe:** P6U\_W, I.P6S\_WG.o, III.P6S\_WG

### Profil ogólnoakademicki - umiejętności

**Charakterystyka U01:**

potrafi wskazać strukturę sieci neuronowej i algorytm jej uczenia odpowiednie dla danego problemu modelowania, regresji, klasyfikacji i in.

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** U01

**Powiązane charakterystyki obszarowe:** P6U\_U, I.P6S\_UW.o, III.P6S\_UW.o

**Charakterystyka U02:**

potrafi zaimplementować różnorodne struktury sieci neuronowych i algorytmy ich uczenia stosując odpowiednie do tego narzędzia programistyczne

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** U07

**Powiązane charakterystyki obszarowe:** I.P6S\_UW.o, III.P6S\_UW.o, P6U\_U

**Charakterystyka U03:**

potrafi stworzyć model neuronowy odpowiedni dla postawionego problemu i istniejących danych

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** U03, U04

**Powiązane charakterystyki obszarowe:** P6U\_U, I.P6S\_UW.o, III.P6S\_UW.o

### Profil ogólnoakademicki - kompetencje społeczne

**Charakterystyka K01:**

jest gotów stale aktualizować i wzbogacać posiadaną wiedzę

Weryfikacja:

kolokwium, ćwiczenia

**Powiązane charakterystyki kierunkowe:** K01

**Powiązane charakterystyki obszarowe:** P6U\_K, I.P6S\_KK