Inteligență Artificială

Lucrare de laborator – Varianta 2

9 aprilie 2022

În această lucrare veți antrena modele pentru a prezice clasa de care aparține un text.

În directorul **curent**, veți găsi datele de antrenare (train_data.npy), etichetele corespunzătoare (train_labels.npy) și datele de testare (test_data.npy). În fișierul train_data.npy se găsește o listă care conține 1000 *string*-uri care reprezintă textele de antrenare. Un text este reprezentat printr-un *string*. În fișierul test_data.npy se găsește o listă care conține 323 texte de testare.

Rezolvați următoarele cerințe:

1. **(2p)** Implementați o metodă care să returneze similaritatea string kernel dintre două șiruri de caractere pe baza biților de prezență a n-gramelor comune de lungime 8. De exemplu, fiind date două string-uri s = „ananas copt” și t = “banana verde” și lungimea n-gramelor p = 4, funcție va returna valoarea 2 (șirurile au în comun 4-gramele „anan” și „nana”).

2. **(2p)** Implementați metoda celor mai apropiați vecini folosind funcția de similaritate de la punctul 1. Pentru a obține punctajul acordat, trebuie să implementați corect modelul și să generați o submisie / fișier cu predicțiile pe datele de test.

2p – acuratețe minimă pe datele de test = 89%

3. **(1p)** Implementați o metodă care să returneze matricea kernel dintre două mulțimi de exemple, folosind funcția de similaritate implementată la punctul 1. Apelați metoda pentru a calcula matricile kernel pentru antrenare și testare. De exemplu, dacă mulțimea de antrenare are 1000 de exemple și mulțimea de testare are 300 de exemple, atunci matricea kernel pentru antrenare va avea 1000x1000 componente, iar matricea kernel pentru testare va avea 300x1000 componente.

4. **(3p)** Antrenați un model KRR pe mulțimea de antrenare, folosind matricea kernel precalculată de la punctul 3, specificând opțiunea kernel=“precomputed” (*). Pentru a obține punctajul maxim, trebuie să găsiți parametrii optimi pentru modelul dat și să generați maxim 3 submisii / fișiere cu predicțiile pe datele de test. În lipsa etichetelor de test, puteți păstra o parte din mulțimea de antrenare pentru validare.

1p – acuratețe minimă pe datele de test = 90%
2p – acuratețe minimă pe datele de test = 92%
3p – acuratețe minimă pe datele de test = 93%

(*) în lipsa utilizării opțiunii kernel=“precomputed”, punctajul maxim pentru punctul 4 este **1p**.

5. **(1p)** Creați un raport al experimentelor însoțit de evaluarea pe un set de validare a diferite combinații de hiperparametri, atât pentru modelul de la punctul 2 cât și pentru cel de la punctul 4. Raportul poate conține tabele sau grafice.

1p - Oficiu

Observații importante:

După implementarea cerințelor de mai sus, trebuie să trimiteți într-un folder denumit {Nume}_{Prenume}_{Grupa}_{Varianta}:

a) Cel mult 1 submisie pentru setul de testare cu metoda de la punctul 2 și cel mult 3 submisii pentru setul de testare cu metoda de la punctul 4. O submisie constă într-un fișier txt denumit {Nume}_{Prenume}_{Grupa}_subiect{i}_solutia_{j}.txt, unde i este numărul subiectului (2 sau 4) și j este numărul submisiei (1, 2 sau 3), în care pe fiecare linie se află predicția pentru câte un exemplu de test. Fișierul va avea în consecință 323 de linii cu câte un număr, aferente celor 323 de exemple de test.

Exemplu submisie:
Nume fișier: Creanga_Ion_123_subiect_4_solutie_1.txt
Conținut:
1
0
1
3
2
...

b) Codul aferent pentru antrenarea modelelor și obținerea soluțiilor trimise. Pentru fiecare submisie, codul trebuie organizat într-un singur fișier .py denumit astfel: Creanga_Ion_123_solutie_1.py

c) Raportul de la punctul 5.