Prepoznavanje bolesti na osnovu fotografija ljudskih pluća primenom CNN

Konvolucione neuronske mreže mogu biti idealne za rešavanje ovog problema ukoliko se uspe razviti dovoljno dobar model, odn. mreža koja će moći da postiže dobre rezultate nad potpuno nepoznatim fotografijama ljudskih pluća (koje pripadaju nekoj od klasa za koje je mreža trenirana). U sklopu ovog projekta bilo je potrebno klasifikovati fotografije u tri grupe: *Normal, Virus i Bacteria*.

Upotrebljene biblioteke

- TensorFlow – rad sa neuronskim mrežama
- Keras – rad sa neuronskim mrežama na visokom nivou apstrakcije (u pozadini koristi TensorFlow)
- Matplotlib – vizualizacija podataka

Priprema podataka

Podaci iz *chest_xray_metadata.csv* učitani su tako da postoje tri liste, odn. po jedna za svaku klasu fotografija: *normal_set*, *bacteria_set*, *virus_set*. Učitani podaci su razdvojeni na *train_set* i *test_set*. Primenjen je *supervised learning* – svaka klasa ima svoju labelu (0, 1 ili 2), kako bi mreža mogla da uporedi svoju procenu sa tačnom klasom.

Posmatrane metrike

Tokom izrade projekta praćeno je kako se ponašaju vrednosti *accuracy* i *loss*, i u skladu sa tim promenama vršene su izmene u modelu kako bi vrednost *accuracy* bila što veća, a vrednost *loss* što manja (barem smo se nadali da će tako biti). Za svaki pokušaj beležene su numeričke vrednosti i grafički prikazi posmatranih parametara za *train* i *validation* set.

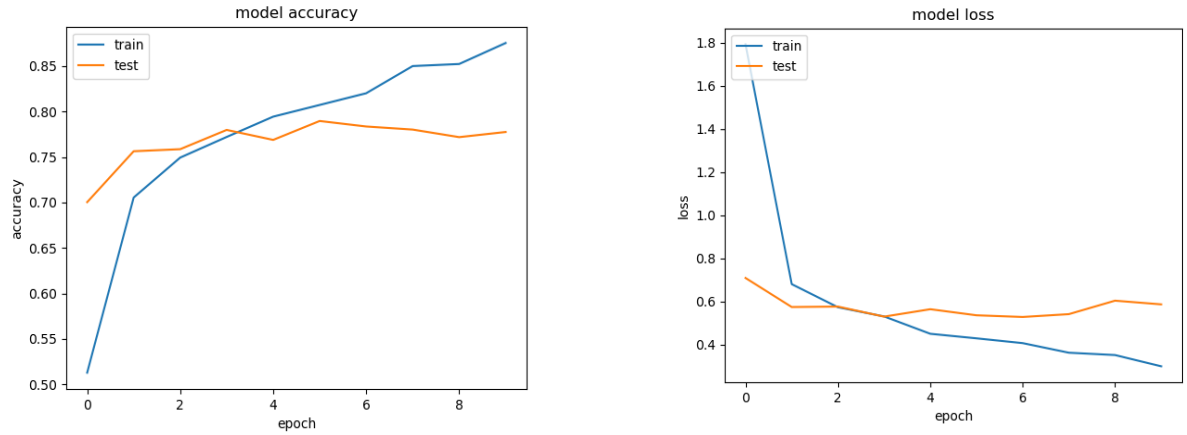
Prvi pristup – Posmatranje broja slojeva

Konvolucioni slojevi sadrže filtere koji detektuju paterne u fotografijama. Sa svakim konvolucionim slojem specificirano je koliko filtera taj sloj treba da ima. Na početku mreže ti filteri su dosta jednostavni – najčešće su to geometrijski filteri koji detektuju ivice na slici, dok u dubljim slojevima CNN ti filteri postaju sofisticiraniji odn. mogu da prepoznaju specifične objekte.

