Klasifikacija Urbanih Zvukova

Branko Grbić, Željko Milovanović - Matematički fakultet, Univerzitet u Beogradu

7. septembar 2023.

Sadržaj

1 Sažetak

U ovom radu predstavljeno je poređenje više modela za audio klasifikaciju urbanih zvukova. Napisana CNN i duboko povezana neuralna mreža u popularnoj Python biblioteci PyTorch, upoređena je sa pretreniranim VGG-11[1] modelom sa normalizacijom serije. Za klasifikaciju sva 3 modela, korišćen je poznati atribut za klasifikaciju zvuka - MFCC (Mel Frequency Cepstrum Coefficient). Skup podataka, URBANSOUNDS8K, korišćen u svrhe klasiifkacije, sadrži 8732 labeliranih zvukova, podeljenih u 10 klasa. Rezultati pokazuju da se VGG-11 znatno bolje ponaša u odnosu na napisani CNN, dajući nam do znanja značaj pretreniranosti modela. Kod koji prati ovaj rad, dostupan je na Github-u

2 Uvod

Audio klasifikacija zastupa jako sirok, ali nezanemarljivo bitan spektar problema. Veliki rast u količini informacija, zahteva od nas sve bolje metode za rešavanje i klasifikaciju tih podataka. Nove tehnologije traže bolja i efektivnija rešenja, u skladu sa novim problemima koji nadolaze, kao što su na primer "pametni"gradovi ili internet stvari, koje očekuju neku vrstu filtera za ono što im neće biti korisno, odnosno kontraproduktivno za njihov rad. Otuda nastaje i značaj nečega kao što je klasifikacija urbanog zvuka, koji se može služiti u odstranjivanju ili detekciji zvukova neprikladnim određenim uređajima.

3 Opis skupa podataka

Skup podataka URBANSOUNDS8K [11], sadrži 8732 audio isečaka manjih ili jednakih od 4 sekunde, labelirani u sledećih 10 klasa:

- Klima
- Truba automobila
- Graja dece
- Lajanje kučića
- Bušenje
- Rad motora
- Pucanj Pištolja
- Mehanićki Čekić
- Sirena
- Ulićna muzika

Podaci su takođe podeljeni u 10 grupa, nasumično izmešani od strane kreatora[3], očekujući od inženjera da trenira svoje modele na 9 grupa, desetu ostavljajući za test, odnosnu proveru tačnosti. Iziskuje se takođe da za svaku različitu test grupu, postoji različit model koji se testira na tome, te će se za finalnu metriku klasifikacije koristiti uprosečena tačnost dobijenih 10 modela, sa dodatnim poželjnim podacima kao što je standardna devijacija tačnosti modela.

4 Preprocesiranje podataka

Iz csv-a učitana je putanja audio fajla, koja se dalje učitava i obrađuje. Frekvencija odabiranja iznosi 44100 Hz. Uzimanje celog uzorka od 30 sekundi predstavlja enormnu količinu podataka u jednoj pesmi za obradu. Baš se zato, smanjuje količina podataka neophodna za obradu podataka

na vremenski period od 2 sekunde. Uzorci ce biti konvertovani u jedan kanal, odakle će se vršiti dalja obrada signala.

Za sve transformacije, korišćena je librose biblioteka. Kasnije, podaci se šalju dalje u PyTorch-evu DataLoader klasu gde se podaci skupljaju u grupe (batch-eve) i prosledjuju na trening, odnosno test

4.1 Više o tehnikama audio procesiranja

Pre nego što se izvrši uvod u transformaciju uzorka, smatramo da treba da ukratko opišemo šta predstavljaju korišćene transformacije

4.1.1 RMS

Termin RMS predstavlja kvadratni koren srednje vrednosti koja se kvadrira (eng. Root Mean Square). Atribut predstavlja energiju uzorka, glasniji fajl će imati veći RMS, koji je zbog svoje karakteristike računa, manje podložan uticajima ekstremnih vrednosti.

