CNN projekat – Simpsonovi

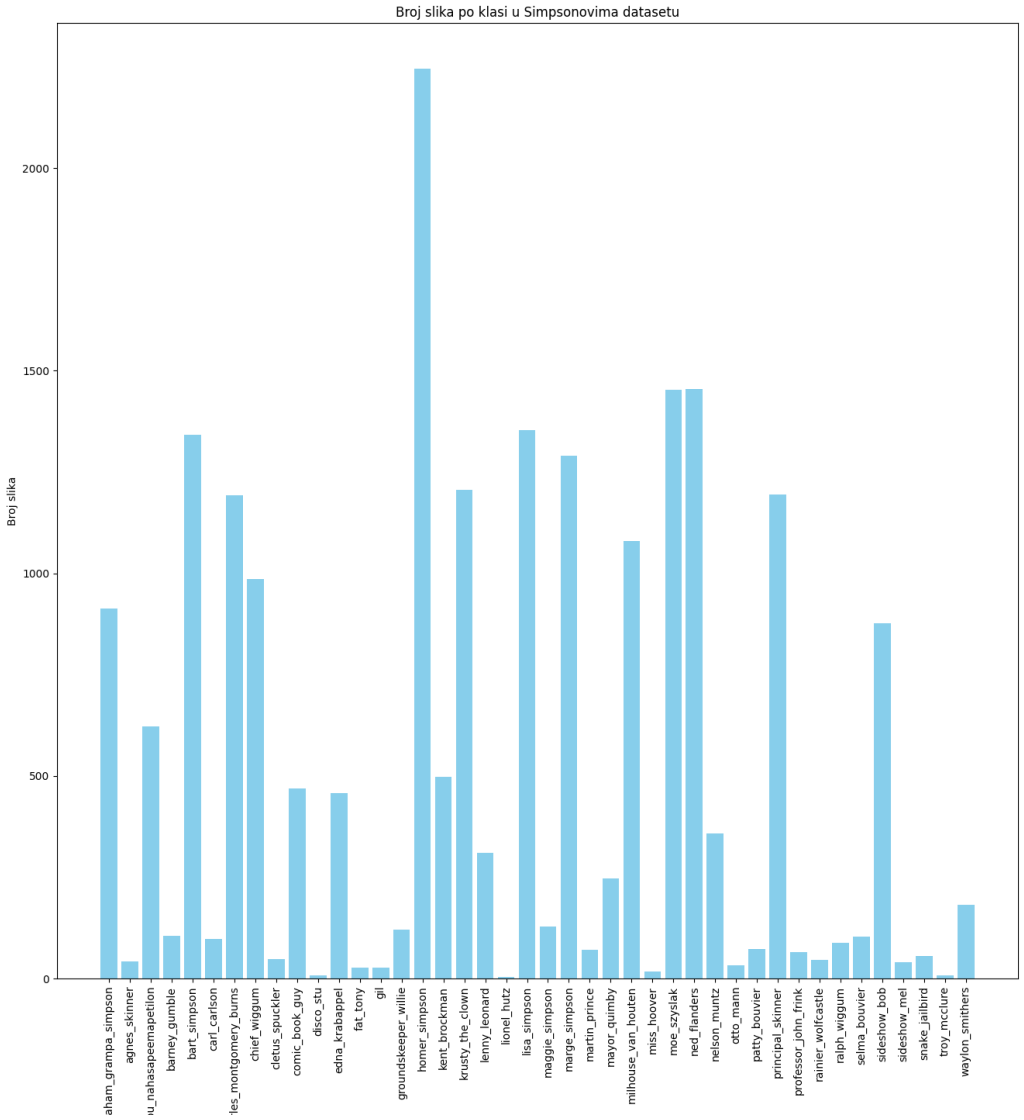
Uvod

U ovom projektu ćemo da prikažemo klasifikaciju karaktera iz popularne serije “Simpsonovi”, primenom konvolucionalne neuralne mreže.

