

# Klasifikacija produkata na self-checkout kasi

Adisa Bolić

Prirodno-matematički fakultet

Univerzitet u Sarajevu

Sarajevo, Bosna i Hercegovina

adisa.bolic@gmail.com

**Sažetak**—Dokoro većina self-checkout kasa u prodavnicama je radila na principu da kupac pomoću skenera skenira barkodove svih produkata i na taj način izvrši kupovinu. Današnja revolucija u polju mašinskog učenja omogućava da se ovaj proces ubrza na način da korisnik samo postavi sve produkte na neku površinu (iznad koje se nalazi kamera) i da se potom automatski detektuje koji se proizvodi tu nalaze. Ovaj rad ima za cilj izvršenje dijela ovog procesa, tj. klasifikaciju produkata koristeći konvolucijsku neuralnu mrežu. Kod je dostupan na [GitHub repozitoriju](#). Za implementaciju korišten je jezik Python te biblioteka PyTorch za sve elemente mašinskog učenja.

**Index Terms**—self-checkout, convolutional neural network, classification, machine learning

## I. UVOD

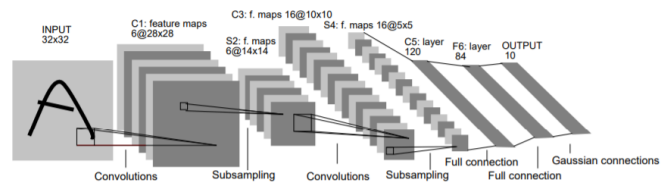
Kako bi se ubrzao i pojednostavio proces kupovine u supermarketima ili restoranima u ovom radu će biti prikazan algoritam koji vrši klasifikaciju produkata na osnovu njihovih slika. Za algoritam je korištena konvolucijska neuralna mreža ResNet50 (detaljnije o arhitekturi ove mreže na [1]) koja je pretrenirana na ImageNet skupu podataka. ImageNet [2] je označen skup slika sa približno 20,000 kategorija sa nekoliko stotina slika u svakoj kategoriji. Na slici 1 prikazano je nekoliko slika iz ovog skupa. ILSVRC [3] (ImageNet Large Scale Visual Recognition Challenge) je takmičenje koje se godišnje održavalo u kojem je bio cilj napraviti model koji će imati što veću preciznost na ovom skupu slika. Ovo takmičenje predstavlja standard za procjenu koliko je neki model dobar. Pored toga, modeli koji su pretrenirani na ImageNet skupu se koriste (u smislu da se kopiraju težine naučene u toj pretreniranoj mreži) kada se želi trenirati model za neki drugi problem, pri čemu se izvrše određene izmjene na mreži (uglavnom se promijene dimenzije zadnjeg sloja mreže da bi odgovaralo broju kategorija u problemu koji se rješava).

U radu [4] navedeno je da se konvolucijske neuralne mreže mogu opisati kao nadogradnja nad običnim višeslojnim unaprijednim mrežama. Konvolucijska, kao i obična neuralna mreža sastoji se od jednog ulaznog, jednog izlaznog i jednog ili više skrivenih slojeva. Kod konvolucijskih neuralnih mreža specifični su konvolucijski slojevi i slojevi sažimanja. Osim njih često se koriste i potpuno povezani slojevi. Konvolucijske neuronske mreže najčešće kreću s jednim ili više konvolucijskih slojeva, zatim slijedi sloj sažimanja, pa ponovo konvolucijski sloj i tako nekoliko puta. Mreža najčešće završava s jednim ili više potpuno povezanih slojeva koji služe za klasifikaciju. Arhitektura konvolucijskih neuronskih mreža pokazala se izrazito dobra u radu sa slikama i prepoznavanju



Slika 1: Primjer slika iz ImageNet skupa slika. [Izvor slike](#)

objekata od interesa sa istih. Na slici 2 prikazana je arhitektura jedne od prvih konvolucijskih neuralnih mreža poznate pod imenom LeNet [5]. Danas postoje mnoge modifikacije ovih mreža koje su se pokazale jako uspješne za razne probleme klasifikacije i detekcije objekata. Detaljnije o ovim mrežama kao i ostalim poljima mašinskog učenja koji su doveli do kreiranja konvolucijskih mreža može se pročitati u radu [6].



Slika 2: Arhitektura LeNet [5] konvolucijske neuralne mreže. [Izvor slike](#) [5]

U problemu koji se rješava u ovom radu dat je skup od približno 45,000 slika sa ukupno 76 klasa produkata. Slike su podijeljene na tri dijela: trening, validacijski i testni skup.

Trening i validacijski skup se koriste za treniranje (validacijski skup se koristi za smanjenje brzine učenja ukoliko se preciznost smanji nakon neke epohe sa ciljem izbjegavanja overfittinga na trening skupu, kao i za odabir najboljeg modela), dok se testni skup koristi nakon treniranja za provjeru preciznosti modela. Primjer tri slike iz skupa dat je na slici 3.



Slika 3: Primjer tri slike iz trening skupa slika. Klase ovih slika su, redom, kiflica, flaširana voda i čokoladica.

## II. PREGLED LITERATURE

Proces kupovine u supermarketima se može i dodatno ubrzati. Poznat primjer ovog ubrzanja prisutan je Amazonovim Go prodavnicama, gdje kupci samo pakuje artikle koje žele sa police i izađu iz prodavnice, a preko aplikacije im se automatski obračuna cijena kupovine. U prodavnici je prisutan veliki broj kamera koje prate sve što kupac radi i pomoću algoritma mašinskog učenja se detektuje kada kupac nešto uzme sa police sa ciljem da to nešto i kupi. Znatno je teže dizajnirati model za ovaj problem zbog mnogobrojnih situacija koje mogu nastati. Detalje i poteškoće kod ovog problema mogu se pročitati u radu [7].

## III. OPIS MODELA

Kao što je ranije spomenuto, korištena je pretrenirana ResNet50 arhitektura konvolucijske mreže. Bitni parametri kod treniranja su:

- batch veličina = 128
- broj epoha = 20
- broj klasa = 76
- veličina trening skupa = 33,991
- veličina validacijskog skupa = 8456
- veličina testnog skupa = 2200
- korištene transformacije i augmentacije: normalizacija, horizontalni i vertikalni flip, rotacija
- dimenzije slika:  $224 \times 224$
- loss funkcija: Cross-entropy loss

Trenirana su dva modela. U prvom modelu su trenirani svi parametri mreže, dok su u drugom modelu trenirane samo težine u zadnjem fully-connected sloju mreže. Oba modela trenirana su sa istim hiperparametrima. Trening je na tri Nvidia GeForce GTX 1080 grafičke trajao oko 30 i 17 minuta, redom. Pošto su sve slike sa dobrim osvijetljenjem i dobre kvalitete, za očekivati je da će mreža vršiti dobre predikcije. Prvi model bi trebao davati bolje rezultate, jer se trenira čitava mreža, a ne samo zadnji sloj.

Na slici 4 dato je nekoliko primjera slika iz skupa nakon primijenjenih transformacija.



Slika 4: Primjer nekoliko slika iz trening skupa nakon primijenjenih transformacija. Boja slika se promijenila uslijed normalizacije.

## IV. REZULTATI

Na slici 5 prikazani su rezultati tokom treniranja prvog modela u prvim i zadnjih nekoliko epoha. Slično, na slici 6 prikazani su rezultati tokom treniranja drugog modela. Treba primijetiti kako je pretpostavka da će prvi model davati bolje rezultate tačna. Međutim, potrebno je provjeriti preciznost oba modela na testnom skupu, pošto je moglo doći do overfitanja na trening skupu. Rezultati na testnom skupu oba modela dati su na slici 7.

Epoha 1/20	
train Loss: 0.4517 Acc: 0.8790	
val Loss: 0.2035 Acc: 0.9354	
Epoha 2/20	
train Loss: 0.1339 Acc: 0.9588	
val Loss: 0.2554 Acc: 0.9372	
Epoha 3/20	
train Loss: 0.0998 Acc: 0.9709	
val Loss: 0.1250 Acc: 0.9613	
Epoha 4/20	
train Loss: 0.0738 Acc: 0.9776	
val Loss: 0.0502 Acc: 0.9834	
Epoha 5/20	
train Loss: 0.0608 Acc: 0.9811	
val Loss: 0.0774 Acc: 0.9762	
Epoha 6/20	
train Loss: 0.0479 Acc: 0.9858	
val Loss: 0.0502 Acc: 0.9849	
Epoha 7/20	
train Loss: 0.0551 Acc: 0.9840	
val Loss: 0.0575 Acc: 0.9834	
Epoha 15/20	
train Loss: 0.0230 Acc: 0.9921	
val Loss: 0.0227 Acc: 0.9929	
Epoha 16/20	
train Loss: 0.0236 Acc: 0.9928	
val Loss: 0.0813 Acc: 0.9825	
Epoha 17/20	
train Loss: 0.0238 Acc: 0.9931	
val Loss: 0.0173 Acc: 0.9955	
Epoha 18/20	
train Loss: 0.0100 Acc: 0.9968	
val Loss: 0.0152 Acc: 0.9956	
Epoha 19/20	
train Loss: 0.0171 Acc: 0.9946	
val Loss: 0.0211 Acc: 0.9947	
Epoha 20/20	
train Loss: 0.0169 Acc: 0.9949	
val Loss: 0.0183 Acc: 0.9946	
Trening završen za: 31m 37s	
Najbolja preciznost (Acc) na validacijskom skupu: 0.995624	

Slika 5: Preciznost i vrijednost loss funkcije (na trening i validacijskom skupu) prvog modela (u kojem se svi parametri mreže treniraju) tokom prvih i zadnjih par epoha.

Pošto su rezultati na testnom skupu slični rezultatima tokom treniranja, znači da nije došlo do overfitanja i model radi ispravno sa velikom preciznošću. Mnogo više detalja i vizualizacija dostupno je u Jupyter Notebooku sa implementiranim kodom na [GitHub respositoriju](#).

## V. ZAKLJUČAK

U ovom radu kreiran je model konvolucijske neuralne mreže za klasifikaciju produkata u restoranu na ukupno 76 klasa. Korištena je pretrenirana (na ImageNet [2] skupu slika) ResNet50 [1] mreža. Postignuta je velika preciznost mreže od 99.45% na testnom skupu slika. U budućnosti je planirano napraviti

```

Parametri modela za treniranje:
  module.fc.weight
  module.fc.bias
Epoha 1/20
-----
train Loss: 0.7119 Acc: 0.8549
val Loss: 0.2642 Acc: 0.9519

Epoha 2/20
-----
train Loss: 0.1688 Acc: 0.9653
val Loss: 0.1489 Acc: 0.9677

Epoha 3/20
-----
train Loss: 0.1140 Acc: 0.9743
val Loss: 0.1381 Acc: 0.9667

Epoha 4/20
-----
train Loss: 0.0884 Acc: 0.9798
val Loss: 0.0998 Acc: 0.9774

Epoha 5/20
-----
train Loss: 0.0773 Acc: 0.9805
val Loss: 0.0954 Acc: 0.9766

Epoha 16/20
-----
train Loss: 0.0333 Acc: 0.9905
val Loss: 0.0449 Acc: 0.9873

Epoha 17/20
-----
train Loss: 0.0328 Acc: 0.9910
val Loss: 0.0398 Acc: 0.9875

Epoha 18/20
-----
train Loss: 0.0325 Acc: 0.9900
val Loss: 0.0463 Acc: 0.9870

Epoha 19/20
-----
train Loss: 0.0315 Acc: 0.9912
val Loss: 0.0414 Acc: 0.9871

Epoha 20/20
-----
train Loss: 0.0301 Acc: 0.9914
val Loss: 0.0398 Acc: 0.9877

Trening završen za: 17m 51s
Najbolja preciznost (Acc) na validacijskom skupu: 0.987701

```

Slika 6: Preciznost i vrijednost loss funkcije (na trening i validacijskom skupu) drugog modela (u kojem se samo parametri zadnjeg sloja mreže treniraju) tokom prvih i zadnjih par epoha.

```

Preciznost modela na 2200 testnih slika: 99.45 %
Prosječni inference time: 360 mikrosekundi
Broj pogrešnih predikcija: 12 / 2200

```

```

Preciznost modela na 2200 testnih slika: 98.55 %
Prosječni inference time: 353 mikrosekundi
Broj pogrešnih predikcija: 32 / 2200

```

Slika 7: Preciznost prvog (gore) i drugog (dole) modela na testnom skupu slika. Inference time predstavlja vrijeme koje je potrebno modelu za klasifikaciju jedne slike.

model koji će vršiti detekciju objekata (da iz slike cijeje scene detektuje sve produkte, lokalizira ih i potom klasificira).

## VI. REFERENCE

- [1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition, 2015
- [2] ImageNet skup slika <https://www.image-net.org/about.php>
- [3] Olga Russakovsky\*, Jia Deng\*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (\* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015
- [4] Damir Kopljar, Konvolucijske neuronske mreže, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, 2016
- [5] Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-Based Learning Applied to Document Recognition, 1988
- [6] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016
- [7] Kirti Wankhede, Bharati Wukkadada, Vidhya Nadar, Just Walk-Out Technology and its Challenges: A Case of Amazon Go, IEEE, 2018