

Izvještaj

Matija Roginić

Uvod:

Cilj rada na ovom predmetu je upoznavanje s tehnologijama i modelima koji se koriste za računalni vid, prepoznavanje objekata i segmenata slike. Prvo što sam napravio je istraživanje koji modeli su trenutno najbolji za dani zadatak (State-of-the-art). U dogovoru s mentorom, sljedeći korak je bio odabir modela i primjena na konkretnom problemu. Za model sam izabrao YOLOv8 što je trenutno najnovija i najbolja verzija od niza YOLO modela. Dataset koji sam koristio je dataset s UNO kartama koji je javno dostupan i kojeg sam preuzeo s Roboflowa.^[1]

Opis modela:

Model YOLOv8 je najnoviji član YOLO „obitelji“ modela. S trenutno najboljim performansama od svih YOLO modela. YOLO predstavlja okvir za treniranje modela za bilo koji od 3 zadatka (detekciju objekata, segmentaciju dijelova slike, te klasifikaciju slika). Postoji 5 verzija YOLOv8 modela, a to su redom: nano, small, medium, large i extra large. Uzlazno su poredani po veličini i točnosti koju pružaju. Dakle, nano je najmanji i najmanje točan od navedenih, ali je i najbrži. S druge strane, extra large je najveći i najtočniji, ali i najsporiji. Verzija koju sam izabrao za ovaj zadatak je nano. Općenito, YOLO modeli su brzi modeli u usporedbi s drugim modelima koji se koriste u ovom području. Brzina se temelji na tome što oni svaku sliku pogledaju samo jednom i zatim se prebacuju na sljedeću. Iz toga dolazi i sam naziv modela: YOLO – You only look once. Model sam po sebi je dosta složen, s kombinacijom konvolucijskih, 2D konvolucijskih te pooling slojeva koji su posloženi u dva dijela (glavu i kralježnicu modela). Na izlazu modela nalazi se sloj Detect koji se sastoji od dvije obične konvolucije, zatim 2D konvolucije te dijela koji izračunava gubitak (box loss i clf loss).^[2]

Treniranje modela:

Osim same klasifikacije slika, radio sam i usporedbu kako se model ponaša s različitim veličinama dataseta, odnosno specifično skupa za učenje. Radio sam s 4 različite veličine, prva sadrži sve slike koje su originalno dostupne u skupu za učenje, zatim sam skup smanjio na pola početnog, pa četvrtinu početnog, te na kraju desetinu početnog. Za svaku veličinu skupa za učenje sam ponovno inicijalizirao model te pokrenuo trening.

	from	n	params	module	arguments
0	-1	1	464	ultralytics.nn.modules.conv.Conv	[3, 16, 3, 2]
1	-1	1	4672	ultralytics.nn.modules.conv.Conv	[16, 32, 3, 2]
2	-1	1	7360	ultralytics.nn.modules.block.C2f	[32, 32, 1, True]
3	-1	1	18560	ultralytics.nn.modules.conv.Conv	[32, 64, 3, 2]
4	-1	2	49664	ultralytics.nn.modules.block.C2f	[64, 64, 2, True]
5	-1	1	73984	ultralytics.nn.modules.conv.Conv	[64, 128, 3, 2]
6	-1	2	197632	ultralytics.nn.modules.block.C2f	[128, 128, 2, True]
7	-1	1	295424	ultralytics.nn.modules.conv.Conv	[128, 256, 3, 2]
8	-1	1	460288	ultralytics.nn.modules.block.C2f	[256, 256, 1, True]
9	-1	1	164608	ultralytics.nn.modules.block.SPPF	[256, 256, 5]
10	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
11	[-1, 6]	1	0	ultralytics.nn.modules.conv.Concat	[1]
12	-1	1	148224	ultralytics.nn.modules.block.C2f	[384, 128, 1]
13	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
14	[-1, 4]	1	0	ultralytics.nn.modules.conv.Concat	[1]
15	-1	1	37248	ultralytics.nn.modules.block.C2f	[192, 64, 1]
16	-1	1	36992	ultralytics.nn.modules.conv.Conv	[64, 64, 3, 2]
17	[-1, 12]	1	0	ultralytics.nn.modules.conv.Concat	[1]
18	-1	1	123648	ultralytics.nn.modules.block.C2f	[192, 128, 1]
19	-1	1	147712	ultralytics.nn.modules.conv.Conv	[128, 128, 3, 2]
20	[-1, 9]	1	0	ultralytics.nn.modules.conv.Concat	[1]
21	-1	1	493056	ultralytics.nn.modules.block.C2f	[384, 256, 1]
22	[15, 18, 21]	1	754237	ultralytics.nn.modules.head.Detect	[15, [64, 128, 256]]