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$$

4.1.2 Spektralni centroid

Spektralni centroid predstavlja centar mase signala.

$$SpectralCentroid = \frac{\sum_f f \cdot S(f)}{\sum_f S(f)}$$

4.1.3 Spektralna širina opsega

Spektralna širina opsega meri širinu signala, odnosno kako je signal distribuiran kroz raličite frekvencije.

$$SpectralBandwidth = \sqrt{\frac{\sum_f (f - \text{Centroid})^2 \cdot S(f)}{\sum_f S(f)}}$$

4.1.4 Spektralno opadanje

Spektralno opadanje (eng. Spectral rolloff) je definisano kao frekvencija ispod koje leži određeni procenat energije.

$$SpectralRolloff = \sum_f S(f) \leq \text{Rolloff Percentage} \times \sum_f S(f_{\max})$$

4.1.5 Stopa nultih prelaza

Stopa nultih prelaza, kao što samo ime nalaže, broji koliko puta signal pređe nultu vrednost

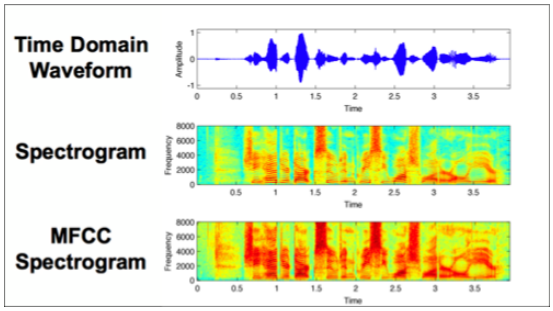
$$ZCR = \frac{1}{N-1}\sum_{n=1}^{N-1} |sgn(x[n]) - sgn(x[n-1])|$$

4.1.6 MFCC

MFCC (eng. Mel Frequency Cepstrum Coefficient) predstavlja kepstar uzorka [2].

Dati uzorak će se procesirati koristeći DFT (diskretnu Furijeovu transformaciju) koja pretvara vremenski domen u frekvencijski. Bitan dodatak jeste da se spektrogram dalje obrađuje korišćenjem

Mel skale[4], koja uzima u obzir ljudsko rezonovanje zvuka. Posle te transformacije, pravi se kepstar, korišćenjem IFT (inverzne Furijeove transformacije) na logaritamskom spektru [10].



Slika 1. MFCC u poređenju sa audio signalom i njegovim spektogramom

4.2 Obrada audio uzorka za FFNN

Korišćeni su sledeći atributi, koji su kasnije usrednjeni radi pretvaranja spektograma u jednu dimenziju:

- RMS
- Spektralni centroid
- Spektralna širina opsega
- Spektralno opadanje
- Stopa nultih prelaza
- MFCC

4.3 Obrada audio uzorka za CNN i VGG-11

Zbog njegove značajnosti, korišćen je samo MFCC atribut - utvrđeno je da nema potrebe za drugim atributima.

5 Modeli

5.1 FFNN

Model učitava 57 atributa, koji dalje prolaze kroz 2 skrivena sloja, veličine 25 i 20, respektivno, gde se poslednji skriveni sloj spaja sa Softmax aktivacionom funkcijom, davajući predikciju klase.

5.2 CNN

Model učitava trodimenzionalni niz, MFCC, sa izvučenih 52 atributa, koji dalje prolaze kroz 6 konvoluciona i 2 skrivena sloja. Model je napravljen kao skraćena varijanta VGG-11, koja je bila prevelika da se istrenira od nule.

