

Prepoznavanje slika korišćenjem neuronskih mreža i *backpropagation* algoritma

Neuronska mreža za prepoznavanje slika realizovana je u programskom paketu MATLAB korišćenjem backpropagation algoritma. Mreža je uspešno testirana na digitalnim slikama ljudskih lica.

Uvod

Veštačke neuronske mreže funkcionišu po uzoru na ljudski mozak i očekuje se da će predstavljati jedan od ključnih koraka u daljem razvoju veštačke inteligencije. One se uspešno mogu primenjivati na mnogim zadacima koji su za ljude jednostavni i prirodni, ali se na računarima veoma teško implementiraju klasičnim metodama. Najvažnija prednost neuronskih mreža jeste to što se one obučavaju na ograničenom skupu primera. Rešavanje problema neuronskim mrežama ne zahteva poznavanje složenih matematičkih i logičkih funkcija koje povezuju ulazne i izlazne podatke, kao što je to potrebno kod algoritamskog programiranja, gde moramo precizno definisati funkcije prenosa nekog sistema.

Da bi se mreža obučila da ispravno aproksimira određenu funkciju, dovoljno je na odgovarajući način mreži predstaviti problem i obučiti je na reprezentativnom skupu podataka (u daljem tekstu – trening skup). Mreža će, ukoliko je to moguće, pronaći odgovarajuće rešenje, tj. funkciju koja povezuje ulazne i izlazne podatke prilikom procesa obučavanja. Važno je naglasiti da neuronske mreže i klasično programiranje nisu tehnike koje se međusobno isključuju.

Arhitektura neuronskih mreža

Ponašanje neuronske mreže zavisi od težinskih koeficijenata između neurona (perceptrona), modela samih neurona, odnosno njihove aktivacione funkcije i od arhitekture mreže.

Aktivaciona funkcija f može imati različite oblike, a najčešće se koriste linearna, odskočna, tangens hiperbolička i sigmoidna funkcija. U ovom radu, korišćena je sigmoidna (logistička) funkcija:

$$f(x) = \frac{1}{1 - e^{-kx}}$$

Razlog što je odabrana sigmoidna funkcija leži u tome što se njen izvod, koji nam je potreban prilikom obučavanja mreže, najjednostavnije računa i pritom daje odlične rezultate.

Arhitekturu neuronskih mreža čini specifično povezivanje neurona (perceptrona). Razlikujemo dve klase neuronskih mreža: mreže sa povratnim vezama (*rekurentne*) i mreže bez povratnih veza (*nerekurentne*, engl. *feed forward*).

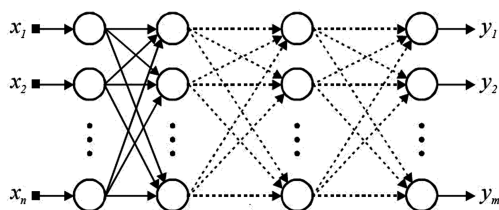
Najznačajniji predstavnik feed forward mreža je *slojevitá neuronska mreža*, koja je i korišćena u ovom radu. Kod slojevitih mreža (slika 1) neuroni su grupisani u slojeve. Karakteristično za ove mreže je da se podaci kreću samo u jednom smeru. Treba naglasiti da je izlaz svakog neurona iz jednog sloja povezan sa svim neuronima u sledećem sloju.

Osnovna podela slojevitih mreža, vrši se prema broju slojeva u mreži, pa razlikujemo jednoslojne i višeslojne. Slojevi višeslojnih mreža dele se, opet, na: ulazni, skriveni sloj i izlazni sloj. Ulazni sloj je prvi, on prima ulazne podatke. Izlazni sloj je poslednji sloj, prikazuje izlaze mreže. Obično ima onoliko neurona koliko i grupa istih slika u trening setu. Skriveni sloj, služi za internu obradu informacija i obuhvata sve slojeve između ulaznog i izlaznog.