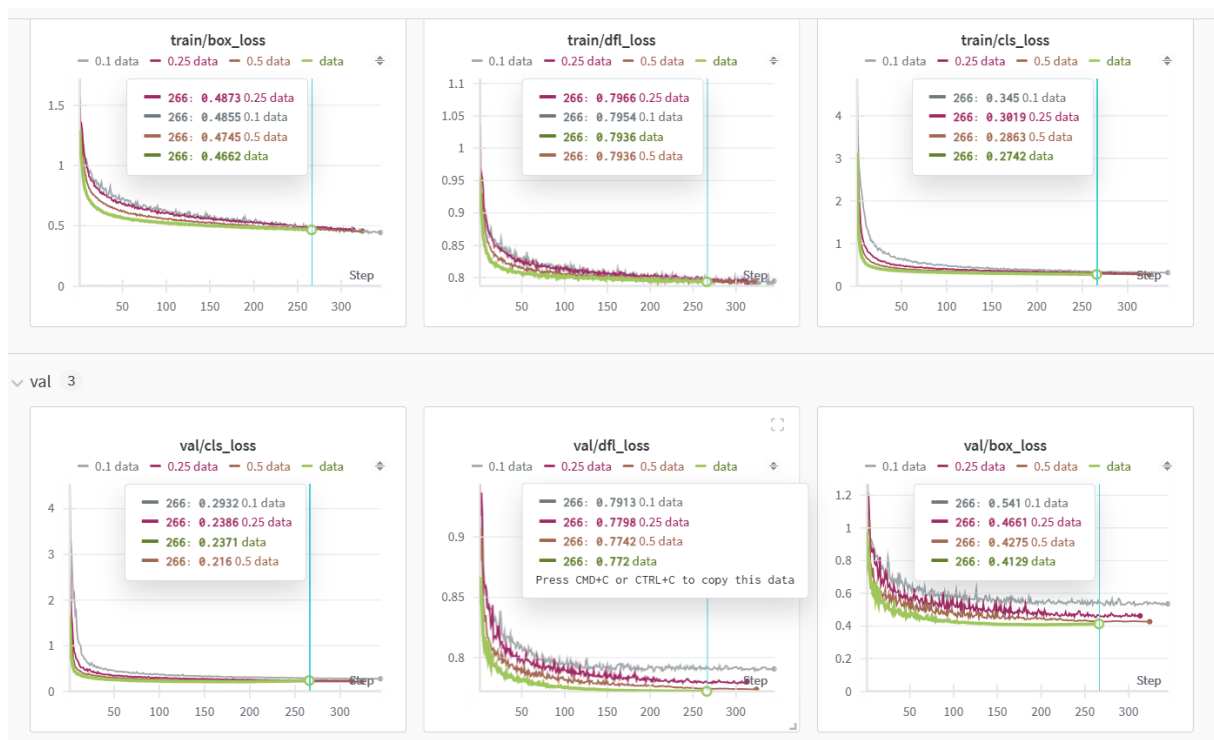
Model summary: 225 layers, 3013773 parameters, 3013757 gradients, 8.2 GFLOPs

Slika 2: Izgled modela

Slika iznad prikazuje sam izgled modela, sloj po sloj, broj parametara, vrstu sloja i argumente koje sloj prima. Model je inicijaliziran bez mijenjanja hiperparametara, odnosno oni su onakvi kakvi su bili preddefinirani u modelu. Ono što sam promijenio je dimenzija slike, na 416×416 što je dimenzija slika u našem datasetu. Također, postavio sam broj epoha na maksimalno 500, s tim što ako u 50 posljednjih epoha nema pomaka u točnosti.

