Universitatea Politehnica din București

Facultatea de Automatică și Calculatoare

Departamentul de Calculatoare

Icon

Description automatically generatedA picture containing diagram

Description automatically generated

Task de predicție

Nume: Ion-Robert Dinica, ISI

București 2022

Cuprins

[Introducere 3](#_Toc93163672)

[Prezentare model 3](#_Toc93163673)

[Hiperparametrii 3](#_Toc93163674)

# Introducere

Pentru rezolvarea acestui task am ales să văd problema ca una de clasificare în care rating-urile sunt clasele pe care modelul trebuie sa le prezică. Astfel, am construit un model bazat pe Învățarea Profundă, avand la baza BERT. Modelul va primi la intrare review-urile și va avea ca output clasa (rating-ul) pentru review-ul respectiv.

# Prezentare model

În continuare voi realiza o prezentare a modelului folosit. Acest model are următoarea arhitectură:

* BERT pentru generarea de embeddings
* Dropout cu valoarea de 0.4 pentru a reduce overfitting-ul
* Un strat liniar (Linear) din modeulul torch
* Funcția de activare relu

Pentru fiecare batch output-ul obtinut de la BERT, care are dimensiunea hidden de 768, ajunge la un start liniar care primeste intrari de dimensiunea 768 și are output-ul de dimensiune 6. Peste iesirea de la stratul liniar este aplicată funcția de activare relu.

Pentru antrenarea modelului fișierul “train.json” a fost împarțit în “train\_dataset” și “validation\_and\_test”, iar cel din urmă a fost mai departe împărțit în “validation\_dataset” și “train\_dataset”. În primul caz raportul de împărțire este 80% pentru antrenare și 20% pentru validare și testare. În cel de-al doilea caz raportul de împărțire a fost 10% pentru testare și 10% pentru validare.

Pentru alegerea celui mai bun model s-a folosit metrica f1 score, astfel încât în etapa de validare se va salva modelul care a produs cea mai buna valoare pentru aceasta metrică.

Testarea modelului se face atât pe datset-ul de test obținut din împârțirea prezentată mai sus, cât și pe dataset-ul obținut din fișierul “test\_wor.json” caz în care se realizează și substituția rating-urilor din acest fișier cu cele prezise de model.

# Hiperparametrii

Pentru acest model s-a folosit optimizatorul Adam, valoarea pentru rata de învățare , iar funcția de loss este cross entropy din modulul torch.

Numarul de epoci utilizate pentru antrenarea modelului este 10, iar dimensiunea pentru batch-urile folosite este 16.

Având în vedere faptul că review-urile din datset-ul oferit erau în limba română, pentru BERT s-a folosit o versiune pre-antrenată pentru limba romană: readerbench/RoBERT-base.

Link catre cod: <https://github.com/kuiper28/prediction-task>