



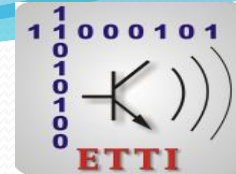
Sistem de recunoaștere și clasificare a plânsetelor nou-născuților folosind Deep Learning

Conducător științific:
Ș.L. Dr. Ing. Horia Cucu

Absolvent:
Radu-Ștefan Deleanu

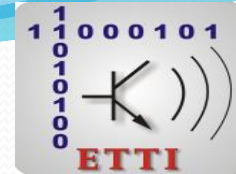
Speech & Dialogue Research Laboratory
Faculty of Electronics, Telecommunications and Information Technology
University POLITEHNICA of Bucharest

Cuprins



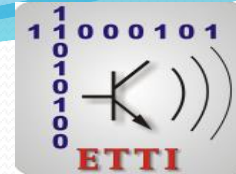
- Introducere și context
- Obiectivele proiectului
- Rețele neurale profunde
- Structura sistemelor
- Tehnologii folosite
- Baza de date SPLANN
- Experimente și rezultate
- Concluzii și perspective

Introducere și context



- Studii recente au arătat că plânsetele nou născuților cu vârste mai mici de 3 luni conțin informație relevantă pentru nevoia pe care o au.
- Părinții neexperimentați ar putea beneficia substanțial de pe urma unui sistem automat capabil să determine nevoia bebelușului.
- Anterior s-au încercat diverse abordări pentru a implementa sisteme de clasificare bazate pe învățare automată, însă încă nu există aplicații comerciale.
- Tema clasificării plânsetelor de nou-născuți este încă în etapa de cercetare.

Obiectivele proiectului



- Generarea unei baze de date formată din trăsături extrase din baza de date SPLANN.
- Proiectarea și dezvoltarea unui sistem de clasificare folosind o arhitectură de rețea neurală profundă personalizată și observarea performanțelor acesteia în funcție de variația hiperparametrilor.
- Implementarea și testarea unor arhitecturi de rețele neurale deja existente pe baza de date generată anterior și studiul performanțelor acestora comparativ cu cele obținute de rețeaua personalizată.
- Cercetarea viabilității modelelor implementate.

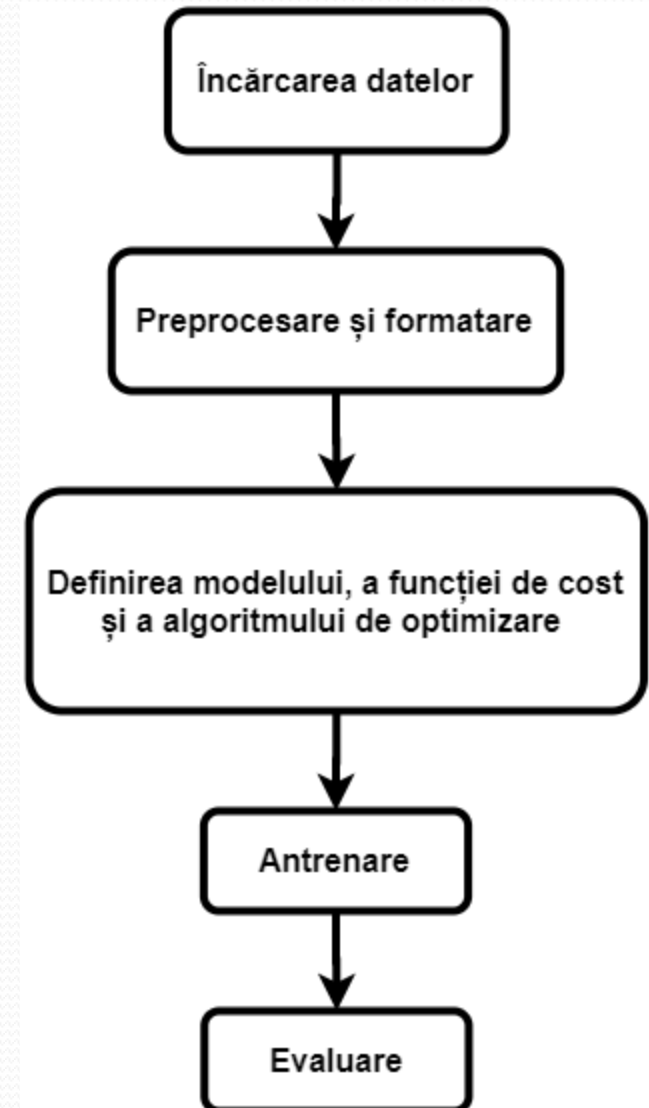
- Au în general un număr mai mare de straturi ascunse.
- Au cunoscut o puternică dezvoltare în ultimii ani, dată fiind evoluția tehnologică.
- Sunt capabile să modeleze date de o complexitate ridicată, reușind să rezolve sarcini mai dificile decât rețelele clasice.
- Au nevoie de un volum masiv de date etichetate pentru a putea modela eficient.