Prvi pokušaj se svodi na određivanje u kolikoj meri broj konvolucionih slojeva utiče na kvalitet rešenja:

1. 5 konvolucionih slojeva (svaki je praćen MaxPooling slojem)
2. 16 konvolucionih slojeva
3. LightCNN

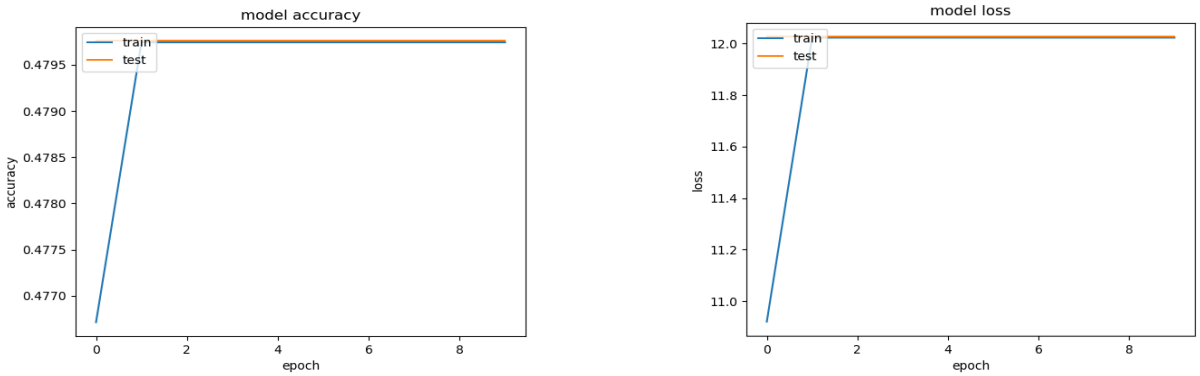
Pretpostavka: Ako postoji više konvolucionih slojeva, postojaće i više filtera koji će iz sloja u sloj sve bolje detektovati specifičnosti koje se tiču tri posmatrane klase, pa će samim tim rezultati biti bolji.



Plot 1 – Model sa 5 konvolucionih slojeva

Može se uočiti da se tokom 10 epoha *accuracy* povećavao. Već tokom prve epohe model dostiže preciznost od oko 50% i nastavlja da raste do oko 87%. Validacioni set dostiže preciznost od oko 77% i nema značajno velikih oscilacija tokom trajanja epoha.

Loss vrednost značajno opada tokom prve dve epohe sa 1.8 na oko 0.7. Nakon toga srazmerno nastavlja da opada do vrednosti oko 0.2. *Loss* validacionog seta varira oko vrednosti 0.6 tokom svih 10 epoha.



Plot 2 – Light CNN 29

Na drugom predmetu u ovom semestru opisivali smo koje CNN arhitekture su se dobro pokazale u identifikaciji ljudskog lica. Ispostavilo se da najveću preciznost dostiže Light CNN 29 (state-of-the-art performanse). Zbog toga smo odlučili da isprobamo ovu mrežu nad našim setom podataka. Međutim, ispostavilo se da ovakva arhitektura nije idealna za rešavanje našeg problema.

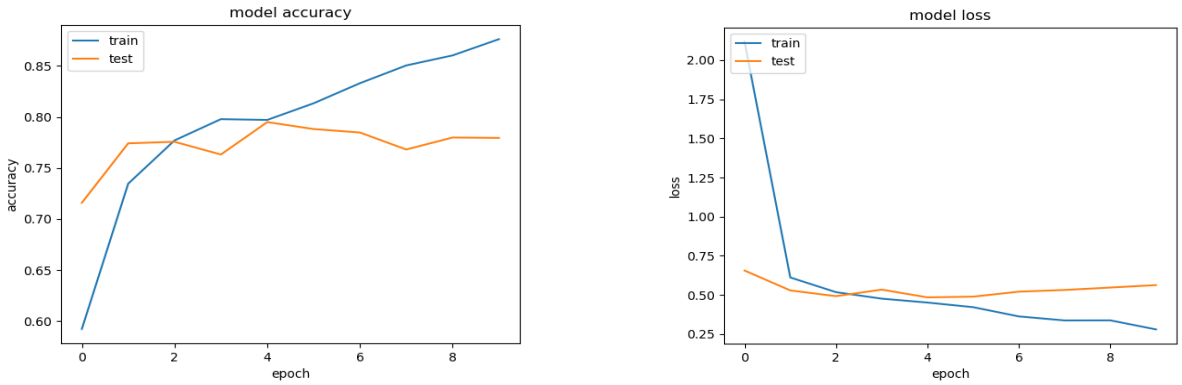
Accuracy se ne poboljšava više nakon prve epohe, kao ni *loss* – Rezultati su značajno lošiji u odnosu na prvi pokušaj (CNN sa 5 konvolucionih slojeva).

