

Tema 2

Recunoaștere Imagine: Compararea Performanței între KNN și Naive Bayes

1.Tema

❖ **Joc: Foarfecă, Piatră, Hârtie**

-aplicatia presupune clasificarea datelor in functie de pozitia mainii in imaginile respective:



2.Set de date

- ❖ Setul de antrenament numit "dataset" este format la randul sau din 3 foldere cu denumiri specifice temei alese: Scissor, Paper si Rock
- ❖ Fiecare folder are urmatoarele caracteristici:
 - un continut de aproximativ 800 de imagini
 - prezinta aceeași extensie (.png)
 - fiecare imagine este etichetata in functie de clasa din care face parte
 - fundalul verde uniform în toate imaginile creează o consistență vizuală în setul de date
- ❖ Setul de date utilizat pentru testarea algoritmului constă într-un număr redus de imagini. La fiecare rulare a programului, imaginile sunt selectate aleatoriu, provenind din setul de antrenament.

3.Etapa de preprocesare si normalizare

- ❖ Imaginile sunt pregatite pentru antrenarea modelului
- ❖ Se parcurg datele din folderul specificat "dataset".
- ❖ Se citește imaginea utilizând OpenCV (cv2.imread).
- ❖ Imaginea este redimensionată la dimensiunile dorite (100x100 pixels) folosind cv2.resize.
- ❖ Imaginea este convertită la tonuri de gri (cv2.cvtColor).
- ❖ Imaginea este aplatizată pentru a fi utilizată în antrenarea modelului (img.flatten)

- ❖ Imaginile și etichetele sunt concatenate pentru a forma setul de date complet.
- ❖ Datele sunt pregătite corespunzător pentru antrenarea modelului și includ procese precum redimensionarea, conversia la tonuri de gri, aplatizarea și normalizarea.

4.Algoritmii utilizati KNN si Naive Bayes

- ❖ KNN și Naive Bayes sunt algoritmi simpli și eficienți, ceea ce îi face potriviți pentru problemele mai puțin complexe.
- ❖ Acestea nu necesită o cantitate mare de date pentru antrenare și pot oferi rezultate acceptabile în condiții de date limitate.

5.Procesul de antrenare si validare

- ❖ Setul de date este împărțit în setul de antrenare și setul de testare folosind “train_test_split” din biblioteca sklearn.
- ❖ Aproximativ 80% din date sunt folosite pentru antrenare și 20% pentru testare
- ❖ KNN este configurat cu 3 vecini, iar modelul Naive Bayes este antrenat fără parametri specifici
- ❖ Este masurata acuratețea, generând raportul de clasificare și afișând matricea de confuzie, oferind informații detaliate despre performanța modelului pentru fiecare clasă (scissor, paper, rock).

KNN

```
Classification Report for knn:
              precision    recall  f1-score   support

    paper      0.91      0.90      0.90      151
     rock      0.88      0.92      0.90      134
  scissor      0.93      0.91      0.92      153

 accuracy      0.91
 macro avg      0.91      0.91      0.91      438
weighted avg      0.91      0.91      0.91      438


Confusion Matrix for knn:
[[136   9   6]
 [  7 123   4]
 [  7   7 139]]
K-Nearest Neighbors Accuracy: 0.908675799086758
```

Naïve Bayes

```
Classification Report for nb:
              precision    recall  f1-score   support

   paper       0.80        0.77        0.78        151
   rock       0.73        0.72        0.72        134
  scissor     0.81        0.86        0.83        153

 accuracy          0.78          0.78          0.78          438
 macro avg         0.78          0.78          0.78          438
 weighted avg      0.78          0.78          0.78          438

Confusion Matrix for nb:
[[116  26   9]
 [ 17  96  21]
 [ 12  10 131]]
Naive Bayes Accuracy: 0.7831050228310502
```

6.Procesul de testare

- ❖ Clasificările obținute prin predicție sunt comparate cu etichetele reale asociate imaginilor de testare
- ❖ Se indică dacă predicția este corectă sau incorectă, iar în cazul unei predicții incorecte, se afișează eticheta reală și cea prevăzută.

```
Rezultatele testării K-Nearest Neighbors:
Predictie Corecta: Eticheta Reală: paper, Predictie: paper
Predictie Gresita: Eticheta Reală: paper, Predictie: rock
Predictie Corecta: Eticheta Reală: paper, Predictie: paper
Predictie Corecta: Eticheta Reală: scissor, Predictie: scissor
Predictie Corecta: Eticheta Reală: rock, Predictie: rock

Procentaj de Reușită: 80.00%

Rezultatele testării Naive Bayes:
Predictie Corecta: Eticheta Reală: paper, Predictie: paper
Predictie Corecta: Eticheta Reală: paper, Predictie: paper
Predictie Corecta: Eticheta Reală: paper, Predictie: paper
Predictie Corecta: Eticheta Reală: scissor, Predictie: scissor
Predictie Corecta: Eticheta Reală: rock, Predictie: rock

Procentaj de Reușită: 100.00%
```

- ❖ Se calculează acuratețea globală a modelului, reprezentând proporția imaginilor corect clasificate

7. Concluzii si observatii

- ❖ Din rezultate se poate observa ca algoritmul KNN are o acuratete mai buna decat Naïve Bayes:

K-Nearest Neighbors Accuracy: 0.908675799086758

Naive Bayes Accuracy: 0.7831050228310502

- astfel, rezulta ca primul algoritm utilizat are o adaptabilitate mai buna asupra setului de date ales
- in raportul de clasificare se vede o usoara scadere a valorilor de performanta in cazul algoritmului Naïve Bayes
- de asemenea, se observa aceeași diferență și la nivelul matricei de confuzie, unde numărul de exemple clasificate drept True Positive și True Negative, este mai mare în cazul algoritmului KNN

În concluzie, aplicatia evidențiază că algoritmul K-Nearest Neighbors (KNN) demonstrează o adaptabilitate superioară asupra setului de date, obținând o performanță mai bună în comparație cu Naive Bayes. Acest aspect sugerează că KNN poate fi o opțiune mai eficientă pentru clasificarea imaginilor asociate cu jocul "Scissor, Paper, Rock".

❖ 8. Biblioteci utilizate

- os
- random
- cv2 (OpenCV)
- numpy
- sklearn
- train_test_split
- KNeighborsClassifier
- GaussianNB
- accuracy_score, classification_report, confusion_matri
- <https://www.kaggle.com/datasets/drgfreeman/rockpaperscissors>