- Au fost realizate 3 sisteme de învățare automată.
 - MLP
 - LinearSVC
 - TabNet
- Sunt independente, dar lucrează pe aceeași bază de date.
- Relevă informații diferite despre natura sarcinii de clasificare.

Structura sistemelor



- Cele 3 sisteme respectă structura alăturată.
- Unele module sunt implementate diferit în funcție de sistem dar au aceleași funcționalități.
- Există și blocuri opționale, care nu au efecte asupra performanțelor modelelor.



Tehnologii folosite

- Extragere de trăsături
 - openSMILE
- Dezvoltarea sistemelor
 - Limbajul de programare Python
 - Platforma Google Colab
- Biblioteci de învățare automată
 - Pytorch
 - Sckit-learn

openSMILE:)
by audEERING™

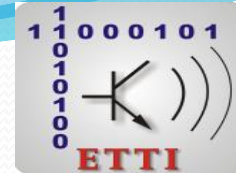


colab

 PyTorch

scikit
learn

Baza de date SPLANN



- Conține 13373 de fișiere audio reprezentative pentru 7 categorii de nevoi:
 - colici, eructație, durere, disconfort, foame, oboseala, durere patologică
- S-a renunțat la durere patologică pentru a avea rezultate comparabile cu celelalte lucrări în domeniu.
- Este puternic debalansată.
- Împărțire pe loturi:
 - 90% set de antrenare
 - 10% set de test
 - Pentru experimentul cu TabNet s-a extras un lot de validare de 10% din setul de antrenare

