

Vizualizarea și Inițializarea Custom a Weight-urilor în Rețele Neuronale Convoluționale

Proiect ASI

Autori: Stoian Robert Alin, Enache Paul, Voinea Mario

Rezumat

Acum proiect explorează conceptul de inițializare customizată a weight-urilor într-o rețea neuronală conlovuțională (CNN). Demonstrăm experimental că este posibil să inițializăm weight-urile unui layer Dense cu o matrice predefinită (în cazul nostru, o imagine stilizată a unei rațe) și să observăm cum această structură evoluă pe parcursul antrenamentului. Proiectul este inspirat din cercetările recente privind „Lazy Training” și comportamentul weight-urilor în rețele neuronale largi.

1. Articole Relevantă și Lucrări Anterioare (State of the Art)

1.1 Effect of the Initial Configuration of Weights on the Training and Function of Artificial Neural Networks

Citare: Grzeszczuk et al., 2020

([arXiv: 2012.02550](#))

Această lucrare demonstrează experimental că, dacă rețeaua nu intră într-o fază de overfitting, greutățile rămân în vecinătatea inițializării. În experimentul autorilor, o literă „a” este desenată direct în matricea de weight-uri a unui strat Dense și se observă că aceasta rămâne vizibilă chiar și după finalizarea antrenamentului.

Aceasta reprezintă **inspirația principală a proiectului nostru**, în care am replicat experimentul folosind imaginea unei rațe ca inițializare custom.

Contribuție cheie: Vizualizarea directă a faptului că weight-urile nu se îndepărtează semnificativ de inițializare în anumite condiții de antrenament.

1.2 On Lazy Training in Differentiable Programming

Citare: Chizat et al., 2018

([arXiv: 1812.07956](#))

Lucrarea introduce conceptul de „**Lazy Training**”, fenomen prin care rețelele neuronale foarte largi își modifică weight-urile doar marginal față de valorile inițiale. Acest comportament contrazice intuiția clasică conform căreia antrenamentul ar trebui să schimbe semnificativ toți parametrii rețelei.

Contribuție cheie: Formalizarea teoretică a comportamentului „leneș” al gradienților în rețele supraparametrizate.

1.3 Neural Tangent Kernel: Convergence and Generalization in Neural Networks

Citare: Jacot et al., 2018
[\(arXiv: 1806.07572\)](#)

Această lucrare fundamentală introduce conceptul de **Neural Tangent Kernel (NTK)**. Autorii demonstrează matematic că, într-o rețea infinit de largă, weight-urile rămân practic fixe pe durata antrenamentului. În acest regim, rețeaua neuronală se comportă ca un model liniar într-un spațiu de feature-uri indus de kernel.

Contribuție cheie: Baza teoretică pentru înțelegerea importanței inițializării și a persistenței weight-urilor pe parcursul optimizării.

1.4 The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks

Citare: Frankle și Carlin, 2018
[\(arXiv: 1803.03635\)](#)

Această lucrare arată că **inițializarea joacă un rol crucial** în succesul antrenării. Autorii demonstrează existența unor subrețele sparse („lottery tickets”) care, cu inițializarea corectă, pot atinge performanțe comparabile cu rețeaua completă. Structura de la momentul inițial dictează traectoria optimizării.

Contribuție cheie: Importanța critică a inițializării în determinarea performanței finale a rețelei.

1.5 Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification

Citare: He et al., 2015
[\(arXiv: 1502.01852\)](#)

Cunoscută sub numele de „**He Initialization**”, această lucrare definește un standard matematic pentru inițializarea weight-urilor în rețelele cu funcții de activare ReLU. Este propusă o varianță specifică a distribuției inițiale pentru a preveni dispariția sau explozia gradienților.

Contribuție cheie: Formula standard pentru inițializarea weight-urilor:

$$\text{Var}(W) = 2 / n \square_i \square$$

2. Ce date folosim si Analiza Exploratorie (EDA)

Setul de date principal: MNIST

- Sursa: Modified National Institute of Standards and Technology database
- Continut: Imagini cu cifre scrise de mana (clasele 0-9)
- Dimensiuni: 60.000 imagini de antrenare si 10.000 de testare
- Format: Grayscale, 28x28 pixeli

Exploratory Data Analysis (EDA):

- **Echilibrul Claselor:** Am analizat distributia etichetelor si am constatat ca setul este perfect echilibrat (aproximativ 6000 de exemple pentru fiecare cifra). Acest lucru elimina necesitatea unor tehnici precum SMOTE sau class weighting
 - **Distributia Pixelilor:** Valorile originale sunt intre 0 si 255. Majoritatea pixelilor sunt 0 (fundal negru), iar informatia este concentrata in centrul imaginii
 - Am normalizat pixelii in intervalul [-1, 1]. Aceasta transformare este pentru ca algoritmul SGD sa convearga corect si pentru a mentine stabilitatea numerica a "desenului" nostru
-

3. Ce modele folosim si necesarul de calcul

Arhitectura Modelului (CNN_Duck):

1. Feature Extractor (Convolutii):

- a. Strat Conv2d: 1 canal intrare, 10 canale iesire, kernel 5x5.
- b. Strat MaxPool2d: Reduce dimensiunea spatiala la jumata (2x2).
- c. Rol: Extrage trasaturi vizuale simple (linii, curbe) din cifre.

2. Classifier (Strat Dense - Experimental):

- a. Input: 1440 neuroni (rezultatul aparatizarii: 10 canale x 12 x 12).
- b. Output: 10 neuroni (probabilitatile pentru cifrele 0-9).
- c. Inovatie: Acest strat nu este initializat random, ci cu matricea noastra "**Rata**".

Necesarul de calcul :

T4 (optional). Modelul este foarte eficient ("lightweight") si poate fi antrenat in sub **2 minute** chiar si pe un CPU standard (Intel Xeon), ca memorie RAM sub **1GB**, iar setul ocupa maxim **50 MB**.

4. Metode de evaluare si comparatie

Evaluarea Cantitativa, Trebuie sa demonstram ca reteaua "invata" si nu ramane blocata.

- Metrica: CrossEntropyLoss (Functia de pierdere).
- Obiectiv: Scaderea Loss-ului sub valoarea de **0.1**.
- Rezultat: Modelul atinge o acuratete ridicata, demonstrand ca structura initiala ("Rata") este capabila sa rezolve task-ul de clasificare a cifrelor.

Evaluarea Calitativa, aceasta este metoda principală prin care validam ipoteza lucrarii.

- Metoda: Extragerea tensorului de greutati **W** din stratul Dense la intervale regulate (fiecare 500 de pasi)
- Comparatia:
 - **Moment T0 (Start)**: Vizualizam matricea de greutati ca o imagine 2D. Aceasta trebuie sa fie clara ("Rata" perfecta).
 - **Moment T_Final (Epoca 3)**: Vizualizam aceeasi matrice dupa antrenare.

Criteriu de Succes: Daca in imaginea finala smile face-ul este inca vizibila, experimentul este reusit. Acest lucru dovedeste vizual ca SGD a pastrat informatia initiala, confirmand ca optimizarea s-a facut local.