Dataset koji koristimo može se pronaći na sledećem linku:
https://www.kaggle.com/datasets/alexattia/the-simpsons-characters-dataset/

Ulazni podaci

Ulazni podaci su slike, raspoređene u 42 različite klase. Svaka klasa predstavljaja jednog od karaktera iz serije. Podaci nisu balansirani, što se može videti iz broja odbiraka po klasi na grafiku ispod:



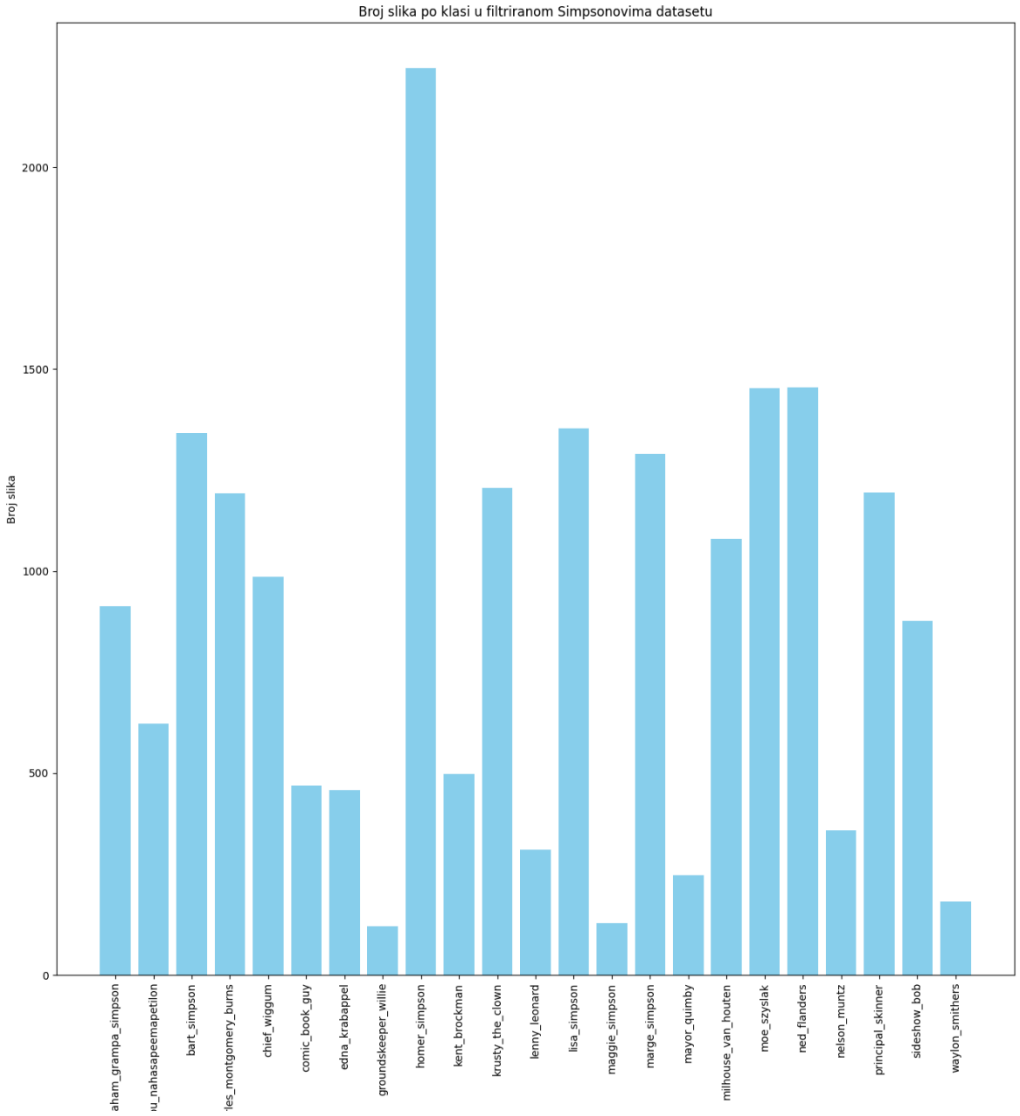
Klasa 'abraham_grampa_simpson' ima 913 slika.
Klasa 'agnes_skinner' ima 42 slika.
Klasa 'apu_nahasapeemapetilon' ima 623 slika.
Klasa 'barney_gumble' ima 106 slika.
Klasa 'bart_simpson' ima 1342 slika.
Klasa 'carl_carlson' ima 98 slika.
Klasa 'charles_montgomery_burns' ima 1193 slika.
Klasa 'chief_wiggum' ima 986 slika.
Klasa 'cletus_spuckler' ima 47 slika.
Klasa 'comic_book_guy' ima 469 slika.
Klasa 'disco_stu' ima 8 slika.
Klasa 'edna_krabappel' ima 457 slika.
Klasa 'fat_tony' ima 27 slika.
Klasa 'gil' ima 27 slika.
Klasa 'groundskeeper_willie' ima 121 slika.
