**Clasificator pentru imagini CNN**

**Microorganisme**

Gradinariu Robert-Iulian

Grupa 30233 CTI

Datasetul pentru acest clasificator de imagini consta in imagini cu microorganisme, acesta fiind impartit in 8 tipuri specifice: Amoeba, Euglena, Hydra, Paramecium, Rod\_bacteria, Spherical\_bacteria, Spiral\_bacteria, Yeast.

Setul de imagini este trecut printr-un set de transformari care sunt utilizate pentru augmentarea imaginilor. Aceste transformari sunt aplicate in timpul incarcarii datelor pentru a creste diversitatea setului de antrenament si a ajuta modelul sa generalizeze mai bine. Primul pas este redimensionarea imaginilor la 128x128 pixeli, rotatie aleatorie cu un unghi maxim de 10 grade, modificari aleatorii ale luminiozitatii, contrastului, saturatiei si nuantei, iar apoi transformarea imaginii in tensor care normalizeaza pixelii in intervalul [0,1].

Ulterior setul este impartit in train ( 80% ) , validation (10%) si test (10%).

Urmeaza definirea modelului care este o retea neurala convolutionala (CNN), care consta dintr-un strat convolutional, o activare ReLU, un strat de pooling si un strat complet conectat.

Analiza performantelor modelului se poate face in functie de learning rate, dimensiunea batch-ului, numarul de epoci si optimizatoare.

Learning Rate-ul controleaza cat de mari sunt pasii pe care ii face optimizatorul atunci cand actualizeaza parametrii modelului. Am testat cu lr=0.001 si lr=0.01 si cel mai mare a fost cel mai benefic.

Batch Size ul reprezinta numarul de exemple din setul de date pe care modelul le proceseaza inainte de a actualiza parametrii. Am incercat cu batch size 32,64 si 128, dar cu un batch size mai mare de ex 128 actualizarile sunt mai stabile si mai precise, chiar daca necesita mai multa memorie.

Numarul de epoci reprezinta de cate ori modelul parcurge intregul set de date de antrenament. In cazul in care numarul de epoci este prea mic, modelul poate fi subantrenat si nu va fi capabil sa invete caracteristici importante din setul de date. In cazul opus, daca sunt prea multe modelul poate face overfitting, invatand din zgomotul din setul de date. Am testat modelul pe 20 de epoci.

Optimizatorii sunt algoritmi care ajusteaza parametrii modelului pentru a minimiza functia de pierdere. SGD ( Stochastic Gradient Descent ) actualizeaza parametrii folosind gradientul estimat pt fiecare lot. Adam e o combinatie intre RMSprop si momentum, adaptand ratele de invatare pentru fiecare parametru, acesta fiind mai eficient pentru setul meu de date.