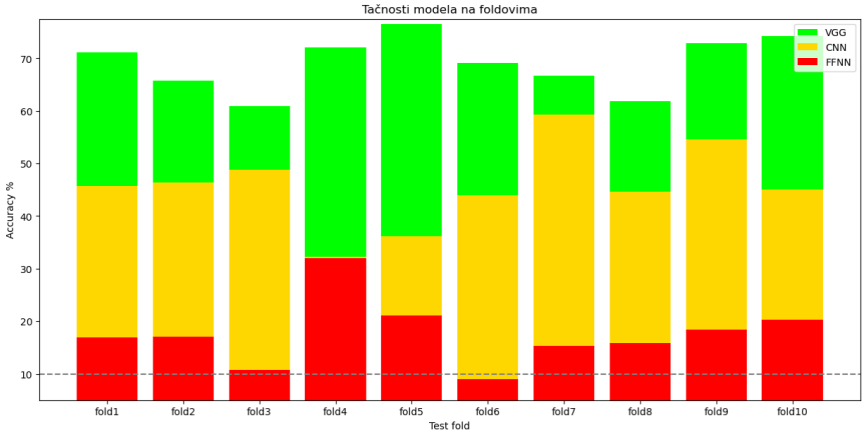
5.3 Trening

Treniranje je rađeno na 10 epoha. Adam[5] optimizacioni algoritam se pokazao najbolje, davajući najbolji rezultat sa stopom učenja 10^{-4} u prvih 5 epoha i 10^{-5} u poslednjih 5. Funkcija gubitka je definisana kao kategorička kros-entropija.

6 Hipoteze

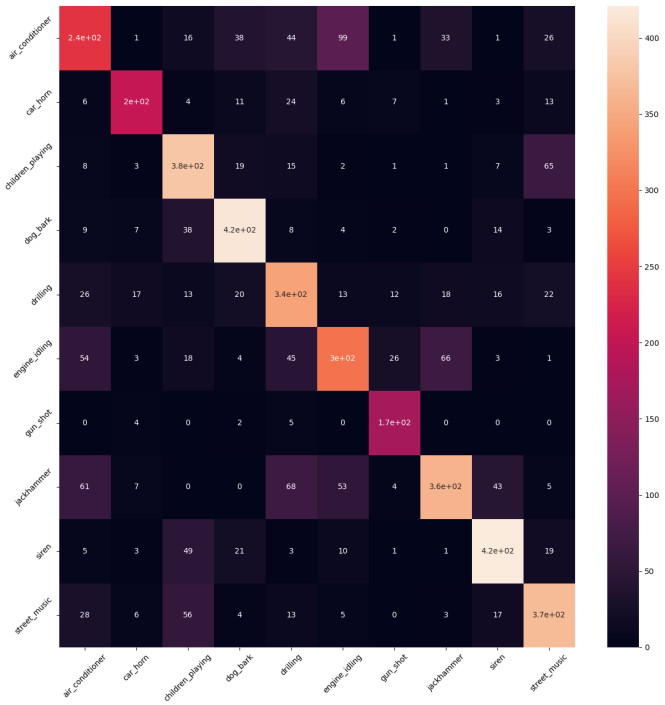
- Usled skraćivanja podataka za FFNN, baš taj model će imati najlošiji rezultat
- Pretrenirani VGG-11 će se pokazati najbolje zbog parametara koji se samo trebaju dotrenirati da budu adaptirani datom skupu podataka
- Ispisana CNN mreža će dati kompetitivan rezultat u odnosu na pretrenirani VGG-11

7 Rezultati



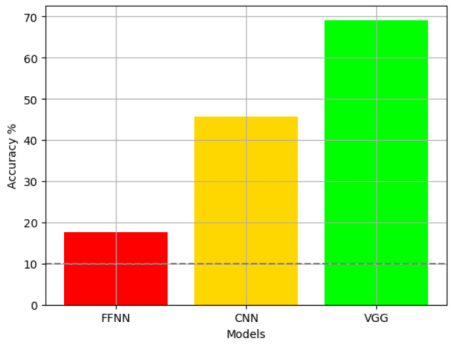
Slika 2. Tačnosti modela po foldovima

Pretrenirani VGG-11 se, kao i po intuciji, pokazao najbolje, dominira nad druga dva modela kroz svaki fold, dajući prosečni rezultat od 69.14%.



Slika 3. Matrica konfuzije za jedan fold VGG-a

Dok je prosečna tačnost FFNN-a 17.67%, a CNN-a 45.68%, što je bilo očekivano, usled malog broja slojeva za oba modela, a za FFNN i mali broj ulaznih atributa.