Klasa 'homer_simpson' ima 2246 slika.
Klasa 'kent_brockman' ima 498 slika.
Klasa 'krusty_the_clown' ima 1206 slika.
Klasa 'lenny_leonard' ima 310 slika.
Klasa 'lionel_hutz' ima 3 slika.
Klasa 'lisa_simpson' ima 1354 slika.
Klasa 'maggie_simpson' ima 128 slika.
Klasa 'marge_simpson' ima 1291 slika.
Klasa 'martin_prince' ima 71 slika.
Klasa 'mayor_quimby' ima 246 slika.
Klasa 'milhouse_van_houten' ima 1079 slika.
Klasa 'miss_hoover' ima 17 slika.
Klasa 'moe_szyslak' ima 1452 slika.
Klasa 'ned_flanders' ima 1454 slika.
Klasa 'nelson_muntz' ima 358 slika.
Klasa 'otto_mann' ima 32 slika.
Klasa 'patty_bouvier' ima 72 slika.
Klasa 'principal_skinner' ima 1194 slika.
Klasa 'professor_john_frink' ima 65 slika.
Klasa 'rainier_wolfcastle' ima 45 slika.
Klasa 'ralph_wiggum' ima 89 slika.
Klasa 'selma_bouvier' ima 103 slika.
Klasa 'sideshow_bob' ima 877 slika.
Klasa 'sideshow_mel' ima 40 slika.
Klasa 'snake_jailbird' ima 55 slika.