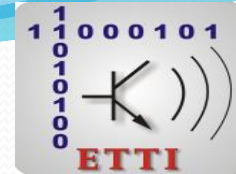
Format cu trăsături extrase CSV

```

1 name;frameTime;audspec_lengthLlnorm_sma_range;audspec_lengthLlnorm_sma_maxPos;auds
2 '0001_Colics_8_6_1-01.wav';0.000000;1.317413e+000;5.268817e-001;0;5.855656e-001;1.
3 '0002_Colics_8_6_1-02.wav';0.000000;2.570976e+000;3.076923e-001;0;1.769539e+000;1.
4 '0003_Colics_8_6_1-03.wav';0.000000;1.473857e+000;5.087720e-001;0;1.079422e+000;1.
5 '0004_Colics_8_6_1-04.wav';0.000000;2.904706e+000;8.823529e-001;0;1.307903e+000;1.
6 '0005_Colics_8_6_1-05.wav';0.000000;4.379579e+000;7.538462e-001;1.538462e-002;7.00
7 '0006_Colics_8_6_1-06.wav';0.000000;2.012293e+000;7.341772e-001;0;1.771937e+000;2.
8 '0007_Colics_8_6_1-07.wav';0.000000;3.047776e+000;5.882353e-001;0;2.142119e+000;2.
9 '0008_Colics_8_6_1-08.wav';0.000000;2.319407e+000;4.210526e-001;0;1.110577e+000;1.
10 '0009_Colics_8_6_1-09.wav';0.000000;2.560218e+000;6.901408e-001;2.816901e-002;8.63
11 '0010_Colics_8_6_1-10.wav';0.000000;2.568320e+000;8.725490e-001;0;8.992715e-001;1.
12 '0011_Colics_8_6_1-11.wav';0.000000;1.475347e+000;4.339623e-001;0;1.375066e+000;1.
13 '0012_Colics_8_6_1-12.wav';0.000000;2.402212e+000;4.453125e-001;1.562500e-002;1.15
14 '0013_Colics_8_6_1-13.wav';0.000000;9.272983e-001;1.194030e-001;5.671642e-001;1.74
15 '0014_Colics_8_6_1-14.wav';0.000000;2.044401e+000;1.944444e-001;1.851852e-002;1.45
16 '0015_Colics_8_6_1-15.wav';0.000000;1.311135e+000;2.207792e-001;3.896104e-002;6.02
17 '0016_Colics_8_6_1-16.wav';0.000000;2.710335e+000;1.683168e-001;4.009901e-001;4.84
18 '0017_Colics_8_6_1-17.wav';0.000000;2.315171e+000;6.438356e-001;0;1.929054e+000;2.
19 '0018_Colics_8_6_2-01.wav';0.000000;3.224159e+000;4.675325e-001;1.298701e-002;1.52
20 '0019_Colics_8_6_2-02.wav';0.000000;2.242179e+000;3.235294e-001;0;2.307882e+000;2.
21 '0020_Colics_8_6_2-03.wav';0.000000;2.205568e+000;6.727273e-001;0;1.805326e+000;2.
22 '0021_Colics_8_6_2-04.wav';0.000000;2.326138e+000;3.437500e-001;0;2.173199e+000;2.
23 '0022_Colics_8_6_2-05.wav';0.000000;2.520874e+000;4.375000e-001;0;1.771976e+000;2.

```

Experimente și rezultate



- Pentru sistemul bazat pe model MLP:
 - S-au încercat câteva variante de arhitecturi, s-au variat hiperparametri
 - S-au folosit straturi complet conectate, cu funcția de activare ReLU, cu funcția de cost Entropie Încrucișată și cu algoritmul de optimizare Adam.
 - S-a întâmpinat fenomenul de supraantrenare (a fost remediat prin folosirea tehnicii de renunțare, balansarea setului de date și reducerea dimensiunii rețelei).
 - Cea mai bună acuratețe a fost de aproximativ 39% pe lotul de test.

Experimente și rezultate

- Pentru experimentele cu LinearSVC:
 - S-a încercat folosirea unor combinații de clase pentru a cerceta influența creșterii complexității sarcinii de clasificare în rezultate.
 - S-a experimentat cu toate combinațiile posibile de câte două clase pentru a determina măsura în care acestea sunt separabile liniar.
 - S-a observat că anumite clase manifestă un înalt grad de similaritate în timp ce altele nu reprezintă o problemă pentru un clasificator liniar.
 - Aceste informații sunt relevante pentru eventuale viitoare abordări ale temei.