Zaključak: Ispostavilo se da veći broj slojeva ipak ne donosi obavezno bolje rezultate. Pretpostavka je bila loša, jer više slojeva sa sobom nosi i više parametara (za test 3 čak oko 14 miliona parametara). Problem kod većeg broja slojeva je to što se konvolucija primenjuje više puta, a dimenzije fotografije opadaju množenjem matrica. Proces je započet sa ulaznim fotografijama dimenzija 150 x 150, što znači da je vrlo moguće da je nakon mnogo primenjenih konvolucija fotografija izgubila smisao, pa su postignuti rezultati bili lošiji.

Drugi pristup – Zero padding

Zadržano je rešenje sa manje konvolucionih slojeva. Povećane su dimenzije ulaznih fotografija u mrežu sa 150 x 150 na 250 x 250. Uveden je *zero padding*.

Pretpostavka: Semantika fotografije će biti bolje očuvana ukoliko je ulazna fotografija veća i ukoliko se kroz slojeve održava ta ista dimenzionalnost. Očekuje se da rezultati budu makar malo bolji, ali da faza treniranja zbog povećane dimenzionalnosti sada bude duža.



Plot 3 – Rezultati nakon primene padding-a i povećanja dimenzija slika

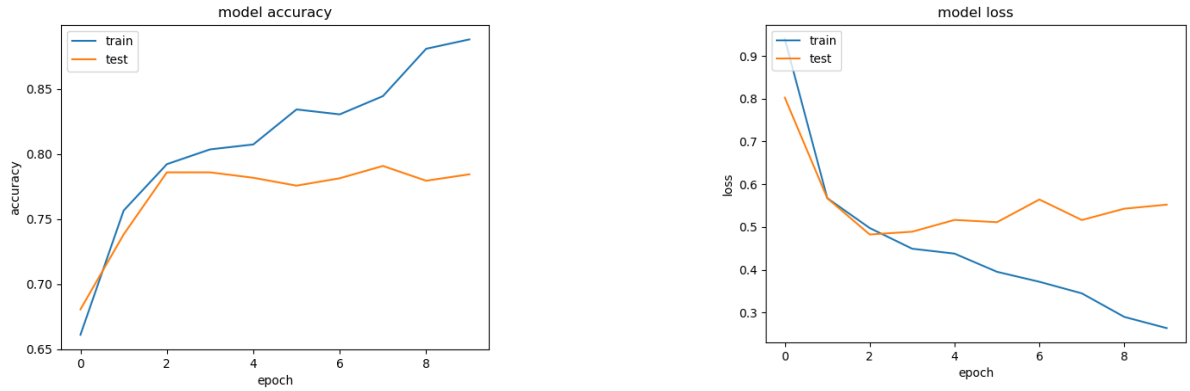
Dobijeni rezultati pokazuju da ne postoje značajne razlike između početnog modela koji nije imao *padding* i trenutnog modela. Grafici su na oko gotovo identični, ali numeričke vrednosti pokazuju da se *accuracy* testnog seta poboljšao za oko 1.5%, a loss je bio manji za oko 1.

Zaključak: Ne postoje statistički značajne razlike između početnog i trenutnog modela. Može se reći da se dobiju mizerno bolji rezultati, ali to je gotovo zanemarljivo. Odavde se može zaključiti da tokom konvolucija i max poolinga nije dolazilo do gubitaka značajnih informacija, odn. očuvan je smisao fotografije tokom treniranja.

Treći pristup – Promena learning *rate*-a

Zadržano je rešenje iz prethodnog pokušaja. Povećan je broj epoha za trening, a *learning rate* je setovan na 0.001. Podešavanje *learning rate*-a, laički rečeno, oslikava koliko brzo mreža uči. Predstavlja „korak“ ka minimumu. To je „pipljiv“ parametar, jer ukoliko je njegova vrednost mala, sporo se kreće ka minimumu, a ukoliko je prevelika, može se desiti da nikada ne iskonvergira ka minimumu.

Pretpostavka: Mreža bolje uči ukoliko je odabran odgovarajući *learning rate* za odabranu arhitekturu. Očekuju se bolji rezultati za *accuracy*.



Plot 4 – Learning rate 0.001

Na priloženim graficima se može uočiti da postoje značajne razlike između vrednosti accuracy koja se dostiže na *train* setu (oko 90%) i na *validation* setu (oko 78%). Došlo je do *overfittinga* mreže, što objašnjava zašto je rezultat nad *validation* setu toliko lošiji.

Zaključak: Smanjenje *learning rate*-a samo po sebi nije dovelo do boljih rezultata, ali smo u ovom pokušaju primetili da imamo problem *overfittinga*. Pretpostavljamo da će dobro odabrani *learning rate* doprineti boljim rezultatima u kombinaciji sa odgovarajućim brojem epoha i nekim metodama optimizacije.

Četvrti pristup – Pretprocesiranje

Prethodnom modelu ćemo povećati broj epoha i to ispratiti mehanizmima koji će sprečiti razvijanje sklonosti ka podacim:

1. blago menjanje ulaznih podataka

   Dodat je ImageDataGenerator koji može da rotira, flipuje, odn. pravi blage promene nad trening skupom podataka, kako bi pri svakoj novoj epohi mreža dobila „osvežen“ skup podataka i tako bila sprečena da razvije sklonost – na ovaj način je povećan diverzitet trening podataka.

2. dodavanje drop-out slojeva

   Dropout pomaže redukovanju *overfittinga* tako što omogućuje *random* ignorisanje izlaza nekog sloja. Npr. rezultujuća matrica nakon konvolucije pomoću nekog filtera može biti setovana na nule i neće biti u tom trenutku važna sledećem sloju. Na taj način izbegnuto je da izlaz jednog sloja eksplicitno zavisi od jednog neurona iz prethodnog sloja.

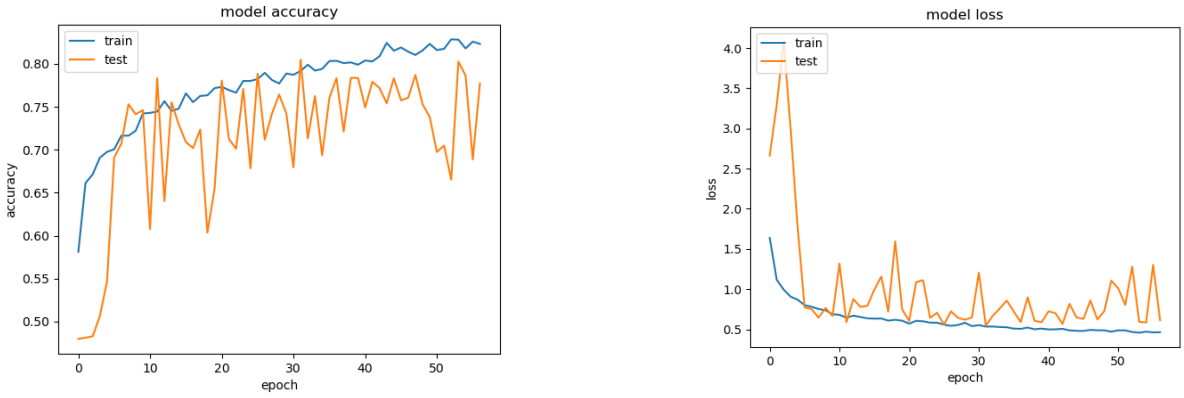
3. dodavanje L2 regularizacije

   Dodata je L2 regularizacija (najčešće korišćena u CNN) – sprečava *overfitting* tako što obezbeđuje da svi filteri imaju gotovo podjednak uticaj na predikciju.

4. dodavanje early-stoppinga i checkpointovanje

   Omogućava da se za trening izabere proizvoljno velik broj epoha, jer će treniranje biti stopirano onog momenta kada rezultati nad validacionom testom ne budu napredovali. Na taj način je sprečen *overfitting*. Checkpointovanje osigurava da će nakon završetka treniranja biti korišteni parametri koji su postigli najbolji rezultat.

Pretpostavka: Primenom ovih mehanizama trebalo bi da *training accuracy* opadne (što je u ovom slučaju cilj – jer kod overfittinga to predstavlja *fake-accuracy*), a validatcioni *accuracy* bi trebalo da poraste.



Plot 5 – Pretprocesiranje i povećanje broja epoha

Sa grafika se može videti da više ne postoji toliko veliki jaz između rezultata dobijenih nad trening setom (oko 82%) i validacionim setom (oko 78%). Pošto se inicijalizacija modela radi sa *random* vrednostima, postoji mogućnost da ti početni parametri budu loši (što objašnjava veliku vrednost za *loss* u prvih par epoha za validacioni set), ali kako odmiču epohe ti parametri se stabilizuju.

Zaključak: Posmatrane metrike su dostigle najbolje rezultate do sada. Primenjeni mehanizmi za sprečavanje *overfittinga* i povećanje broja epoha doprinele su tome da ne postoje značajne razlike između preciznosti nad trening i validacionim setom. Ova verzija modela je odabrana za finalnu verziju.

Finalna verzija modela

Model se sastoji od 6 konvolucionih slojeva. Posle svakog konvolucionog sloja postoji *BatchNormalization* sloj, a nakon njega *MaxPooling* sloj. Broj neurona po slojevima je 32 za prva dva sloja, 64 za sledeća dva i 128 za poslednja dva. (U nekoj od ranijih verzija išli smo linearno do 512, ali to je rezultovalo sa mnogo parametara i rezultati nisu bili dobri – potreban je jednostavniji model za rešavanje ovog problema).

Upotreba *BatchNormalization* sloja omogućava korišćenje većih *learning rate*-ova, poboljšava brzinu treniranja. Mreža je manje osetljiva na početne (*random*) težine.

MaxPooling slojevi su korišteni jer, pored toga što redukuju *overfitting*, rade kao *noise supressant* – odbacuju *noisy* aktivacije i ekstrahuju dominantne *features*. Smanjuju računsku kompleksnost procesiranja podataka kroz redukciju dimenzionalnosti.

Korišten je *padding* – tako da se čuva dimenzionalnost slike. Kako ivice mogu nositi značajne informacije, upotrebom paddinga omogućava se da svaki piksel sa fotografije može da se nađe u centru filtera tokom konvolucije.

Postoji *trade-off* između brzine i preciznosti – veći *stride* će doprineti manjoj računskoj zahtevnosti, ali s druge strane neće imati povoljan uticaj na *accuracy*. Stoga je odabrana mala vrednost *stride*-a kako bi se sačuvale značajne informacije i poboljšala preciznost u određivanju tačne klase.

Korištena je *ReLu* aktivaciona funkcija u svakom konvolucionom sloju. Ona je korisna jer jako negativne vrednosti preslikava u nulu (neaktivni neuroni), a pozitivne vrednosti ostavlja takve kakve jesu, jer one predstavljaju meru aktivacije neurona (za razliku od sigmoidne funkcije gde bi se pozitivne vrednosti preslikavale oko jedinice i tako bismo izgubili značajne informacije).

Postoji *Dropout* sloj koji pomaže i izbegavanju da izlaz jednog sloja eksplicitno zavisi od jednog neurona iz prethodnog sloja. Uveden je i *Flatten* sloj koji je neophodan za povezivanje konvolucionih slojeva sa potpuno povezanim izlaznim slojem.

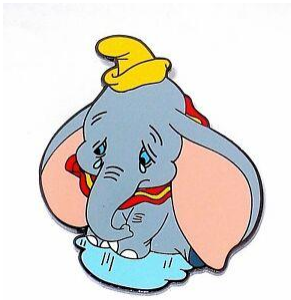
Softmax aktivaciona funkcija je korištena u poslednjem *dense* sloju – izlaz iz modela (odn. skorove) tumači kao verovatnoće; vrednosti se svode na opseg između 0 i 1.