Klasa 'troy_mcclure' ima 8 slika.
Klasa 'waylon_smithers' ima 181 slika.

Balansiranje podataka

S obzirom da su podaci nebalansirani moraćemo da primenimo neke metode za balansiranje. Prvo što treba da se uradi jeste da se odbace klase koje sadrži manje od 5% od klase sa najvećim brojem odbiraka (izabran je ovaj način jer ostane 23 klasa, dok originalnim pristupom ostane samo 10, što je jako malo od početnih 42).

Naš grafik nakon odbracivanja klasa izgleda ovako:

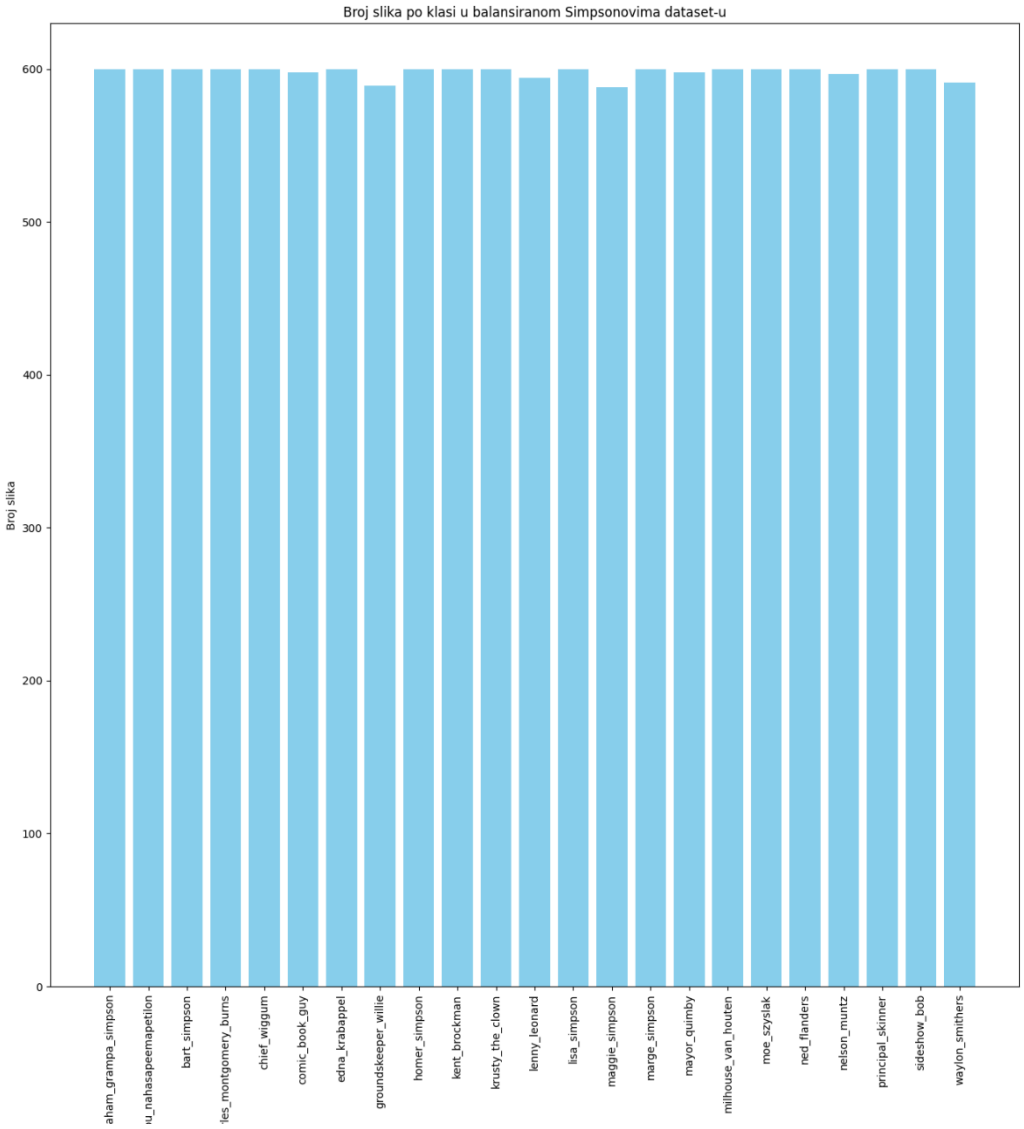


Klasa 'abraham_grampa_simpson' ima 913 slika nakon filtriranja.
Klasa 'apu_nahasapeemapetilon' ima 623 slika nakon filtriranja.
Klasa 'bart_simpson' ima 1342 slika nakon filtriranja.
Klasa 'charles_montgomery_burns' ima 1193 slika nakon filtriranja.
Klasa 'chief_wiggum' ima 986 slika nakon filtriranja.
Klasa 'comic_book_guy' ima 469 slika nakon filtriranja.
Klasa 'edna_krabappel' ima 457 slika nakon filtriranja.
Klasa 'groundskeeper_willie' ima 121 slika nakon filtriranja.
Klasa 'homer_simpson' ima 2246 slika nakon filtriranja.
Klasa 'kent_brockman' ima 498 slika nakon filtriranja.
Klasa 'krusty_the_clown' ima 1206 slika nakon filtriranja.
Klasa 'lenny_leonard' ima 310 slika nakon filtriranja.
Klasa 'lisa_simpson' ima 1354 slika nakon filtriranja.
Klasa 'maggie_simpson' ima 128 slika nakon filtriranja.
Klasa 'marge_simpson' ima 1291 slika nakon filtriranja.
Klasa 'mayor_quimby' ima 246 slika nakon filtriranja.
Klasa 'milhouse_van_houten' ima 1079 slika nakon filtriranja.
Klasa 'moe_szyslak' ima 1452 slika nakon filtriranja.
Klasa 'ned_flanders' ima 1454 slika nakon filtriranja.
Klasa 'nelson_muntz' ima 358 slika nakon filtriranja.
Klasa 'principal_skinner' ima 1194 slika nakon filtriranja.
Klasa 'sideshow_bob' ima 877 slika nakon filtriranja.
Klasa 'waylon_smithers' ima 181 slika nakon filtriranja.