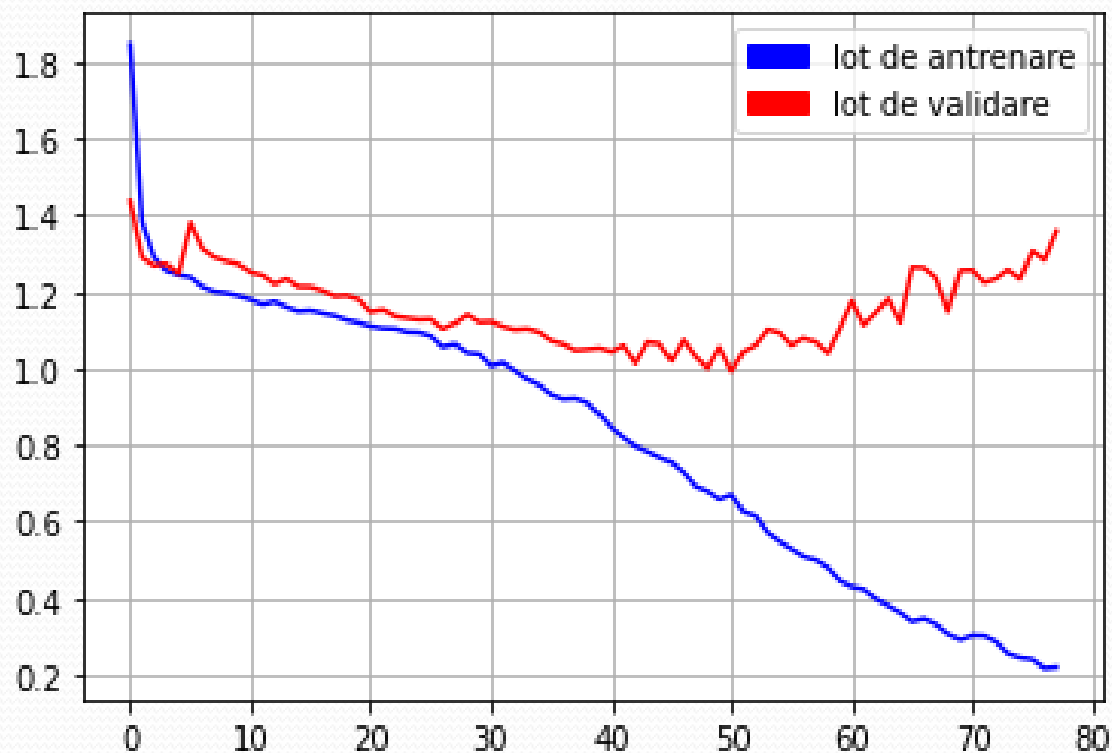
Grad de similaritate între clase

Clasa	0 (colici)	1 (disconfort)	2 (eructație)	3 (foame)	4 (durere)	5 (oboseală)
0 (colici)		96.15%	76%	95,90%	97,64%	100%
1 (disconfort)	96.15%		71,42%	52,85%	54,99 %	97.18%
2 (eructație)	76%	71,42%		87,23%	89,02%	51,85%
3 (foame)	95,90%	52,85%	87,23%		66%	98.76%
4 (durere)	97,64%	54,99 %	89,02%	66%		99,10%
5 (oboseală)	100%	97.18%	51,85%	98.76%	99,10%	