Važno je reći da broj slojeva, u opštem slučaju, ne određuje performanse mreže. Broj neurona u ulaznom sloju ekvivalentan je broju ulaznih podataka,

Marko Bežulj (1988), Pančevó, G. Dimitrova
20A, učenik 3. razreda ETŠ “Nikola Tesla” u
Pančevu

MENTOR:



Slika 1. Shematski prikaz slojevite neuronske mreže.
 x – ulaz, n – broj neurona u ulaznom sloju, y – izlaz,
 m – broj neurona u izlaznom sloju.

Figure 1. Scheme of the layered neural network.
 x – input, n – number of perceptrons in the input
layer, y – output, m – number of perceptrons in the
output layer

odnosno broju piksela slike koja se obrađuje. Broj neurona u izlaznom sloju jednak je broju izlaza mreže odnosno broju različitih slika iz trening seta na kojima se mreža uči.

Najpraktičniji način obučavanja mreže je promena težinskih koeficijenata, baš zato što je nakon projektovanja i hardverske realizacije mreže teško promeniti organizaciju ili model neurona. Obučavanje neuronskih mreža može biti nadgledano (*supervised*) i nenadgledano (*unsupervised*)

Kod nadgledanog obučavanja mreži se prosleđuju ulaz (pobuda) i očekivani izlaz, a zatim upoređuju stvarni i očekivani odziv i razlika se prosleđuje proceduri za obučavanje, koja na osnovu razlike menja težinske koeficijente mreže. Najbolji primer ovakve obuke je *backpropagation* algoritam.

Princip forward propagation i treniranje višeslojne neuronske mreže *backpropagation* algoritmom

Forward propagation algoritam podrazumeva prosleđivanje podataka kroz mrežu i izračunavanje izlaza mreže. Zasniva se na klasičnom radu neurona koji su slojevito povezani. Neuroni svoje ulaze sumiraju i propuštaju kroz aktivacionu funkciju čime se definiše izlaz. Svaki neuron svoj izlaz prosleđuje na sve neurone u sledećem sloju. *Backpropagation* algoritam, kao i svi metodi nadgledanog obučavanja, svodi se na minimiziranje grešaka između dobijenog i željenog odziva mreže.

Korekcija greške počinje u izlaznom sloju pošto je mreži predstavljen ulazni podatak i proces forward propagation završen. Izlaz svakog neurona izlaznog sloja upoređuje se sa željenom vrednošću izlaza, a zatim se na osnovu toga za svaki neuron

izračunava vrednost greške. Nakon toga računaju se greške prethodnih slojeva sve do ulaznog. Kada se računanje grešaka završi, koriguju se težinski koeficijenti veza.

Proces učenja neuronske mreže prestaje onog trenutka kada ukupna greška mreže, za sve podatke iz trening seta, dostigne zadatu minimalnu vrednost.

Softverska implementacija

Algoritam se sastoji iz nekoliko osnovnih delova. Datoteka *loadpict.m* učitava sve podatke (konkretno slike) i filtrira ih tako da sve budu iste rezolucije i u istoj paleti boja, crno-beloj. Datoteka *koef.m* (prilog 1) generiše mrežu sa proizvoljnim sinaptičkim težinama veza. Pokreće se samo prilikom prve inicijalizacije mreže. Težine veza su u opsegu od -0.5 do $+0.5$ i nalaze se u matricama $w1, w2$ i $w3$. *Backprop.m* (prilog 4) je deo programa koji obuhvata proces treniranja mreže.

Program koji je napisan u MATLABu potpuno je fleksibilan i može se jednostavno izmeniti i prilagoditi nekom drugom slučaju. U fajlu *parametri.m* (prilog 2) čuvaju se osnovni podaci o konstrukciji mreže.

Iz priloženog fajla možemo zaključiti da je u ulaznom sloju broj neurona jednak broju piksela slike, odnosno da se svaki piksel slike prosleđuje na zaseban neuron. U srednjem sloju ima $2/3$ neurona u odnosu na ulazni sloj. U literaturi ne postoji podatak koliko je potrebno neurona u skrivenom sloju niti koliko on utiče na rad. Tako da je ova vrednost određena eksperimentalno kao optimalan odnos brzine učenja i rezultata

Datoteka *calc.m* (prilog 3) predstavlja funkciju koja izvršava *forward propagation*. Ona prosleđuje trenutni ulaz x , koeficijente mreže matrice $w1, w2, w3$ i broj neurona u svakom sloju. Kada se izračuna izlaz svakog neurona u ulaznom sloju, matrica *izl1*, svaki od njih svoj izlaz prosleđuje u aktivacionu funkciju, datoteka *tresh.m*. Nakon toga, na sličan način računa se i izlaz drugog sloja, matrica *izl2* i konačni izlaz mreže, matrica *izl3* odnosno *izl*.

Nakon inicijalizacije programa (učitavanja parametara, trening seta i generisanja koeficijenata mreže) definiše se koeficijent učenja mreže i zatim sledi proces obučavanja mreže, datoteka *backprop.m*. Obučavanje počinje generisanjem idealnog izlaza mreže za svaku sliku iz trening seta, koji se kasnije koristi za upoređivanja i računanje ukupne greške

sistema. Idealni izlaz definiše se tako da redni broj neurona označava grupu istih slika i samo on daje na izlazu 1 kada je na ulazu slika iz njegove grupe.

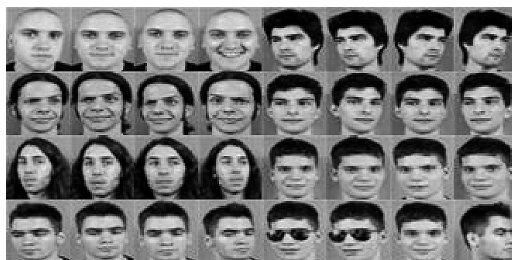
Izlaz mreže organizovan je na sledeći način: svaka različita slika zauzima jedan neuron na izlazu i kada mreža prepozna određenu sliku, izlaz teži ka tome da samo izlazni neuron koji odgovara grupi u kojoj je ta slika (što je ujedno i redni broj tog neurona) ima na izlazu ima 1, a ostali neuroni 0. Nakon definisanja idealnog izlaza, sve slike iz trening seta prolaze kroz mrežu i računa se početna ukupna greška sistema. Kada je definisana početna greška sistema počinje proces treniranja. Taj proces traje sve dok ukupna greška sistema ne dostigne unapred određenu minimalnu vrednost. Minimalna vrednost je definisana promenljivom *err_limit*, kao broj slika u trening setu pomnožen sa 0.0005. Sastavni deo bloka za obučavanje je i blok za testiranje mreže. Međutim, ako bi se mreža testirala pri svakoj iteraciji obučavanja, vreme obučavanja bi bilo duže, a efikasnost mreže bi ostala ista. Broj iteracija na koliko se vrši testiranje, nalazi se u promenljivoj *err_rec*.

Da u slučaju greške proces treniranja mreže ne bi trajao beskonačno dugo, uveden je brojač iteracija koji zaustavlja algoritam ako broj iteracija pređe preko unapred zadate vrednosti promenljivom *it_limit*. Koeficijent brzine učenja mreže definisan je tako da se menja prema ukupnoj grešci sistema. Početna vrednost iznosi 0.75. Kada ukupna greška dostigne 200 puta veću vrednost *err_limita*, greška se smanjuje na 0.5, a kada dostigne 20 puta veću vrednost *err_limita* smanjuje se na 0.3. Jedna iteracija backpropagation algoritma sadrži uzimanje slučajne slike iz trening seta, propagaciju slike unapred kroz mrežu, poređenje dobijenog sa željenim izlazom i ispravljanje koeficijenata.

Verifikacija i rezultati

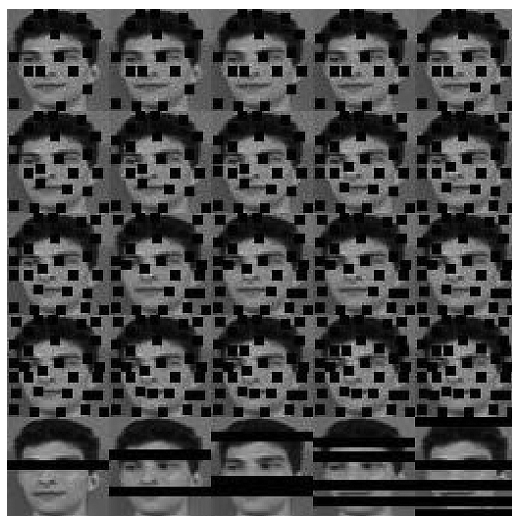
Na slici 2, prvih dvadeset osam slika sedam lica (grupa) predstavlja trening set neuronske mreže, dok su poslednje četiri slike samo neki od primera kojima je mreža dodatno testirana.

Za prvo testiranje iz trening seta (slika 2) uzeta je slika iz četvrte grupe (poslednja u drugom redu, gledano s leva) i postepeno je zatamnjavana kroz 25 različitih faza što je prikazano na slici 3. Na prvoj sličici u slici 3, zatamnjeno je 15% slike, a na svakoj sledećoj zatamnjen je 1% više (jedan cmi kvadratić



Slika 2. Trening set i neke od slika za testiranje mreže

Figure 2. Training set consists of seven groups and four test images



Slika 3. Prvi test primer

Figure 3. First test image

više) i to pravilo važi sve do dvadesete slike. Dvadeset prva slika zacrnjena je za 10% i svaka sledeća je zacrnjena za 10% više.

Iz izlaznih podataka zaključeno je da mreža daje 99.9% tačnosti za šum do 30%. Ako pogledamo petu sliku u trećem redu, možemo reći da ni čovek ne bi mogao pouzdano da zna ko je na slici. Zadnjih 5 slika daje bolje rezultate čak i pri šumu od 40% zato što je šum ravnomerno raspoređen u određenoj oblasti, tj. postoje veći delovi slike koji su bez šuma.

Za testiranje je korišćen potprogram, datoteka *testallout.m*, koja za priloženu sliku ispisuje sve izla-

ze mreže. Nakon završenog obučavanja mreže, izlaz na osnovu slika iz trening seta odstupao je od idealnog, ali u okvirima očekivanog. Naime, ukupna greška ove neuronske mreže iznosi 0.012393, što je u proseku $0.443 \cdot 10^{-3}$ što znači da daje zadovoljavajuće rezultate. U navedenom slučaju bilo je potrebno 320 iteracija treniranja. Po izvršenim testiranjima potrebno je od 250 do 500 iteracija za ukupnu grešku koja je definisana kao broj slika u trening setu pomnožen sa $0.5 \cdot 10^{-3}$. Ako bi se ukupna greška povećala, broj potrebnih iteracija bi se znatno smanjio, na račun mogućnosti prepoznavanja mreže. Nakon optimizacije *backpropagation* algoritma, vreme potrebno za obuku mreže na datom primeru se svodi na opseg od 2 do 5 minuta, dok je za *forward propagation* potrebno manje od jedne sekunde.

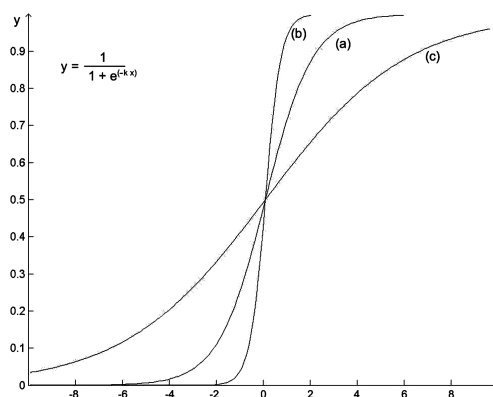
Iz ovih rezultata zaključuje se da je obučena mreža u stanju da obrađuje podatke u realnom vremenu. Bitno je naglasiti da je ključan deo u ubrzanju algoritma formiranje srednje vrednosti slike trening seta i zatim umanjivanje svake slike za tu vrednost. Tada na svakoj slici ostaju samo karakteristične linije i očigledan je zaključak da je lakše istrenirati mrežu. Bez navedenog postupka mreži je potrebno više od 20 000 iteracija za obučavanje do zadovoljavajućeg nivoa prepoznavanja.

Poslednje četiri slike sa slike 3 predstavljaju karakteristične test primere mreže, jer se znatno razlikuju od slika iz trening seta. Izlaz mreže za prvu od četiri slike je pokazao da mreža sa sigurnošću od 99.6% prepoznaje da je to slika iz grupe 6.

Drugu sliku, i pored toga što je lice na slici приметно više okrenuto u levu stranu, mreža sa sigurnošću od 93% svrstava sliku u grupu 6.

Treća slika još jedan je primer kog nema u trening setu – lice je okrenuto direktno ka posmatraču, a mreža sa sigurnošću od 99.8 svrstava sliku u grupu šest, a sa sigurnošću od 27% sliku u grupu 7. Razlog što je najveći procenat prepoznavanja dodeljen grupi šest, verovatno je posledica nedostatka frontalnog položaja lica na slikama u trening setu, a koji se pri maloj rezoluciji zaista poklapa sa frontalnim pogledom lica iz grupe 7.

Četvrta slika iz test skupa nešto je komplikovaniji primer. Mreža sada sa samo 48% tvrdi da to jeste sedma slika, međutim u odnosu na ostale izlaze ovaj se znatno izdvaja. Kada bi u trening setu bilo neko drugo lice u sličnom položaju, mreža bi



Slika 4. Izgled aktivacione funkcije pri različitim parametru k : (a) $k = 1$, (b) $k = 3$, (c) $k = 0.333$

Figure 4. Function of activation for different k parameter: (a) $k = 1$, (b) $k = 3$, (c) $k = 0.333$

verovatno i na izlaznom neuronu te slike imala veći procenat prepoznavanja.

Jedna od metoda izoštravanja rezultata mreže jeste povećavanje parametra k u *threshold* funkciji.

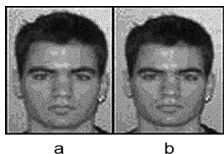
Na slici 4 kriva *a* predstavlja izgled aktivacione funkcije kada je $k = 1$, i ovaj oblik je korišćen za treniranje mreže i testiranje. Smanjivanjem parametra k , izgled funkcije sve više teži ka linearnom (kriva *c*). Taj oblik se nije pokazao koristan u ovom radu. Međutim povećanje parametra k za posledicu ima da sigmoidna funkcija sve više liči na odskočnu. To ima za posledicu da ukoliko su rezultati veći 55%, mreža daje sigurnije izlaze, dok manje rezultate smanjuje. Dakle, dobro projektovan trening set (koji obuhvata sve varijante objekta koji se očekuje) znači povećanje parametra k nakon procesa obučavanja i poboljšava rezultate mreže.

Obučavanje sa parametrom k kada je ono veće od 1 nije mnogo ispitivano. Ono daje lošije rezultate za slike čija forma mnogo odstupa od slika kojih nema u trening setu. Sa druge strane u mnogo više slučajeva za rezultat daje 100% tačnosti, ali događa se da daje 100% za više od jedne slike. Bitno je napomenuti da je trajanje obučavanja oko 30% duže.

Jedan od problema koji se javlja je to što se mreži moraju prosledivati dobro izdvojene slike, a

što podrazumeva fiksna rastojanja između objekta koji je bitan na slici i ivica slike.

Na slici 5 vidimo dve naizgled iste slike iz grupe 7. Međutim, mreža daje veoma različite rezultate za svaku, zato što je prva slika uvećana, tj. razdaljina od lika do ivice slike 5a manja je nego u trening setu kod slika u trening setu, pa pri promeni veličine slike lik na jednoj slici izgleda veći. Slika 5b primer



Slika 5. Problem prepoznavanja zbog promene plana

Figure 5. Problem of recognition due to the difference in shot distance

je bolje izdvojene slike pa je veličina glave približnija veličini iz trening seta.

Pri vrednosti $k = 1$, mreža prepoznaje sliku 5a kao sliku iz grupe šest sa sigurnošću od 37% a svrstava je u grupu 7 sa sigurnošću od samo 0.18%, dok je za vrednost $k = 3$ slika 5a svrstana u grupu 6 sa manjom sigurnošću od 26% ali bez rezultata koji bi implicirali da slika pripada grupi 7 (0%).