Rezultati i analiza:

Nakon provedenog treninga, ono što vidimo je da niti u jednom od 4 slučaja veličine dataseta trening ne traje punih 500 epoha. Najkraće traje kada model treniramo sa svim slikama (266 epoha), a najdulje kad imamo najmanji dataset (345 epoha). Između se nalaze slučajevi s četvrtinom i polovinom originalnog dataseta kada trening traje 313 i 324 epohe.



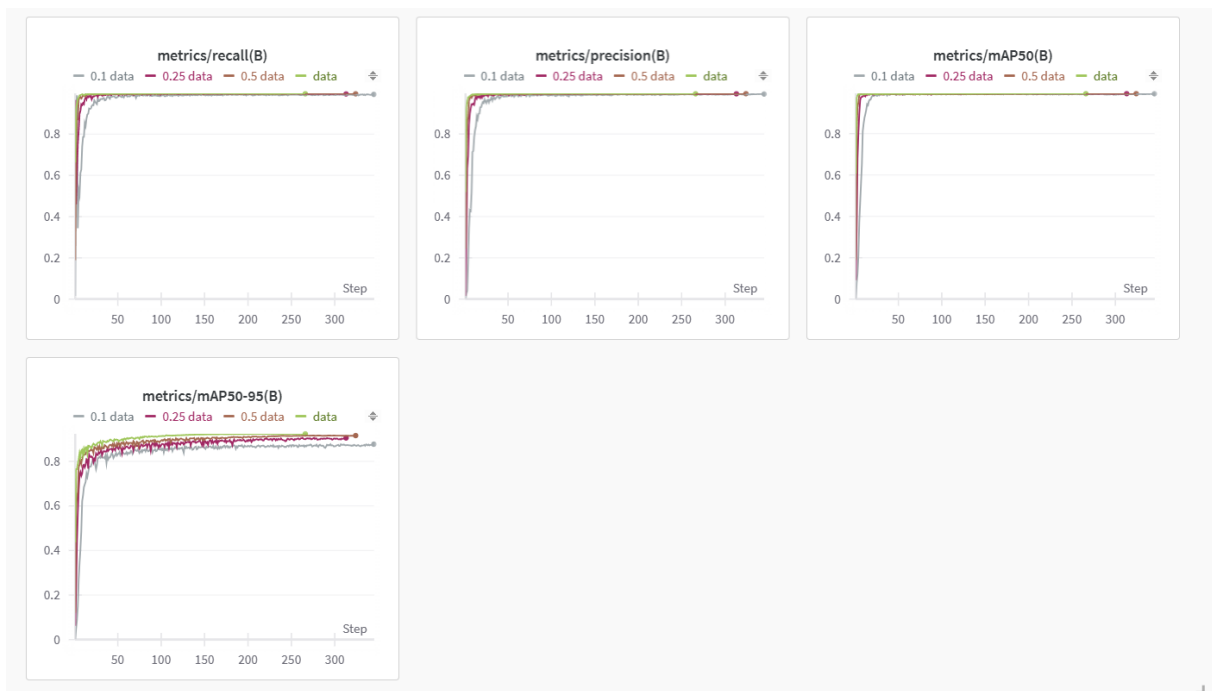
Slika 3: Grafovi pogreška na skupu za trening i validaciju

Prvo da objasnimo vrste funkcija gubitka čiji grafovi su gore prikazani. Box loss nam govori o tome kolika je greška u predviđanju granica objekata na slici, Cls loss prikazuje pogrešku u predviđanju klasa karata sa slike, dok Dfl loss mjeri grešku u tome je li objekt prisutan na slici ili nije. Iz grafova možemo vidjeti da se vrijednost funkcija gubitka smanjuje kontinuirano na skupu za treniranje, dok na skupu za validaciju vidimo da funkcija u slučaju kad imamo cijeli dataset doseže minimum kod Cls i Box lossa u određenom trenutku, a zatim blago raste. Kod Cls lossa taj minimum se postiže u 190. epohi i iznosi 0.2103, a kod Box lossa minimum se postiže u 200. epohi i iznosi 0.4076. Kod Dfl lossa iznos se smanjuje, a zatim stagnira te se zaustavlja na vrijednosti 0.772. Vrijednosti za ostale veličine dataseta prikazat ću u tablici ispod.