Kao što možemo videti sada smo sveli ukupan broj klasa sa 42 na 23.

Iako smo uklonili klase idalje postoji problem nebalansiranih podataka. Ovo dalje rešavamo oversamplovanjem i undersamplovanjem tako da svaka klasa ima 600 odbiraka. U slučaju da radimo oversamplovanje radimo augmentaciju podataka tako što biramo nasumično sliku kojoj ćemo primeniti augmentaciju (rotiranje, pomeranje, odsecanje, zumiranje, okretanje i dopunjavanje nedostajućih piksela) i napraviti kopiju koja je izmenjena sve dok ne stignemo do 600 odbiraka te klase. Za undersampling brišemo nasumično podatke dok ne stignemo do 600 odbiraka.

Nakon tog procesa dobijamo grafik koji izgleda ovako:



Klasa 'abraham_grampa_simpson' ima 600 slika nakon augmentacije.
Klasa 'apu_nahasapeemapetilon' ima 600 slika nakon augmentacije.
Klasa 'bart_simpson' ima 600 slika nakon augmentacije.
Klasa 'charles_montgomery_burns' ima 600 slika nakon augmentacije.
Klasa 'chief_wiggum' ima 600 slika nakon augmentacije.
Klasa 'comic_book_guy' ima 598 slika nakon augmentacije.
Klasa 'edna_krabappel' ima 600 slika nakon augmentacije.
Klasa 'groundskeeper_willie' ima 589 slika nakon augmentacije.
Klasa 'homer_simpson' ima 600 slika nakon augmentacije.
Klasa 'kent_brockman' ima 600 slika nakon augmentacije.
Klasa 'krusty_the_clown' ima 600 slika nakon augmentacije.
Klasa 'lenny_leonard' ima 594 slika nakon augmentacije.
Klasa 'lisa_simpson' ima 600 slika nakon augmentacije.
Klasa 'maggie_simpson' ima 588 slika nakon augmentacije.
Klasa 'marge_simpson' ima 600 slika nakon augmentacije.
Klasa 'mayor_quimby' ima 598 slika nakon augmentacije.
Klasa 'milhouse_van_houten' ima 600 slika nakon augmentacije.
Klasa 'moe_szyslak' ima 600 slika nakon augmentacije.
Klasa 'ned_flanders' ima 600 slika nakon augmentacije.
Klasa 'nelson_muntz' ima 597 slika nakon augmentacije.
Klasa 'principal_skinner' ima 600 slika nakon augmentacije.
Klasa 'sideshow_bob' ima 600 slika nakon augmentacije.
Klasa 'waylon_smithers' ima 591 slika nakon augmentacije.

Napokon, dobijamo dataset s kojim možemo raditi, podaci su balansirani.

Prikaz primera iz svake klase

Podela na skupove

Deljenje podataka na ove skupove je važno jer omogućava objektivnu procenu performansi modela i njegovu sposobnost generalizacije na novim podacima. Bez ovakvog deljenja, postoji rizik od preobučavanja modela, gde bi model mogao savršeno raditi na poznatim podacima ali bi bio neupotrebljiv za bilo kakve nove podatke.

Imajući ovo u vidu delimo podatke na skup za treniranje (60%), skup za validaciju (20%) i skup za testiranje (20%)

```
Ukupno slika u trening skupu: 9164
Ukupno slika u validacionom skupu: 2291
Ukupno slika u testnom skupu: 2300
```

Predprocesiranje podataka

Za predprocesiranje, osim augmentacije podataka zbog oversampling metode, imamo sekvencijalni model koji sadrzi sledeće slojeve:

```
from keras import Sequential
from keras import layers

data_augmentation = Sequential([
    layers.Rescaling(1./255, input_shape=(64, 64, 3)),
    layers.Resizing(height=64, width=64)
])
```

Primenjujemo normalizaciju, tj. skaliranje slika sa opsega [0, 255] na [0, 1] jer računar bolje obrađuje ovakve podatke, takodje vršimo promenu veličine slike na 64x64 što će nam sačuvati dosta vremena prilikom treniranja jer su slike manje rezolucije.