Primenjivani su *Adam* i *RMSprop optimizeri* ali nije došlo do značajnih razlika u postignutim rezultatima, stoga u finalnoj verziji modela koristimo *Adam*, jer izvori na internetu tvrde da je to trenutno najčešće korišten *optimizer* koji postiže najbolje rezultate.

Model se završava jednim *fully* connected slojem koji ima 3 neurona (onoliko koliko ima klasa). Taj poslednji sloj (koji je zapravo *softmax* sloj) daje predikciju za svaku od tri klase.

Postignuti rezultati		
	Accuracy	Loss
Train set	oko 82%	oko 0.46
Validation set	oko 78%	oko 0.61
Test set	**37.5%**	

Nervni slon rezultati



Plot twist!

Ispostavilo se da je rezultat nad nepoznatim testnim skupom podataka veoma loš (svega 37.5%) i da ovo nikako ne može biti finalna verzija modela. Pretpostavljamo da je problem što je su trening set i validacioni set u odnosu 50:50 – trebalo bi povećati trening set, a smanjiti validacioni kako bi mreža imala veći skup podataka iz kojeg će da uči.

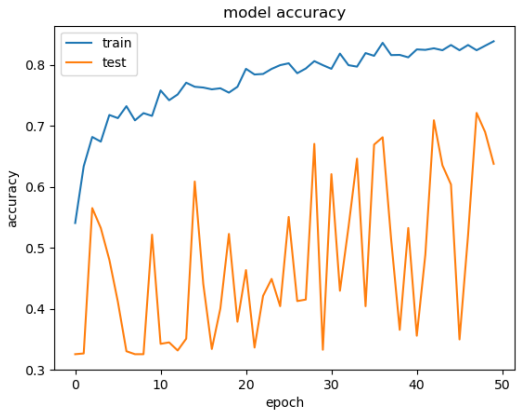
Uočeno je da mreža za svaku fotografiju iz testnog skupa pogađa samo jednu klasu (*normal* fotografije), pa automatski ne postoji ni jedan pogodak za fotografiju tipa *bacteria* ili *virus*. Pretpostavljamo da bi bilo dobro da se model još malo uprosti – smanjenje broja konvolucionih slojeva i

povećanje broja potpuno povezanih slojeva kako bi se poboljšalo korišćenje podataka koje su filteri izdvojili.

Zaista finalna verzija modela

Sprovedene su sledeće izmene nad modelom:

1. Izbačeni su po jedan konvolucioni sloj sa 128 filtera i 32 filtera – model sada ima 4 konvoluciona sloja sa 32, 64, 64 i 128 filtera.
2. Zamenjen je jedan BatchNormalization sloj sa Dropout slojem.
3. Dodata su dva potpuno povezana sloja sa 128 i 64 neurona, nakon *Flatten* layera.
4. Izbacili smo *ImageDataGenerator* – pretpostavljamo da smo previše menjali slike u fazi treniranja, pa se možda zbog toga mreža nije dobro snašla sa testnim skupom podataka.
5. Smanjili smo broj koraka po epohi, tako da se iskoristi po pola trening skupa u svakoj epohi.
6. Povećali smo batch size sa 32 na 64.
7. Promenili smo odnos trening seta podataka i validacionog seta sa 50:50 na 80:20 – podatke smo organizovali tako da bude podjednak broj fotografija svake klase i u trening i u validacionom skupu. Nasumično je izbačeno 50% fotografija iz skupa *bacteria*, kako bi nam skupovi bili jednaki (tako je sprečena sklonost ka jednoj klasi).



Plot 6 Finalni model

Grafik prikazuje postignutu preciznost nad trening skupom podataka (oko 81%) i nad validacionim skupom (oko 72%).

Postignuta preciznost nad testnim skupom podataka je 72%, što je značajno bolji rezultat u odnosu na prethodnih 37.5%.

Zaključak

Razvijeni CNN model nije idealan, ali predstavlja polaznu osnovu za dalje učenje ove oblasti. Kroz izradu projekta upoznali smo se sa osnovnim principima rada konvolucionih neuronskih mreža i donekle stekli osećaj koliko odabir parametara utiče na preciznost rada mreže. Svaka promena može dovesti do velikog poboljšanja performansi (ali i do pogoršanja), a dobar odabir parametara zahteva veliko iskustvo i poznavanje ove tematike.