Tablica 1: Vrijednosti funkcija gubitaka za svaku veličinu dataseta

Omjer veličine dataseta u odnosu na originalni	Box loss		Cls loss		Dfl loss	
	Minimum	Krajnje	Minimum	Krajnje	Minimum	Krajnje
0.1	0.5247	0.5356	0.2758	0.2758	0.7891	0.7906
0.25	0.4596	0.4621	0.2142	0.2142	0.7789	0.7803
0.5	0.4265	0.4265	0.2316	0.2316	0.773	0.7737
1.0	0.4076	0.4129	0.2103	0.2371	0.772	0.772

Iz tablice iznad vidimo da ako promatramo sve veličine dataseta, funkcija gubitka u nekim slučajevima doseže minimum pa zatim blago raste do kraja, dok u nekim slučajevima pada do kraja.



Slika 4: Grafovi metrika modela

Recall (odziv) je mjera koja iznosi omjer između broja stvarno pozitivnih i zbroja stvarno pozitivnih i lažno negativnih. Precision (preciznost) mjera koja iznosi omjer između broja stvarno pozitivnih i zbroja stvarno pozitivnih i lažno pozitivnih. Iz grafova vidimo da su obje mjere u ovom slučaju iznad 0.99 za sve veličine dataseta. To znači da je broj lažno negativnih i lažno pozitivnih primjera jako malen. Sljedeća metrike koje promatramo je mjera mAP50 i mAP50-95. mAP50 je mjera koja promatra omjer presjeka i unije između predviđenih i stvarnih okvira za svaki objekt na slici. U ovom slučaju taj omjer je fiksiran na 0.5, pa mjera govori o tome u kojem broju slučajeva je omjer presjeka i unije veći ili jednak od 0.5. Posljednja metrika, mAP50-95 uzima prosjek mAP mjera s time da omjer presjeka i unije povećavamo od 0.5 do 0.95 s korakom od 0.05. Iz grafova vidimo da mAP50 mjera iznosi blizu 1 za sve 4 veličine dataseta, dok mAP50-95 mjera prelazi 0.9 za sve veličine dataseta osim za onu gdje je veličina dataseta 10% originalne, a maksimum doživljava u slučaju cijelog dataseta gdje iznosi 0.9235. To nam govori da model postavlja okvir na približno iste koordinate u odnosu na one stvarne. Za daljnju analizu možemo pogledati ostale grafove na sljedećem linku: [YOLOv8 Workspace – Weights & Biases \(wandb.ai\)](https://wandb.ai/Weights&Biases/YOLOv8-Workspace)

Zaključak:

Za kraj, moramo komentirati rezultate. Iz analize rezultata možemo zaključiti da model dobro obavlja svoju zadaću te da su vrijednosti svih mjera iznimno visoke, jedina mjera koja daje malo niže rezultate je mAP50-95 što je očekivano jer je ona i najstroža. Iz toga zaključujemo da je model kojeg sam koristio prikladan za ovaj dataset, te da je dataset sam po sebi vrlo dobar, odnosno da nema puno šumova u podacima. Naravno, sljedeći korak bila bi primjena ovog modela na nekom datasetu koji nije ovako idealan, no za smo upoznavanje s modelom, ovaj dataset je odigrao dobru ulogu.

Literatura:

[1] Joseph Nelson, Uno Cards Computer Vision Project, 18.12.2023., [Uno Cards Dataset > Overview \(roboflow.com\)](#)

[2] Sovit Rath, YOLOv8 : Comprehensive Guide to State Of The Art Object Detection, 22.01.2024., [YOLOv8 : Comprehensive Guide to State of the Art Object Detection \(learnopencv.com\)](#)