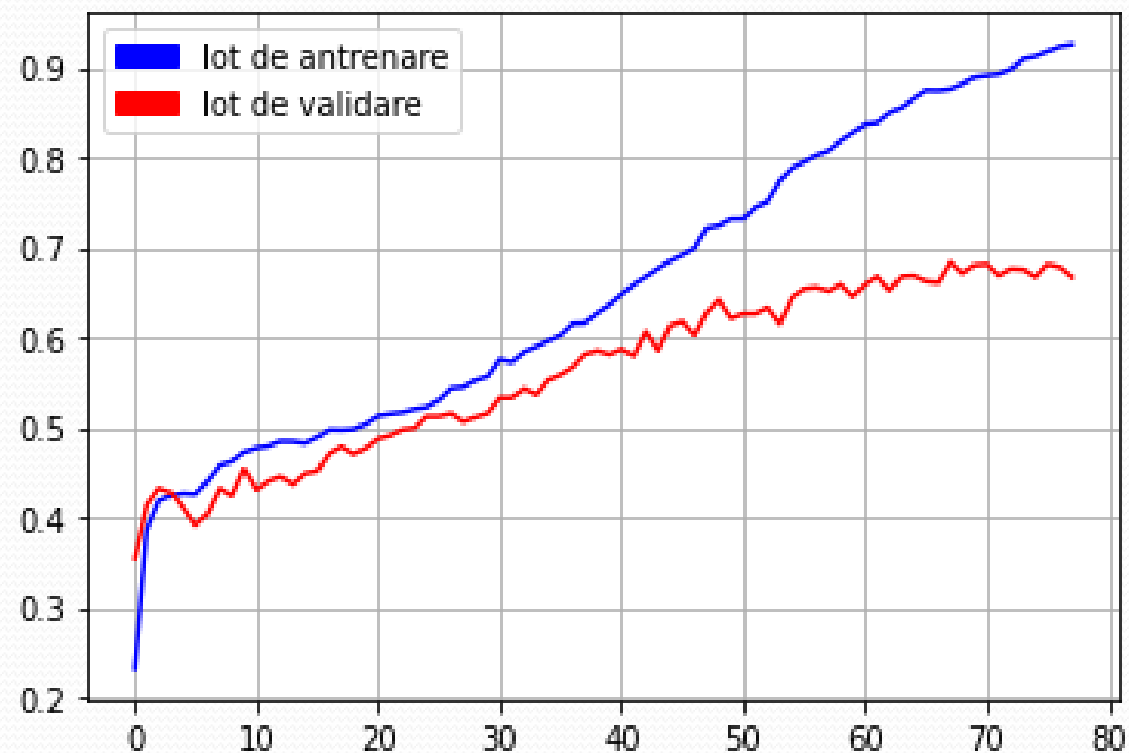
Experimente și rezultate

- Pentru experimentele cu TabNet:
 - S-a cercetat viabilitatea unui sistem care să profite de natura tabulară a setului de date format din trăsături extrase.
 - S-au folosit diverse configurații ale parametrilor rețelei în scopul obținerii unei performanțe mai bune.
 - S-au obținut cel mai mare scor din cadrul proiectului și anume o acuratețe de 47% pe lotul de test.

