

Deep Hallucination Classification

Proiectul se concentreaza are ca scop clasificarea imaginilor de tipul "Deep Hallucination" intr-una dintre cele 96 de clase. Cele doua metode de invatare automata utilizate in cadrul acestui proiect sunt Naive Bayes si KNN.

Modelul Naive Bayes

Detalii implemntare:

- Fiecare imagine este citita initial sub forma unui vector de pixeli, acest vector fiind de forma $64 \times 64 \times 3$, intrucat o imagine este compusa din 64 de linii, fiecare continand 64 de coloane, astfel avand 64×64 de pixeli, iar pentru fiecare dintre acesti pixeli cunoscandu-se cele 3 valori RGB.

- Aceste imagini sunt transformate intr-un vector 1D pentru a face prelucarea datelor mai usoara, iar apoi sunt adaugate intr-un vector de trasaturi care contine informatiile pentru fiecare imagine in parte, acest procedeu fiind identic pentru toate cele 3 seturi de date, cele de antrenament, de validare si de testare.

- Valorile din cadrul acestor vectori de trasaturi sunt apoi discretizate, obtinandu-se astfel o forma prelucrata a celor trei vectori mentionati mai sus.

- Se foloseste algoritmul Multinomial Naive Bayes (din cadrul bibliotecii sklearn) pentru a antrena modelul pe datele de antrenament (vectorul de trasaturi prelucrat si clasele corespunzatoare fiecarei imagini reprezentate in cadrul acestuia).

- Dupa antrenare, modelul este folosit pentru a prezice clasele imaginilor de validare. Acuratetea modelului este evaluate utilizand datele de validare si este calculata ca raport intre numarul de predictii corecte si numarul total de imagini de validare.

- Dupa afisarea ratei de acuratete a predictiei facuta pentru imaginile de validare este construita si afisata si matricea de confuzie, mai multe detalii despre aceasta fiind prezentate mai jos.

- Acelasi procedeu se repeat si pentru imaginile de test, unde modelul nostru incearca in mod similar ca in cazul imaginilor de validare sa prezica clasele corespunzatoare acestora, aceste prezicerii fiind in final scrise intr-un csv.

Hyperparameter tuning

De-a lungul numeroaselor rulari ale algoritmului au fost incercate multiple variante pentru parametrii, dar si pentru prelucrarea datelor.

-In ceea ce priveste prelucrarea datelor, aspectul de care a trebuit tinut cont a fost numarul de interval in care dorim sa se faca discretizarea datelor. In final a fost aleasa valoarea 17, intrucat aceasta obtinea cel mai bun scor in cadrul competitiei, insa un lucru interesant de mentionat ar fi faptul ca valoarea 6 a fost cea aleasa initial, intrucat in cadrul datelor de validare obtinea acuratetea cea mai mare (ce e drept diferenta fiind una infima de numai 0.001), insa in cadrul competitiei submisia in care numarul de intervale era 6 fiind cu aproximativ 0.01 mai slaba decat cea in care numarul acestora era 17. Mai multe detalii despre acuratetea predictiilor pe testele de validare pot fi observate in cadrul tabelului si imaginilor de mai jos.

-De asemenea s-a merita precizat si faptul ca s-a incercat si o a doua normalizare, predecesoare celei despre care am vorbit pana acum, in care scadeam din trasaturile unei imagini media acestora, rezultatul fiind mai apoi impartit la deviatia standard a imaginii. Am renuntat repede la aceasta abordare intrucat dupa numai cateva teste s-a observant faptul ca rata de acuratete obtinuta pe testele de validare este una mult mai scazuta.