Formiranje modela neuralne mreže

Naš model sadrži 6 konvolucionih slojeva, gde svaki drugi prati po jedan maxpooling sloj radi izvlačenja karakteristika, takodje nakon maxpooling sloja imamo i dropout sloj koji nam odbacuje random neurone i smanjuje mogućnost da dodje do preobučavanja.
Kriterijumska funkcija koju koristimo je SparseCategoricalCrossentropy koja daje najbolje performanse za klasifikaciju.
Aktivacione funkcije konvolucionih slojeva su ReLu, jer je on generalno i najbolji za konvolucione neuralne mreže, dok je aktivaciona funkcija poslednjen sloja softmax, kako bi omogućili klasifikaciju višeklasnih podataka, što je naš slučaj jer imamo 23 klase.
Za metodu optimizacije koristimo Adam koji adaptivno prilagođava konstantu obučavanja, pri čemu smo kao početnu stavili 0.001

Specifikacije neuralne mreže su sledeće:

Layer (type)	Output Shape	Param #
sequential (Sequential)	(None, 64, 64, 3)	0
conv2d (Conv2D)	(None, 64, 64, 32)	896
conv2d_1 (Conv2D)	(None, 62, 62, 32)	9248
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
dropout (Dropout)	(None, 31, 31, 32)	0
conv2d_2 (Conv2D)	(None, 31, 31, 64)	18496
conv2d_3 (Conv2D)	(None, 29, 29, 64)	36928
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 64)	0
dropout_1 (Dropout)	(None, 14, 14, 64)	0
conv2d_4 (Conv2D)	(None, 14, 14, 256)	147712
conv2d_5 (Conv2D)	(None, 12, 12, 256)	590080
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 256)	0
dropout_2 (Dropout)	(None, 6, 6, 256)	0
flatten (Flatten)	(None, 9216)	0
dense (Dense)	(None, 1024)	9438208
dropout_3 (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 23)	23575

Total params: 10265143 (39.16 MB)
Trainable params: 10265143 (39.16 MB)
Non-trainable params: 0 (0.00 Byte)

Preobučavanje

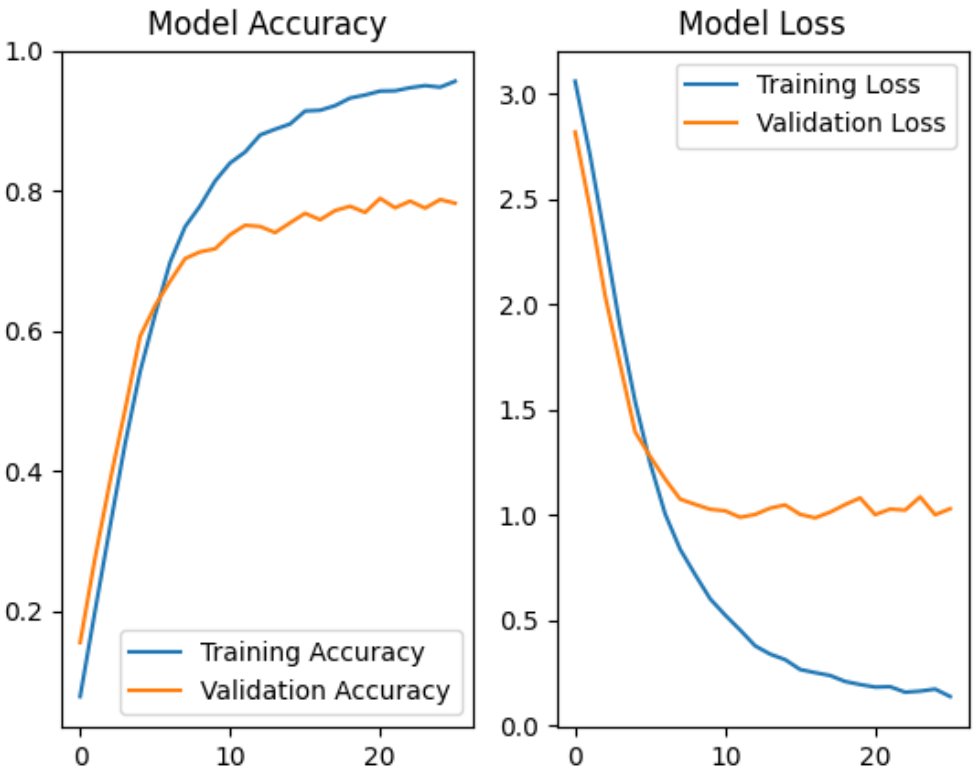
Preobučavanje predstavlja model koji ima visoku tačnost na trening skupu podataka, ali značajno slabije performanse na validacionom ili testnom skupu.
Do preobučavanja može doći zbog malog skupa podataka, prekomplikovanog modela, nedovoljne regularizacije ili predugog obučavanja.