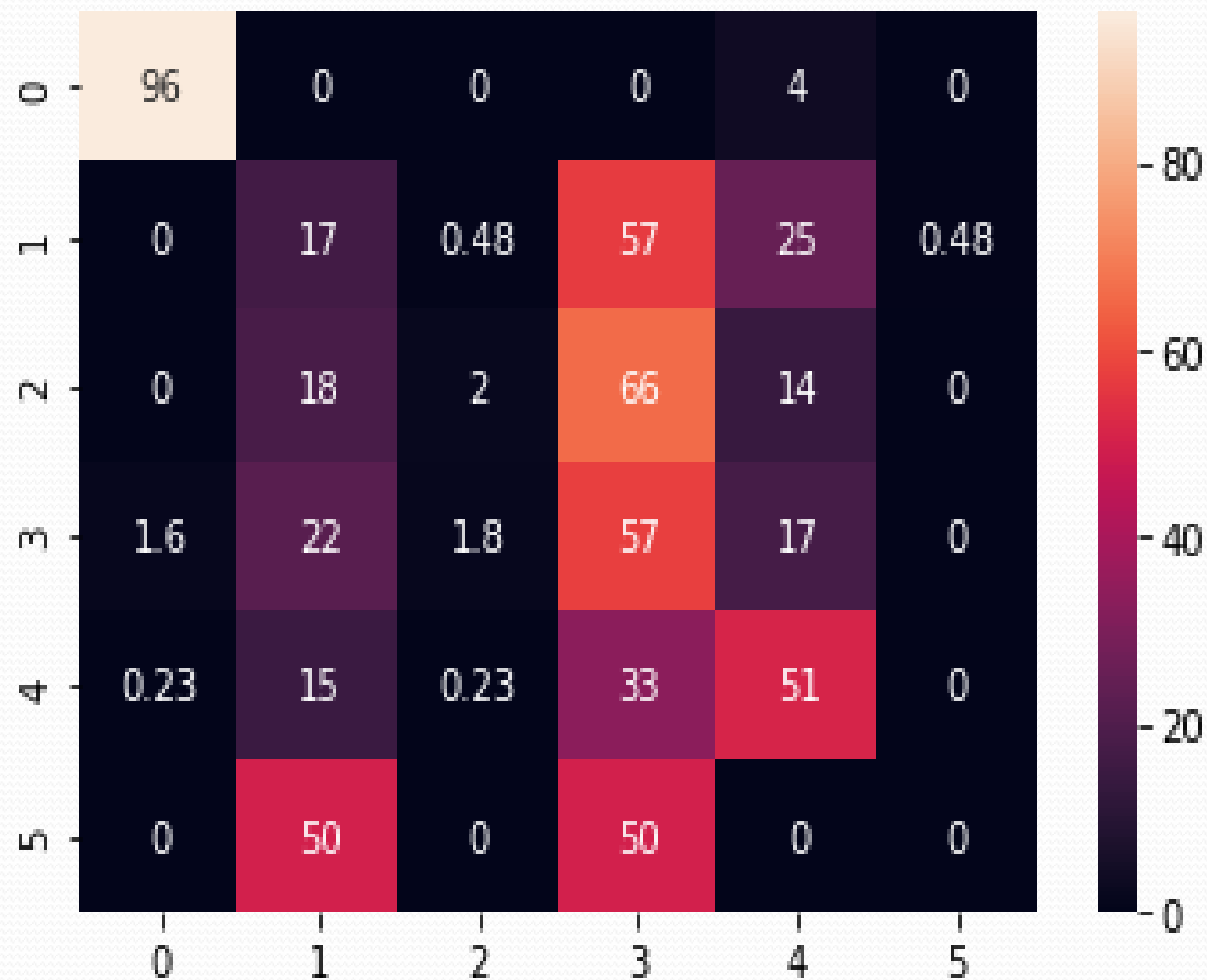
Evoluția valorilor funcției de cost pe epoci



Evoluția acurateții pe epoci



Matrice de confuzie TabNet



Concluzii

- În urma experimentelor s-a relevat complexitatea sarcinii de clasificare de plânset de bebeluș, date fiind rezultatele fiecărui sistem.
- Experimentele cu MLP au arătat că o astfel de arhitectură nu este neapărat eficientă, dată fiind dimensionalitatea înaltă a datelor (6373 de parametri).
- Clasificatorul LinearSVC a arătat similaritatea claselor și măsura în care acestea pot fi separate liniar.
- Rețeaua TabNet a produs cele mai bune rezultate, profitând de natura tabulară a setului de date.

Contribuții personale

- Extragerea de trăsături din baza de date pusă la dispoziție de laboratorul Speed.
- Dezvoltarea modulelor de citire de date, preprocesare și formatare.
- Proiectarea și implementarea integrală în cod a sistemelor de clasificare.
- Conceperea și realizarea experimentelor.
- Interpretarea rezultatelor obținute.

Perspective de dezvoltare

- Aplicarea unui procedeu de optimizare automată a parametrilor rețelei TabNet, în scopul obținerii unei performanțe mai bune, date fiind rezultatele promițătoare obținute de aceasta.
- Implementarea unui mecanism capabil să realizeze o selecție a celor mai bune trăsături, reducând astfel gradul ridicat de dimensionalitate al datelor, cu posibilitatea de îmbunătățire a performanțelor modelelor de tip MLP.
- Încercarea unei abordări diferite care folosește tehnici de învățare profundă, spre exemplu folosirea unei rețele neurale convoluționale, care lucrează pe spectrograme extrase din fișierele audio;

Vă mulțumesc pentru atenție!