Izlaz koji mreža daje za sliku 5b nešto je bolji. Za vrednost $k = 1$, mreža sliku 5b svrstava u grupu 6 sa sigurnošću od 52% a u grupu 7 sa čak 98.7%. Kada je $k = 3$, mreža sliku 5b svrstava u grupu 6 sa sigurnošću od 76%, ali zato sa sigurnošću od 100% tvrdi da slika pripada grupi 7.

Iz ovih rezultata vidi se da prvu sliku mreža ne prepoznaje ispravno i vidi se još jedan primer kako menjanje parametra k utiče na rezultate. Slika 5b prepoznata je bolje, mada je došlo do mešanja sa grupom 6, kao i u primeru frontalnog pogleda slike iz grupe 6 koja se mešala sa grupom 7. Vidi se i kako je promena parametra k povećala tačnost, ali i grešku svrstavši sa većom sigurnošću sliku u pogrešnu grupu.

Razlozi za ove rezultate mogu se objasniti na više načina. Slike 5a i 5b su slikane na pozadini koja je drugačije osvetljena nego slike iz trening seta što takođe predstavlja problem pri prepoznavanju slike mada ne dolazi mnogo do izražaja jer se radi o crno-belim slikama. Ipak važno je naglasiti, da pozadina takođe utiče na tačnost rezultata mreže.

Zaključak

Neuronske mreže nisu dobro istražena oblast. Zbog toga se u literaturi teško pronalaze konkretni podaci o osobinama mreže i zavisnostima izlaza od menjanja parametara. Tako da se svaki početak istraživanja u ovoj oblasti zasniva na parametrima koji se pretpostave ili odrede uz pomoć malog broja merenja.

Jedna od mana ovog rada je to što radi sa crno belim slikama. Naravno, postoji način da se ova mreža unapredi tako da radi sa slikama u boji. Najjednostavnije i najbrže rešenje jeste da se naprave 3 mreže – svaka za po jednu komponentu RedGreen-Blue sistema, tako da svaka mreža sa 33.3% učestvuje u krajnjem odgovoru.

Nije određen kapacitet slika koje neuronska mreža može da nauči, a ova je testirana na slučajevima koji su neobični i u većoj meri različiti od slika iz trening seta. Ukoliko dođe do zasićenja mreže, to znači da za veliki broj različitih slika u trening setu mreža ne može da postigne konkretni kvalitet. Može se povećati broj neurona u skrivenom sloju ili čak dodati još jedan skriveni sloj, ali bez testiranja ne zna se koliko to može da pomogne. Najbolje rešenje tog problema bilo bi da se napravi više manjih mreža koje bi obrađivale određeni broj grupa slika iz trening seta. Pri korišćenju ove metode bitno je i propuštati sve slike kroz svaku mrežu i pritom učiti mreže da na izlazu daju sve jedinice za slike koje im ne pripadaju, kako bi se bolje izoštrila tačnost mreže

Bitno unapređenje za ovaj program bilo bi napisati pred program koji pri učitavanju slike vrši kvalitetno izdvajanje objekta koji želimo da mreža prepozna. Još jedno od unapređenja ovog rada bilo bi pisanje programa u nekom od programskih paketa C ili Delphi i povezivanje sa kamerom i testiranje rada u realnom vremenu ili pravljenje nezavisnog sistema povezivanjem procesora (npr: arm, atamega isl.), DSP ili FPGA čipa sa kamerom.

Prepoznavanje lica jedan je od najkomplikovanih primera prepoznavanja slika, ali ako bi se program koristio za jednostavnije oblike, na primer saobraćajne znakove, mreža bi davala još sigurnije rezultate

Literatura

Jocković M., Ognjanović Z., Stankovski S. 1997. *Veštačka inteligencija*. Beograd: Krug

Tadeusiewicz R. 1992. *Sieci neuronowe*. Kraków

<ftp://ftp.sas.com/pub/neural/FAQ.html>

<http://solair.eunet.yu/~ilicv/ANN.html>

Marko Bežulj

Neural Networks and Backpropagation Algorithm in Image Recognition

A neural network for image recognition was designed in MATLAB, using the backpropagation algorithm. In order to teach the neural network how to approximate the given function, it is enough to train it on the representative set of data (training set). The network was successfully trained, and then tested on the set of digital images of human faces.

Prilozi