Metodi zaštite koje smo do sad primenili su Dropout slojevi u mreži a sada dodajemo i EarlyStopping metod čiji je cilj da zaustavi treniranje u slučaju da dođe do preobučavanja.

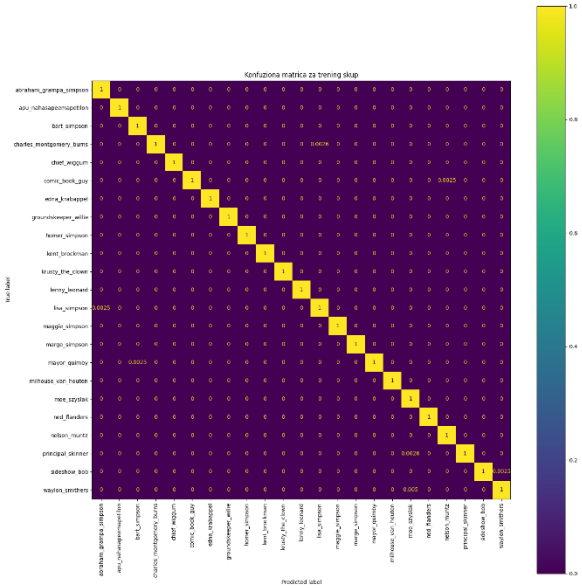
```
callback = EarlyStopping(
    patience=5,
    monitor="val_accuracy",
    mode="max"
)
```

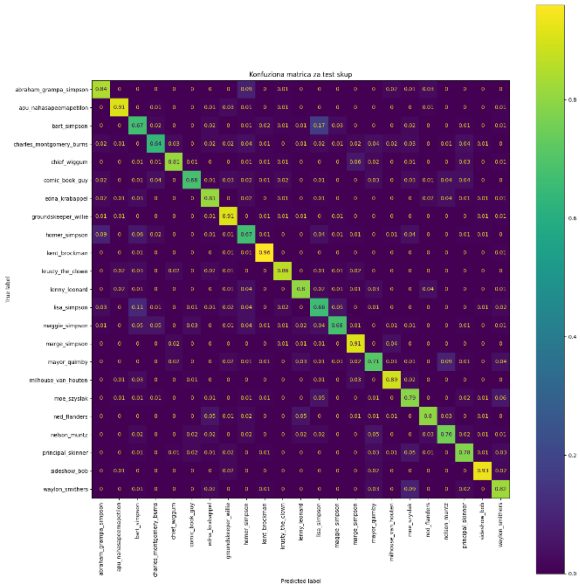
Performanse finalno obučenog modela

Grafik performanse neuralne mreže kroz epohe obučavanja nad trening i validacionim skupom:



Matrica konfuzije za trening i test skup:






```
Tačnost modela na trening skupu je: 99.91%
Tačnost modela na test skupu je: 79.52%
```

Primeri dobro i loše klasifikovanih primera dataseta: