SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 000

Prepoznavanje emocija iz izraza lica pomoću strojnog učenja

Matej Ciglenečki

Zagreb, lipanj 2020.

Umjesto ove stranice umetnite izvornik Vašeg rada.
Da bi ste uklonili ovu stranicu obrišite naredbu `\izvornik`.

SADRŽAJ

1. Uvod

Poreba za prepoznavanjem emocija **što emocije govore** **gdje se koristi prepozavanje emocija** **korištena metoda za klasifikaciju emcoija**

2. Podatkovni skup

2.1. Uvod

Podatkovni skup sastavni je dio u izvedbi treniranja modela. Treniranje se svodi na ulazne i izlazne podatke gdje su ulazni podaci podskup podatkovnog skupa. Korišteni podatkovni skupovi su Cohn-Kanade (CK) i skup slika preuzetih sa Google-a na temelju ključne riječi. Podatkovni skup dijelimo na dva djela. Skup za treniranje i skup za testiranje. Svi podaci koji su u skupu za treniranje iskorištavaju se za treniranje i optimiziranje odabranog modela dok se skup za testiranje koristi samo za evaluaciju točnosti treniranog modela. 80% nasumičnih slika odabrane su za treniranje a ostalih 20% koristi se za evaluaciju.

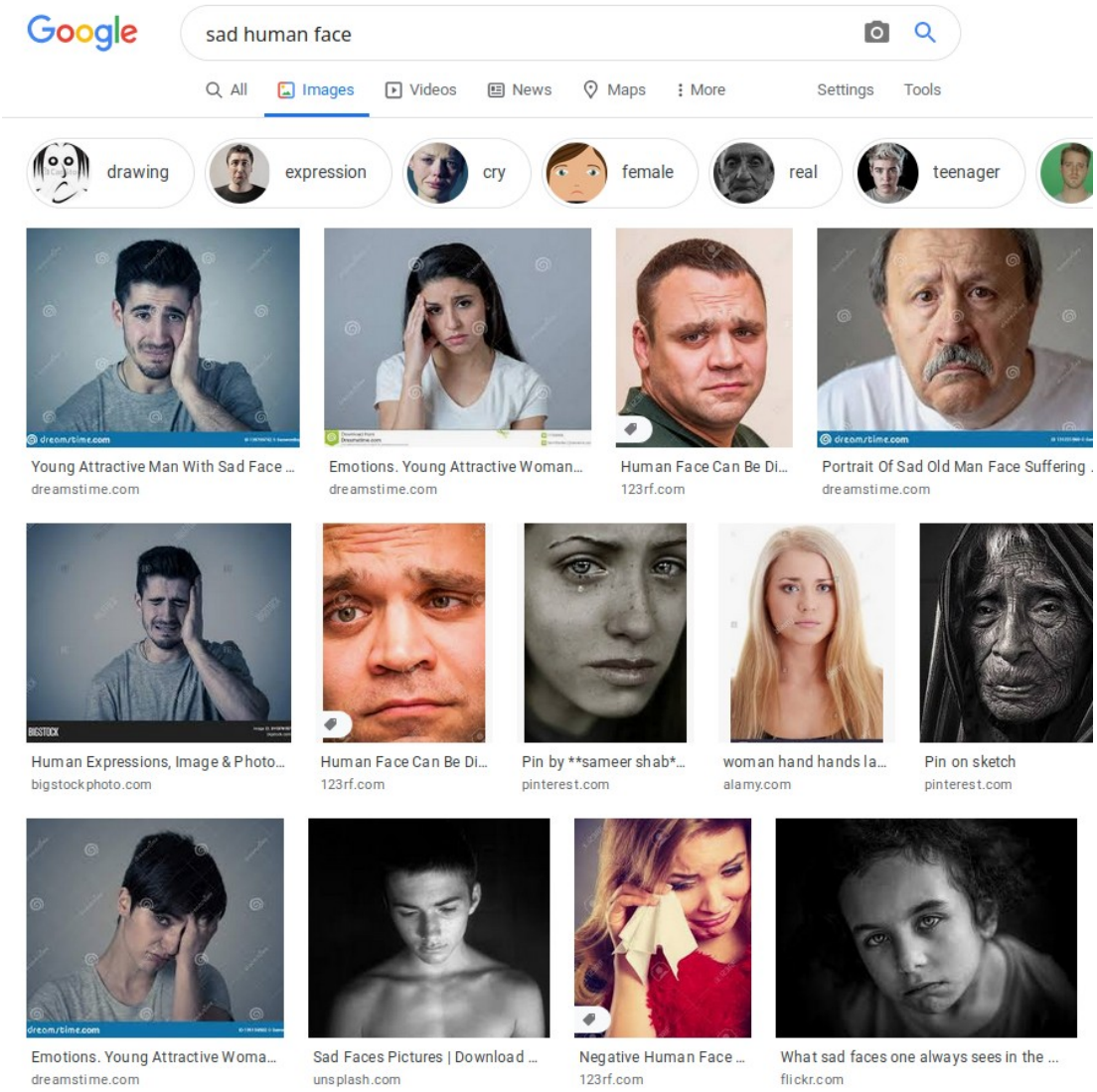
2.2. Extended Cohn-Kanade podatkovni skup

Cohn-Kanade podatkovni skup sastoji se od 593 sekvenci slika od 123 subjekta (osoba). Pojedina sekvenca sastoji se od 10 do 60 slika. Početna slika u sekvenci je neutralna emocija dok je zadnja slika vrhunac izraza emocije. Subjekti na slikama imaju od 18 do 50 godina, 69% su žene, 81% euro-Amerikanci i 6% su subjekti ostalih rase. Rezolucija pojedine slike iznosi 640x480 ili 640x490 piksela u 8-bitnom crno-bijelom ili 24 bitnom puno-bojnom formatu[2]. TODO: primjer slike CK+ dataseta

2.3. Ručno generirani podatkovni skup

Slike CK+ podatkovnog skupa slikane su u istom okruženju (ista prostorija u kojoj se slikaju subjekti, ista kamera, slična svjetlina slike...). Nedostatak raznolikosti među slikama stvara potrebu za uvođenjem apstraktnijih slika ljudskih lica ne bi li model klasificirao emocije ljudskih lica koja nisu slična samo CK+ podatkovnom skupu. Zbog toga se uvodi podatkovni skup Google slika. Google omogućuje pretraživanje slika

po zadanom upitu te dodatnim parametrima koji olakšavaju pronalaženje ljudskog lica koji predstavlja određenu emociju. Npr. pronalazak ljudskih lica koji iskazuju tužnu emociju mogao bi biti "sad human face" sa tipom Google slike "lice". Rezultati tog upita prikazani su na slici 2.1.



Slika 2.1: Rezultati Google upita "sad human face"

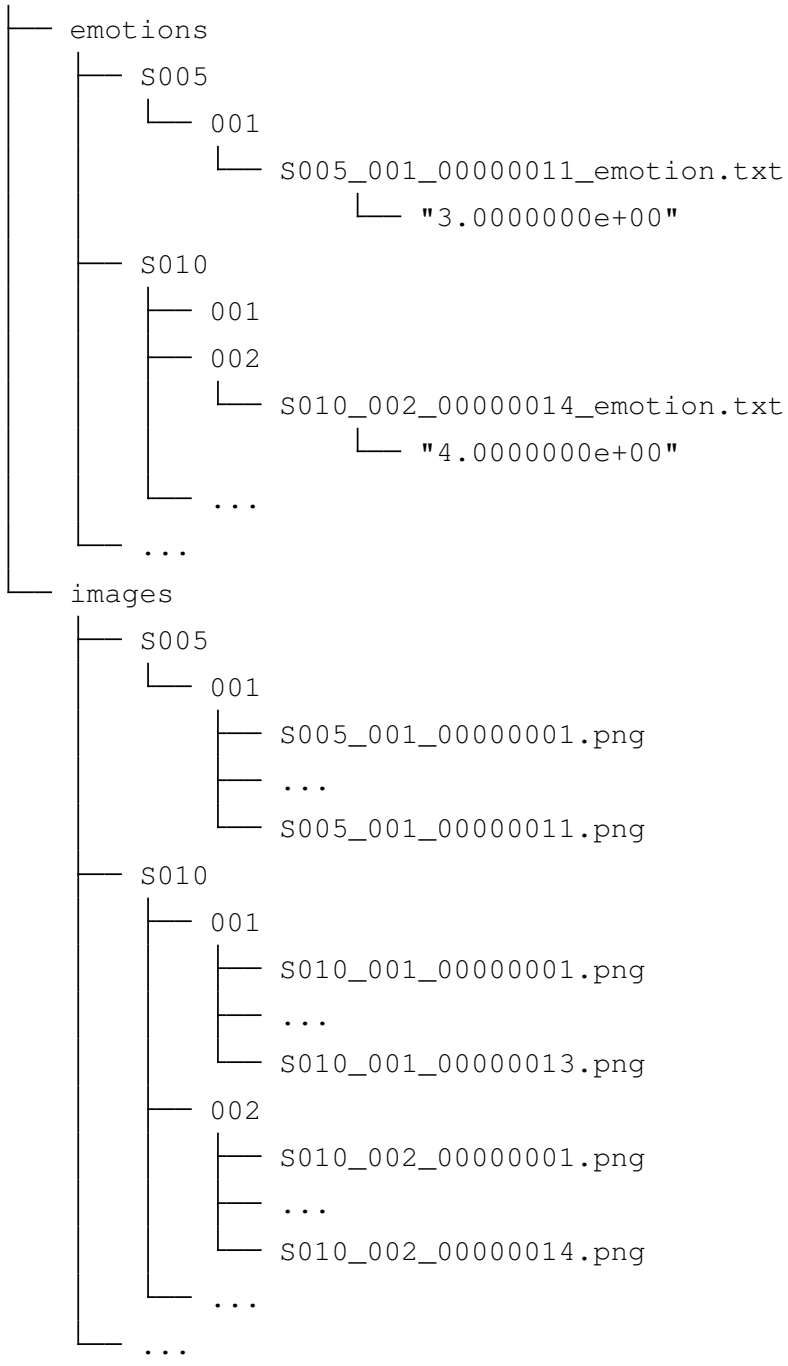
Rezultati ovog upita su slike veće raznolikosti (različit scenarij i drugačiji kut gledanja) što pridonosi apstrakciji emocije u ukupnom podatkovnom skupu. Dobivene slike ne odgovaraju uvijek nužno zadanom upitu zbog čega je potrebno dodatno provjeriti valjanost pojedine slike. Proces filtriranja objašnjen je u poglavlju 2.4.2

2.4. Priprema podatkovnih skupova

2.4.1. Priprema Cohn-Kanade podatkovnog skupa

Struktura podataka

Dijelovi podatkovnog skupa značajni za treniranje dijele se na direktorij sekvence i emocije. Svaki subjekt (npr. S005) ima svoj direktoriji u kojem se nalaze pod direktoriji za emociju koju je subjekt odglumio (npr. 001, 002...) a u njemu se nalaze sekvence. Za 327 sekvenca postoji odgovarajuća emocija koja je definirana samo za krajnju sliku sekvence i njezina putanja je definirana jednako kao i za sekvencu.



Slika 2.2: Struktura podataka CK+ podatkovnog skupa

Obrada podataka

Jedina slika u sekvenci za koju je definirana emocija je zadnja slika, što znači da je potrebno je potrebno odrediti vektor emocije ostalih slika u sekvenci na temelju krajnje vrijednosti emocije. Emocija za pojedinu sliku definirana je kao red emocija. Svaki in-

deks reda predstavlja emociju kojih ima kojih ima 8. Indeks pojedine emocije definiran je na slici 2.3 a vrijednost na indeksu predstavlja intenzitet emocije

```
emocije = {
    1: "neutral",
    2: "anger",
    3: "contempt",
    4: "disgust",
    5: "fear",
    6: "happy",
    7: "sadness",
    8: "surprise",
}
```

Slika 2.3: Deklaracija emocija

Znajući da je emocija za početnu sliku neutralna a za krajnju maksimalna sekvencijska emocija, emocije za ostale slike dodijeljene su linearno na način da se odredi intenzitet neutralne emocije 2.1 i sekvencijske emocije 2.2 gdje je n ukupan broj slika u sekvenci a i slika kojoj se određuje vektor emocije.

$$p_n = \frac{i-1}{n-1} \tag{2.1}$$

$$\begin{aligned} p_s &= 1 - p_n \\ p_s + p_n &= 1 \end{aligned} \tag{2.2}$$

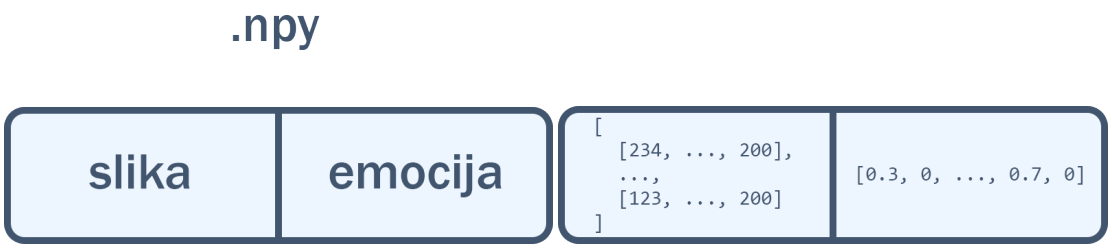
Primjer vektora emocije za sekvencu "disgust" koja sadrži 10 slika:

broj slike	vektor emocije
$i = 1$	[1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
...	...
$i = 6$	[0.4, 0.0, 0.0, 0.6, 0.0, 0.0, 0.0, 0.0]
...	...
$i = n = 10$	[0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 0.0]
i	[p_n, 0.0, 0.0, p_s, 0.0, 0.0, 0.0, 0.0]

Slika 2.4: Vektor emocije za pojedinu pojedinu sliku u sekvneci u CK podatkovnom skupu

Spremanje slike i vektora emocije u .npy podatak

Za daljnje korištenje svaka slika i njezin vektor emocije će pretvorena u numpy red [1] i spremljen kao .npy podatak. Prvi element numpy reda je slika a drugi je odgovarajući vektor emocije. Ovime je osigurano da izračun emocija za pojedinu sliku je izračunat samo jedanput što će smanjiti vrijeme potrebno za treniranje. Nakon provođenja transformacije podataka u .npy podatak ukupan broj slika i pripadajućih vektora emocija iznosi 5703.



(a) Struktura .npy podatka **(b)** Sadržaj .npy podatka

Slika 2.5: .npy podatak

2.4.2. Priprema Google podatkovnog skupa

Prikupljeni podaci

Umjesto preuzimanja jedne po jedne slike korištena je google-images-download skripta [4] koja preuzima sve moguće slike na temelju zadanog upita. Upiti korišteni za pronalaženje odgovarajućih emocija nalaze se na slici 2.6 a svi su popraćeni dodatnim "Google search" parametrom koji pretražuje slike samo ljudskih lica. Svaka slika spremljena je u direktoriji čije je ime upit koji je bio korišten prilikom preuzimanje te slike.

```
"sad human face"
"neutral human face"
"neutral expression"
"angry human face"
"angry expression"
"contempt"
"fear human face"
"fear expression"
"surprise human face"
"surprise expression"
"disgusted human face"
"disgust expression"
```

Slika 2.6: Upiti za preuzimanje ljudskih lica sa Google-a

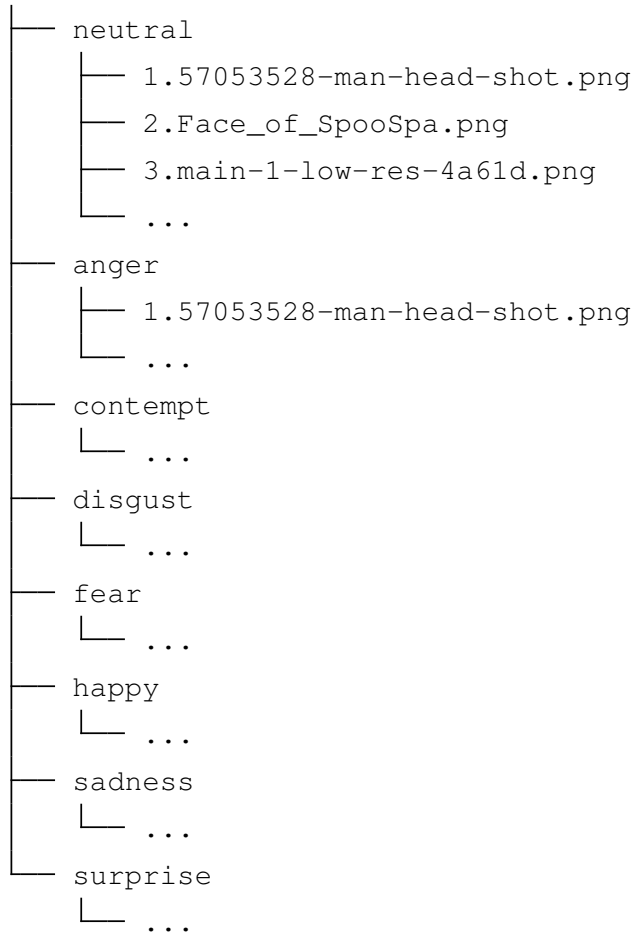
Nakon preuzimanja svih mogućih slika potrebno je ručno proći kroz svaki direktorij svakog upita i izbaciti slike koje ne zadovoljavaju sljedeće uvjete

- Na slici se nalazi samo jedno ljudsko lice
- Na slici se nalazi ljudsko lice čija emocija odgovara upitu pomoću kojeg je slika preuzeta
- Veličina slike je manja od 20MB
- Rezolucija slike je veća od 20px po duljini i visini
- Slika nije duplikat prethodno viđene slike
- Slika nije dio CK+ podatkovnog skupa

Uklanjanjem slika koje ne zadovoljavaju bilo koje od navedenih uvjeta dobiven ukupan broj slika povoljnih za treniranje iznosi 3160 a njihova ukupna veličina je 2,3GB.

Struktura podataka

Nakon filtriranja slika nepogodnih za treniranje potrebno je objediniti upite čiji su rezultati ljudska lica efektivno istih emocija. Primjer takva dva upita su "neutral human face" i "neutral expression". Nakon objedinjavanja rezultata upita i preimenovanja direktorija stvorena je struktura podataka dana na slici 2.7



Slika 2.7: Struktura podataka Google podatkovnog skupa

Obrada podataka

Obrada Google podatkovnog skupa bit će manje zahtjevna od CK+ podatkovnog skupa zbog toga što će se vektor emocije za pojedinu sliku odrediti samo na temelju upita korištenog za preuzimanje slike. Rezultat svakog vektora emocije bit će vektor dobiven metodom "One hot encoding". "One hot encoding" je metoda dodjele binarne vrijednosti za kategoriju u koju neki uzorak pripada ili ne pripada[3]. Ovom metodom uzorak (slika) može pripadati samo jednoj kategoriji (emocija). Slika koja pripada određenoj emociji za tu će emociju imati vrijednost 1 a za sve ostale 0. Na slici 2.8 prva kolumna označava ime direktorija u kojem se slike nalaze, druga kolumna predstavlja vektor emocije koje će slike u direktoriju poprimiti.

emocija (ime direktorija)	vektor emocije
neutral	[1,0,0,0,0,0,0,0]
anger	[0,1,0,0,0,0,0,0]
contempt	[0,0,1,0,0,0,0,0]
disgust	[0,0,0,1,0,0,0,0]
fear	[0,0,0,0,1,0,0,0]
happy	[0,0,0,0,0,1,0,0]
sadness	[0,0,0,0,0,0,1,0]
surprise	[0,0,0,0,0,0,0,1]

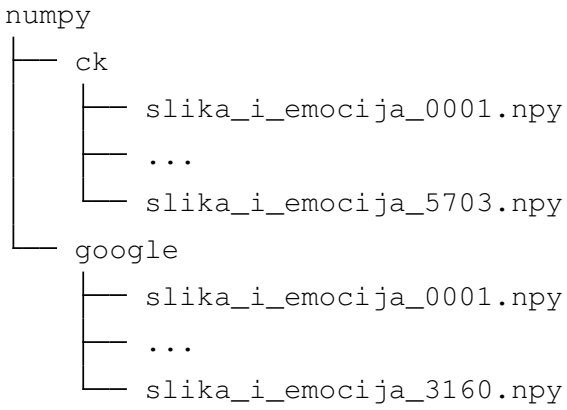
Slika 2.8: Vektor emocije za pojedinu emociju u Google podatkovnom skupu

Spremanje slike i vektora emocije u .npy podatak

Nakon dodjele vektora emocije za pojedinu sliku potrebno je pretvoriti sliku i vektor emocije u .npy podatak. Kako se radi o slici i vektoru emocije, postupak pretvorbe slike i vektora emocije u pojedinačni .npy podatak jednak je kao i kod CK+ podatkovnog skupa definiranom u poglavlju 2.4.1.

2.4.3. Objedinjavanje podatkovnih skupova

Nakon obrade CK+ i Google podatkovnog skupa svi .npy podaci bit će spremljeni u direktoriji "ck" ili "Google" ovisno o tome iz kojeg je podatkovnog skupa slika dobivena. Struktura svih .npy podataka prikazana je na slici 2.9. Ovom strukturom moguće je definirati udio svakog podatkovnog skupa koji će biti korišten za treniranje modela.



Slika 2.9: Struktura .npy podataka

3. Treniranje

Treniranje je proces u kojem model postupno mijenja svoje parametre ne bi li došao do idealnih parametara, čime bi model optimalno služio onome čemu je namijenjen. U slučaju klasifikacije za treniranje je potrebno imati skup podataka nad kojim će se provoditi treniranje i labele tih podataka koje govore klasu podataka. U slučaju klasifikacije emocija, skup podataka nad kojim se trenira su slike ljudskih lica dok je klasa emocija koja je prikazana na pojedinoj sliku. Kad treniranje započne, u model se pošalju podaci za koje model pokušava predvidjeti koje su klase ubačeni podaci. Nakon toga potrebno je usporediti predviđanja klasa koje je stvorio model sa pravim klasa. Pogrešku koju je model napravio prilikom predviđanja potrebno je javiti modelu ne bi li ispravio parametre. Mijenjanjem modela parametra model postaje sve bolji za predviđanje predviđanje rezultata na skupu za treniranje. Bitno je naglasiti da daljnjim treniranjem model ne postaje bolji za općeniti slučaj predviđanja. Treniranjem model se prilagođava podatkovnom skupu za treniranje. Podatkovni skup za treniranje je podskup svih podataka za koje je model namijenjen a to znači da skup za treniranje nije nužno najbolja mjera općenitog slučaja za podatke.

Treniranje modela nakon određenog trenutka može pogoršati sposobnost modela da predviđa klase na općenitom slučaju. Uzrok toga je što treniranjem nakon točke TODO:NAME model ima veću točnost na skupu za treniranje ali točnost na općenitim, neviđenim slučajevima postaje sve manja. Zbog toga je potrebno pronaći optimalan trenutak u kojem treba prekinuti treniranje modela. Prekidanje treniranja modela nije moguće odrediti pomoću točnosti modela na skupu za treniranje jer ta točnost neprestano raste daljnjim treniranjem. Zbog toga je potrebno uvesti validacijski skup koji neće biti korišten prilikom treniranja. Validacijski skup o tom slučaju igra ulogu neviđenih podataka nad kojima se mjeri točnost. Prilikom treniranja točnost nad validacijskim skupom će rasti do točke TODO:ANME. U tom trenutku točnost nad validacijskim skupom će biti maksimalna a daljnjim treniranjem točnost će se smanjivati jer model postaje previše prilagođen skupu za treniranje (engl. *overfitting*).

TODO: slika - skup podataka **načini treniranja**

3.1. Duboke neuronske mreže za analizu slika

3.1.1. Neuron

Neuron je osnovni dio neuronske mreže. Svaki neuron na svoj ulaz prima vektor x vrijednosti za koje neuron predviđa $\hat{y}$ izlaz. Skup x vrijednosti je također pomnožen sa vektorom w koji predstavlja težine (engl. *weights*) i zbrojen sa vrijednošću b koja predstavlja sklonost (engl. *bias*)

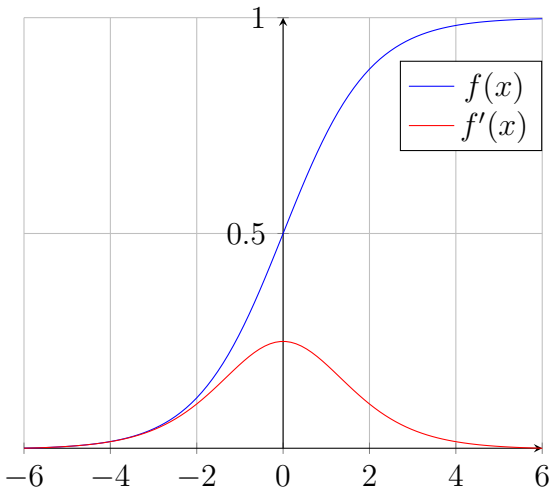
$$\boldsymbol{z} = w_1 x_1 + w_2 x_2 + ... + w_n x_n = \boldsymbol{w^T} \cdot \boldsymbol{x} \tag{3.1}$$

$$\hat{y} = g(vectz) \tag{3.2}$$

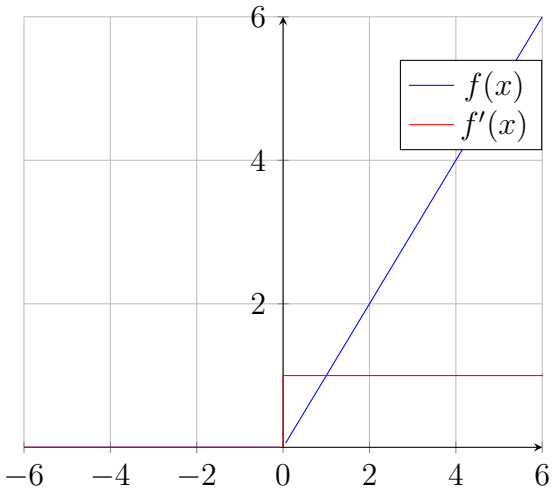
Aktivacijska funckija

Nakon zbrajanja kroz neuron izlaz je potrebno provesti kroz aktivacijsku funkciju određuje koliko je signal tog neurona bio zastupljen u ukupnom izlazu, tj. koliko je "aktiviran". Cilj koji se postiže tom funkcijom je da ukupan izlaz neurona bude između vrijednosti 0 i 1. Dakle to su funkcije za koje vrijedi 3.3. Primjer takvih funckija je sigmoid prikazan na slici 3.1 i rektifikacijska funckija *(ReLU - engl. Rectifier)* prikazana na slici 3.2. Prilikom aktivacije neurona koristiti će se rektifikacijska funckija

$$f : \mathbb{R} \to [0, 1] \tag{3.3}$$



Slika 3.1: Sigmoid



Slika 3.2: Linearna rektifikacijska funkcija

3.1.2. Sloj neurona

Sloj neurona sastoji se od skupa neurona. Sloj neurona sastoji se od vektora a koji označava aktivaciju sloja neurona. U slučaju prvog sloja vrijedi $a = x$. Račun izlaza sloja neurona računa se slično kao i kod pojedinog neurona. Za svaki sloj neurona l izlaz z jednak je umnošku prethodne aktivacije a_{l-1} sa trenutnim težinama sloja neurona w_l zbrojen sa sklonošću b_l

$$\boldsymbol{z_l = w_l \cdot a_{l-1} + b_l} \tag{3.4}$$

$$\hat{y} = g(z) \tag{3.5}$$

3.1.3. Neuronska mreža

Neuronska mreža sastoji se od više slojeva neurona koji su međusobno povezani. Slojevi su povezani tako da je svaki neuronski sloj razine l povezan sa svakim neuronskim slojem razine $l + 1$. Povezani su na način da se izlaz neuronskog sloja l dovodi na ulaz neuronskog sloja $l + 1$. Ovaj algoritam prijenosa podataka s jednog sloja na drugi zove se *feedforward* algoritam. TODO: slika slojeva. Slojevi se dijele u tri različite grupe: ulazni, skriveni i izlazni sloj. Ulazni sloj je prvi sloj na koji se dovodi ulazni podataka, slika. Izlazni sloj je sloj koji na izlazu daje klasifikaciju ulazne slike, u ovom slučaju vektor predviđenih emocija. Skriveni slojevi su među-slojevi koji obrađuju podatke prethodnog izlaznog sloja neurona. Što je neuronski sloj dublje razine to je apstrakcija podataka koju on računa veća. Izlaz skrivenog sloja poslat će se na ulaz sljedećeg

skrivenog sloja a tim prijelazom apstrakcija podataka će biti uvećana. Uloga svakog neuronskog sloja je da obrađuje podatke na razini apstrakcije definiranje njegovom dubinom. Većim brojem slojeva neurona stvara se dublja neuronska mreža koja može računati na većoj razini apstrakcije. Interpretacija razine apstrakcije u slučaju slika svodi se na promatranje različitih djelova slike različite veličine. Iako se promatraju različiti dijelovi slike, međuzavisnost neuronskih slojeva omogućava neuronskoj mreži da zna koje je svojstvo slike proizašlo iz prethodnog svojstva slike. Primjer bi bio pronalaženje ruba usnice i ruba oka na slici. Iako ta dva rubovi izgledom mogu biti slični bitna je spoznaja da jedan rub proizlazi iz slike usnice a drugi iz slike oka. TODO: slika usnica oko, sličan rub.

3.1.4. Funkcija gubitka

Funkcija gubitka (engl. *Loss function*) je funkcija koja govori koliko je neuronska mreža daleko ili blizu rezultata koji se želi postići. Rezultat funkcije gubitka bit će pogreška koju je mreža napravila prilikom predviđanja vektora emocije određene slike. Taj rezultat bit će korišten za ispravljanje parametara neuronske mreže. Funkcija gubitka treba ukazati na kojim je predviđanjima neuronska mreža pogriješila i za koliko. Križni entropijski gubitak (engl. *Cross Entropy Loss*) je funkcija gubitka dana sa 3.6 koja će biti korištena prilikom određivanja gubitka. Funkcija prima dva parametra, ispravna vrijednost označenu kao vektor x je i predviđenu vrijednost označenu kao vektor y. U slučaju predviđanja emocija x je stvaran vektor emocije za pojedinu sliku dok je y previđen vektor emocije stvoren kao izlaz neuronske mreže. Vektor w predstavlja težišta za pojedini element vektora x i y. Elementi vektora w su $w_1, w_2, ..., w_n$ gdje svaki w_i predstavlja koeficijent koji će biti pomnožen sa rezultatom gubitka. Razlog uvođenja težišta je neravnomjerna zastupljenost klasa u podatkovnom skupu za treniranje. Klase s manjim brojem uzoraka imat će veće težište prilikom računanja gubitka i samim time će biti "bitnije" prilikom treniranja. Kazna pogreške predviđanja za tu klasu tj. emociju bit će veća nego u slučaju ne korištenja težišta.

$$CrossEntropyWeighted\left(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}\right) = -\sum_{i} w_i \left(x_i log\left(y_i\right)\right) \tag{3.6}$$

3.1.5. Treniranje neuronske mreže

Neuronska mreža predstavlja nelinearnu funkciju koja se sastoji od mnogo parametara. To su spomenuti parametri pojedinog neurona spomenuti na formuli 3.1, težišta

neurona $\boldsymbol{w}2$ i sklonost neurona b. Mijenjanje tih parametra parametra na način da se rezultat funkcije gubitka smanjuje naziva se treniranje. Cilj treniranja je minimizirati gubitak na način da se pronađe lokalni minimum u n dimenzionalnom prostoru, gdje je broj n broj težina u neuronskoj mreži. Metoda pronalaska minimuma zove se spuštanje gradijentom (engl. *gradient descent*). Računanje spusta svodi se na izračun derivacije parametra neuronske mreže prema 3.7 gdje je greška neuronskog sloja E, a aktivacijski izlaz i w težina sloja

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial a} \cdot \frac{\partial a}{\partial w} \tag{3.7}$$

Ovom jednadžbom dobiven je smjer negativnog gradijenta koji vodi ka minimumu. Gradient se računa prilikom svake iteracije treniranja, tj. nakon svakog dovođenja slike na ulaz mreže.

..... **stohasitcki ne ide uvijek prema minimumum** **nakon svakog ulaza se racuna** **lokalni minimum =/= pravi**

općenito2 **problem treniranja dubokih mreža** **gradient degradation problem**

3.1.6. Rezidualna neuronska mreža

3.1.7. ResNet 50

probleme koje riješava

kako funckionira

3.1.8. Prijenosno učenje

transfered learning

zasto ga koristimo umjesto cijelokupnog treniranja

nedostatak labeliranih slika

3.2. Implementacija treniranja u PyTorch-u

3.2.1. Priprema podataka

Augmentacija slika

Transformacija podataka

3.2.2. ResNet 50

3.2.3. Adam optimizator

3.2.4. Funckija gubitka

križni entropijski gubitak

3.2.5. Treniranje

4. Testiranje/evaluacija

5. Zaključak

Zaključak.

LITERATURA

[1] Numpy array. URL https://numpy.org/doc/stable/reference/generated/numpy.array.html.

[2] Patrick Lucey, Jeffrey Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, i Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. stranice 94 – 101, 07 2010. doi: 10.1109/CVPRW.2010.5543262.

[3] rakshithvasudev. One hot encoding. URL https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-to-use-it-e3c618

[4] Hardik Vasa. google-images-download. URL https://github.com/hardikvasa/google-images-download.

Prepoznavanje emocija iz izraza lica pomoću strojnog učenja

Sažetak

Sažetak na hrvatskom jeziku.

Ključne riječi: Ključne riječi, odvojene zarezima.

Title

Abstract

Abstract.

Keywords: Keywords.