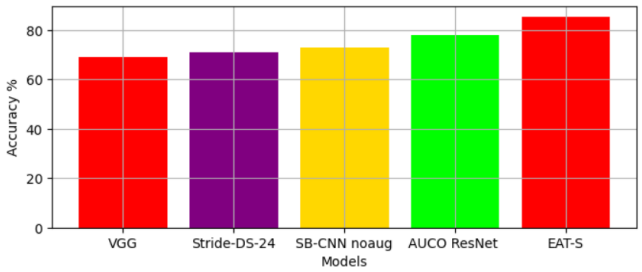


Slika 4. Tačnosti modela

Prema uputstvu autora skupa podataka, potrebno je takođe bilo i izračunati standardno odstupanje na 10 foldova za svaki model, a standardna odstupanja naših modela su:

- 5.98 za FFNN
- 7.41 za CNN
- 4.93 za VGG

Takođe, zahvaljujući postojanju javnih članaka i radova mogli smo da proverimo kako su se naši modeli rangirali, pa smo odlučili da uporedimo VGG sa njima.



Slika 5. VGG naspram drugih javno dostupnih modela

VGG se rangirao 5. u odnosu na javno dostupne modele na sajtu *Papers with code*, što se može i videti na slici 5. Pomenuti modeli takođe nisu koristili dodatne podatke za trening.

Poredili smo sa:

- EAT-S [6]
- AUCO ResNet - NO TRANSFER / NO DATA AUGMENTATION [7]
- SB-CNN noaug [8]
- Stride-DS-24 [9]

8 Diskusija

Svrha odabira ovog projekta je bila unapređivanje našeg znanja iz domena mašinskog učenja i korišćenja PyTorch biblioteke kako implementacijom neuralnih mreža od nule tako i poređenjem naših modela sa postojećim koji dominiraju u industriji. Sva 3 modela pokazuju znake da su naučili iz naših atributa, zaključujući bolje od nasumičnog nagadjanja.

Gledajući hipoteze, jasno se može primetiti da FFNN daje najmanju tačnost, potvrđujući našu intuiciju o pojedonstavljenju modela i atributa. S druge strane, hipoteza da će CNN dati kompetitivan rezultat sa VGG-om se nije ispostavila tačnom, što opravdavamo kako jednostavnijim modelom, tako i činjenicom da nije pretreniran, odnosno da već nije imao određenu moć rezonovanja atributa. Navedenim se zaključuje da pretrenirani VGG-11 odnosi čistu pobedu kada je tačnost u pitanju.

9 Dodaci

Korisnik može pokrenuti projekat koristeći terminal, birajući različite opcije kao što su npr. model s kojim se trenira, da li želi samo testirati, da li želi sačuvati rezultate itd. Više informacija može se naći u README.md fajlu

Dodata je funkcija klasifikacije audio fajla koji korisnik manuelno može uneti. Rezultat se određuje na osnovu sistema glasanja učitanih 10 modela zaduženih za određivanje klase.

10 Literatura

[1] Very Deep Convolutional Networks for Large-Scale Image Recognition, Karen Simonyan, Andrew Zisserman

[2] Neural network based recognition of speech using MFCC features, Pialy Barua,Kanij Ahmad, Ainul Anam Shahjamal Khan, Muhammad Sanaullah (2014)

[3] Justin Salamon, Christopher Jacoby and Juan Pablo Bello Music and Audio Research Laboratory (MARL), New York University Center for Urban Science and Progress (CUSP), New York University

[4] Douglas O'Shaughnessy (1987). Speech communication: human and machine. Addison-Wesley. str. 150.

[5] Adam: A Method for Stochastic Optimization, Diederik P. Kingma, Jimmy Ba

[6] End-to-End Audio Strikes Back: Boosting Augmentations Towards An Efficient Audio Classification Network, Avi Gazneli, Gadi Zimerman, Tal Ridnik, Gilad Sharir, Asaf Noy

[7] AUCO ResNet: an end-to-end network for Covid-19 pre-screening from cough and breath, Vincenzo Dentamaro, Paolo Giglio, Donato Impedovo, Luigi Moretti, Giuseppe Pirlo

[8] Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification, Justin Salamon, Juan Pablo Bello

[9] Environmental Sound Classification on Microcontrollers using Convolutional Neural Networks, Jon Nordby

[10] Min Xu; et al. (2004). "HMM-based audio keyword generation"

[11] J. Salamon, C. Jacoby and J. P. Bello, "A Dataset and Taxonomy for Urban Sound Research", 22nd ACM International Conference on Multimedia, Orlando USA, Nov. 2014.