Mai jos se pot observa date legate de valorile ratele de acuratete obtinute pentru diverse valori ale numarului de interval

```
main x
D:\Facultate\II\S_2\AA\T_2\Proiect\Scripts\python.exe D:/Facultate/II/S_2/IA/Proiect_ML/Proiect_Sol_1/main.py
Acuratete de validare pentru 2 intervale: 0.036
Acuratete de validare pentru 4 intervale: 0.178
Acuratete de validare pentru 6 intervale: 0.192
Acuratete de validare pentru 8 intervale: 0.19
Acuratete de validare pentru 10 intervale: 0.19
Acuratete de validare pentru 12 intervale: 0.189
Acuratete de validare pentru 14 intervale: 0.188
Acuratete de validare pentru 16 intervale: 0.188
Acuratete de validare pentru 18 intervale: 0.188
Acuratete de validare pentru 20 intervale: 0.189
Acuratete de validare pentru 22 intervale: 0.187
Acuratete de validare pentru 24 intervale: 0.187
Acuratete de validare pentru 26 intervale: 0.187
Acuratete de validare pentru 28 intervale: 0.187
Acuratete de validare pentru 30 intervale: 0.19
Acuratete de validare pentru 32 intervale: 0.187
Acuratete de validare pentru 34 intervale: 0.187
Acuratete de validare pentru 36 intervale: 0.185
Acuratete de validare pentru 38 intervale: 0.184
Acuratete de validare pentru 40 intervale: 0.186
Acuratete de validare pentru 42 intervale: 0.183
Acuratete de validare pentru 44 intervale: 0.185
Acuratete de validare pentru 46 intervale: 0.185
Acuratete de validare pentru 48 intervale: 0.183
Acuratete de validare pentru 50 intervale: 0.184
Acuratete de validare pentru 52 intervale: 0.183
Acuratete de validare pentru 54 intervale: 0.184
Acuratete de validare pentru 56 intervale: 0.184
Acuratete de validare pentru 58 intervale: 0.184
Acuratete de validare pentru 60 intervale: 0.184
Acuratete de validare pentru 62 intervale: 0.184
Acuratete de validare pentru 64 intervale: 0.184
Acuratete de validare pentru 66 intervale: 0.184
Acuratete de validare pentru 68 intervale: 0.184
Acuratete de validare pentru 70 intervale: 0.184
Acuratete de validare pentru 72 intervale: 0.184
Acuratete de validare pentru 74 intervale: 0.184
Acuratete de validare pentru 76 intervale: 0.184
Acuratete de validare pentru 78 intervale: 0.184
Run TODO Problems Terminal Python Packages Python Console
```

```
main x
Acuratete de validare pentru 78 intervale: 0.184
Acuratete de validare pentru 80 intervale: 0.184
Acuratete de validare pentru 82 intervale: 0.184
Acuratete de validare pentru 84 intervale: 0.184
Acuratete de validare pentru 86 intervale: 0.184
Acuratete de validare pentru 88 intervale: 0.184
Acuratete de validare pentru 90 intervale: 0.184
Acuratete de validare pentru 92 intervale: 0.184
Acuratete de validare pentru 94 intervale: 0.184
Acuratete de validare pentru 96 intervale: 0.184
Acuratete de validare pentru 98 intervale: 0.184
Acuratete de validare pentru 100 intervale: 0.184
Acuratete de validare pentru 102 intervale: 0.184
Acuratete de validare pentru 104 intervale: 0.184
Acuratete de validare pentru 106 intervale: 0.184
Acuratete de validare pentru 108 intervale: 0.184
Acuratete de validare pentru 110 intervale: 0.184
Acuratete de validare pentru 112 intervale: 0.184
Acuratete de validare pentru 114 intervale: 0.184
Acuratete de validare pentru 116 intervale: 0.184
Run TODO Problems Terminal Python Packages Python Console

main x
Acuratete de validare pentru 79 intervale: 0.184
Acuratete de validare pentru 81 intervale: 0.184
Acuratete de validare pentru 83 intervale: 0.184
Acuratete de validare pentru 85 intervale: 0.184
Acuratete de validare pentru 87 intervale: 0.184
Acuratete de validare pentru 89 intervale: 0.184
Acuratete de validare pentru 91 intervale: 0.184
Acuratete de validare pentru 93 intervale: 0.184
Acuratete de validare pentru 95 intervale: 0.184
Acuratete de validare pentru 97 intervale: 0.184
Acuratete de validare pentru 99 intervale: 0.184
Acuratete de validare pentru 101 intervale: 0.184
Acuratete de validare pentru 103 intervale: 0.184
Acuratete de validare pentru 105 intervale: 0.184
Acuratete de validare pentru 107 intervale: 0.184
Run TODO Problems Terminal Python Packages Python Console
```

```
main
D:\Facultate\II\S_2\AA\T_2\Sol_1_Instanta_2\Scripts\python.exe D:/Facultate/II/S_2/IA/Proiect_ML/Sol_1_Instanta_2/main.py
Acuratete de validare pentru 1 intervale: 0.036
Acuratete de validare pentru 3 intervale: 0.153
Acuratete de validare pentru 5 intervale: 0.183
Acuratete de validare pentru 7 intervale: 0.184
Acuratete de validare pentru 9 intervale: 0.185
Acuratete de validare pentru 11 intervale: 0.191
Acuratete de validare pentru 13 intervale: 0.191
Acuratete de validare pentru 15 intervale: 0.189
Acuratete de validare pentru 17 intervale: 0.191
Acuratete de validare pentru 19 intervale: 0.188
Acuratete de validare pentru 21 intervale: 0.191
Acuratete de validare pentru 23 intervale: 0.186
Acuratete de validare pentru 25 intervale: 0.19
Acuratete de validare pentru 27 intervale: 0.188
Acuratete de validare pentru 29 intervale: 0.185
Acuratete de validare pentru 31 intervale: 0.187
Acuratete de validare pentru 33 intervale: 0.186
Acuratete de validare pentru 35 intervale: 0.186
Acuratete de validare pentru 37 intervale: 0.184
Acuratete de validare pentru 39 intervale: 0.185
Acuratete de validare pentru 41 intervale: 0.186
Acuratete de validare pentru 43 intervale: 0.183
Acuratete de validare pentru 45 intervale: 0.185
Acuratete de validare pentru 47 intervale: 0.184
Acuratete de validare pentru 49 intervale: 0.185
Acuratete de validare pentru 51 intervale: 0.184
Acuratete de validare pentru 53 intervale: 0.183
Acuratete de validare pentru 55 intervale: 0.184
Acuratete de validare pentru 57 intervale: 0.184
Acuratete de validare pentru 59 intervale: 0.184
Acuratete de validare pentru 61 intervale: 0.184
Acuratete de validare pentru 63 intervale: 0.184
Acuratete de validare pentru 65 intervale: 0.184
Acuratete de validare pentru 67 intervale: 0.184
Acuratete de validare pentru 69 intervale: 0.184
Acuratete de validare pentru 71 intervale: 0.184
Acuratete de validare pentru 73 intervale: 0.184
Acuratete de validare pentru 75 intervale: 0.184
Acuratete de validare pentru 77 intervale: 0.184
Run  TODO  Problems  Terminal  Python Packages  Python Console
```