

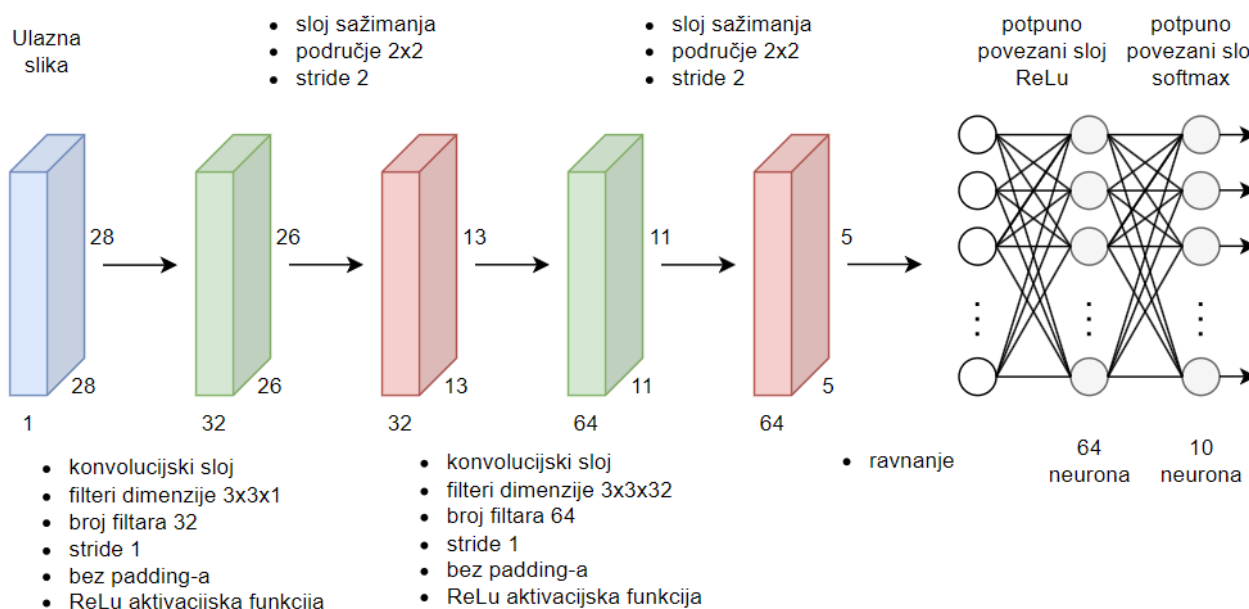
VJEŽBA 8: KONVOLUCIJSKE NEURONSKE MREŽE.

I. Cili vježbe: *Primijeniti znanje stečeno o konvolucijskim neuronskim mrežama na problemu klasifikacije rukom pisanih znamenki korištenjem Keras API-a.*

II. Opis vježbe:

II.1. Konvolucijske neuronske mreže

Konvolucijska neuronska mreža (engl. *Convolutional Neural Network* - CNN) je vrsta umjetne neuronske mreže koja je posebno efikasna u obradi podataka poput slika. Ključna karakteristika CNN-a su konvolucijski slojevi koji koriste operaciju konvolucije kako bi ekstrahirali značajke iz ulaznih podataka. CNN-ovi su vrlo učinkoviti u prepoznavanju obrazaca u slikama te se široko koriste u zadacima računalnog vida kao što su klasifikacija slika, detekcija objekata i segmentacija slika. Na slici 8.1. je dan primjer konvolucijske neuronske mreže koja se sastoji od: konvolucijskog sloja sa 32 filtara dimenzije 3x3, sloja sažimanja s područjem 2x2, konvolucijskog sloja sa 64 filtara dimenzija 3x3, sloja sažimanja s područjem 2x2, potpuno povezanog sloja od 64 neurona sa ReLu aktivacijskim funkcijama te izlaznog sloja od 10 neurona sa softmax aktivacijskom funkcijom.



Sl. 8.1. Primjer konvolucijske neuronske mreže.

II.2. Keras API callbacks

Keras callback je objekt u biblioteci Keras koji se može koristiti za izvođenje različitih akcija tijekom treniranja mreže, poput izvršavanja određenog koda kada se postigne određeni prag točnosti ili gubitka, pohranjivanja modela nakon svake epohe ili promjenu stope učenja tijekom treninga i sl. Metoda `.fit` sadrži argument `callbacks` preko kojeg je procesu učenja moguće predati željene akcije (vidi primjer 8.1.).

Primjer 8.1.

```
from tensorflow import keras

# lista koja sadrzi zeljene callbacks: Tensorboard i ModelCheckpoint
my_callbacks = [
    keras.callbacks.TensorBoard(log_dir = 'logs',
                                update_freq = 100),
    keras.callbacks.ModelCheckpoint(filepath='best_model.h5',
                                    monitor='val_accuracy',
                                    mode='max',
```

```

]                                     save_best_only=True)

# pokreni trening mreže
model.fit(X_train_n,
          y_train,
          epochs = 50,
          batch_size = 64,
          callbacks = my_callbacks,
          validation_split = 0.1)

```

TensorBoard je alat u okviru preglednika za vizualizaciju i praćenje performansi modela u TensorFlow biblioteci. Obično se koristi za praćenje performansi modela na skupu za učenje i validaciju pomoću raznih metrika (npr. točnost klasifikacije). Tensorboard se pokreće na sljedeći način iz terminala:

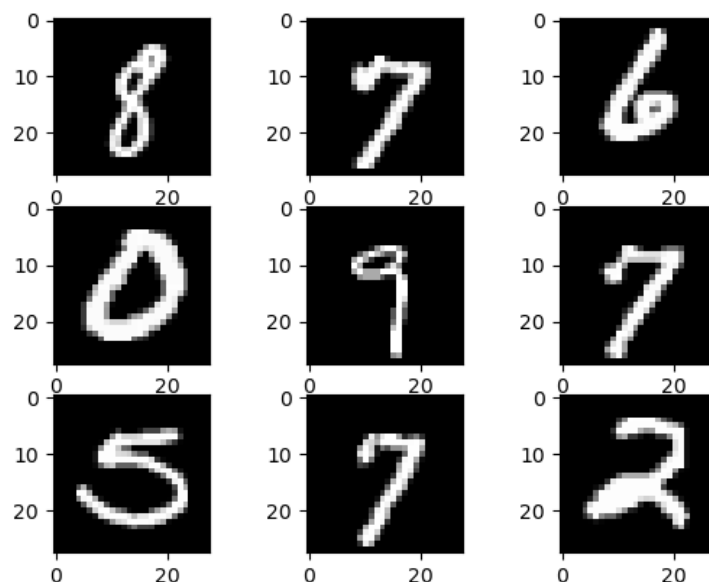
```
tensorboard --logdir logs
```

pri čemu je `logs` putanja do direktorija koji sadrži logove. Nakon toga je potrebno otvoriti adresu <http://localhost:6006/> pomoću web preglednika te će se prikazati odgovarajuća vizualizacija u web sučelju.

ModelCheckpoint je callback koji se koristi tijekom treninga modela kako bi se automatski pohranjivali modeli (ili samo težine modela) tijekom procesa treninga na trajnu memoriju. Moguće ga je konfigurirati tako da se npr. pohrani model nakon svake epohe ili da se pohranjuje samo jedan najbolji model prema odabranoj metrici.

IL.3 MNIST podatkovni skup

U ovoj vježbi razmatra se problem klasifikacije rukom pisanih znamenki. Za izgradnju modela za klasifikaciju rukom pisanih znamenki na raspolaganju je skup podataka pod nazivom MNIST. Ovaj skup sadrži slike rukom pisanih znamenki koje su pisali zaposlenici u *United States Census Bureau* i američki studenti. Slike su zapisane u sivim tonovima odnosno svaki piksel na slici ima vrijednost u rasponu od 0 do 255. Slike su normirane na dimenziju 28 x 28 piksela. Svaka slika ima odgovarajuću oznaku tj. labelu (0, 1, 2, 3, 4, 5, 6, 7, 8, 9). MNIST sadrži skup podataka za učenje od 60,000 slika, te skup podataka za testiranje koji sadrži 10,000 slika. Primjer slika iz skupa podataka za učenje dan je na slici 8.2.



Sl. 8.2. Primjer slika iz skupa MNIST.

III. Priprema za vježbu:

Nema posebne pripreme za vježbu.

IV. Rad na vježbi:

1. Riješite dane zadatke.

Zadatak 1

Potrebno je izgraditi potpuno konvolucijsku neuronsku mrežu na temelju MNIST skupa podataka te izvršiti njenu evaluaciju. U prilogu vježbe nalazi se skripta 8.1. koja učitava MNIST skup podataka. Dopunite skriptu na odgovarajućim mjestima:

- 1) Izgradite konvolucijsku neuronsku mrežu pomoću Keras API. Struktura mreže je dana slikom 8.1. Prilikom treniranja neuronske mreže trebate koristiti 10% podatkovnih primjera za validaciju mreže.
- 2) Dodajte callback kojim ćete pratiti točnost klasifikacije na skupu za učenje i skupu za validaciju u Tensorboard-u.
- 3) Dodajte callback za pohranjivanje najboljeg modela na temelju točnosti klasifikacije na validacijskom skupu.
- 4) Pokrenite proces treniranja mreže te pratite tijek treniranja u Tensorboard-u. Provjerite je li pohranjen na disk najbolji model.
- 5) Izračunajte točnost najboljeg modela (mreže) na skupu podataka za učenje i skupu podataka za testiranje.
- 6) Prikažite matricu zabune na skupu podataka za učenje i na skupu podataka za testiranje. Komentirajte dobivene rezultate.

Zadatak 2

U prilogu vježbe nalazi se skripta 8.2. koja učitava izgrađenu mrežu i sliku `test.png` sa diska. Dodajte u skriptu kod kojim ćete klasificirati sliku pomoću mreže. Promijenite sliku pomoću nekog grafičkog alata (npr. nacrtajte znamenku 2 u Paintu) i ponovo pokrenite skriptu. Pokušajte sa svim znamenkama. Kako položaj ili rotacija znamenke utječe na rezultat klasifikacije?

V. Izvještaj s vježbe

Kao izvještaj s vježbe prihvaća se web link na repozitorij pod nazivom PSU_LV.