

Usporedba metoda SimCLR i BYOL za samonadzirano učenje

Panić, Iva

Zagreb, siječanj 2022.

Sadržaj

1. Uvod	3
2. Samonadzirano učenje	4
3. Metoda SimCLR	4
4. Metoda BYOL	6
5. Eksperimentalni rezultati	7
6. Zaključak	9
7. Literatura	10

1. Uvod

U zadnjih par godina, područje umjetne inteligencije postiglo je ogroman napredak u razvoju sustava koji mogu učiti iz velikih količina pažljivo označenih podataka. Ovakav način učenja – nadzirano učenje – ima dokazano dobre rezultate u obučavanju modela koji se iznimno dobro ponašaju u zadatku za koji su obučeni. Nažalost, postoji ograničenje koliko područje umjetne inteligencije može napredovati koristeći samo nadzirano učenje zbog jedne velike razlike između ljudi i strojeva – ljudima zdrav razum pomaže da naučimo nove zadatke, poput prepoznavanja nekog objekta iz slike, iz relativno malog broja označenih slika zbog toga što se oslanjamo na prethodno stečeno znanje o tome kako svijet funkcionira. Stroj zahtijeva puno više slika da nauči prepoznati taj objekt i opet postoji mogućnost da ga neće moći prepoznati ako ga vidi iz nekog prethodno neviđenog kuta jer stroj nema spoznaju o svijetu te zaključuje samo na temelju označenih slika koje je vidio. Bilo bi savršeno da je objekt iz tog kuta već označen, no tu nastaje problem: nemoguće je označiti sve na svijetu. Kako učiniti da približimo način učenja strojeva ljudskom načinu učenja? Samonadzirano učenje jedan je od najperspektivnijih načina za izgradnju pozadinskog znanja sličnog onom ljudskom i omogućuje sustavima umjetne inteligencije da uče bez potrebe za ogromnim brojem označenih podataka. U ovom seminaru obradit će se dvije metode za samonadzirano učenje: kontrastivna metoda SimCLR i nekontrastivna metoda BYOL.

2. Samonadzirano učenje

Samonadzirano učenje metoda je strojnog učenja koja uči iz neoznačenih podataka.

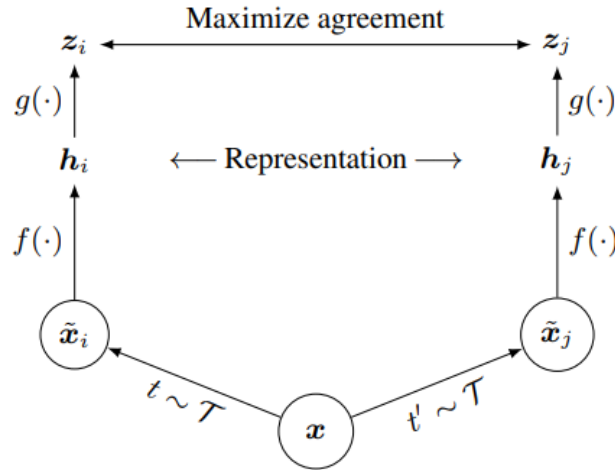
Algoritam samonadziranog učenja uči u dva koraka: 1) zadatak se rješava na temelju pseudo-oznaka koje pomažu pri inicijalizaciji mrežnih težina i 2) stvarni zadatak izvodi se uz nadzor ili nenadgledano učenje. Glavna prednost samonadziranog učenja nad nadziranim učenjem je da se može odvijati s manje podataka i s podacima niže kvalitete. Podaci za učenje mogu se podijeliti na pozitivne i negativne primjere – pozitivni odgovaraju ciljnim primjerima, a negativni su svi ostali. Te dvije vrste podataka koriste se u *kontrastivnom* samonadziranom učenju. S druge strane, za učenje se mogu koristiti i samo pozitivni primjeri. Takav način samonadziranog učenja naziva se *nekontrastivnim*.

3. Metoda SimCLR

SimCLR jednostavni je programski okvir za kontrastivno učenje vizualnih reprezentacija.

Ukratko, SimCLR funkcionira na sljedeći način:

- Izabere se slika iz skupa podataka i na nju se primijene nasumično odabrane transformacije (transformacijska funkcija), poput izrezivanja i zrcaljenja – na taj način dobijemo par izmijenjenih slika x_i i x_j .
- Svaka od tih slika „provuče“ se kroz enkoder kako bismo dobili njihove reprezentacije.
- Na reprezentacije se primijeni nelinearni potpuno povezani sloj kako bismo dobili nove reprezentacije z .
- Cilj je maksimizirati sličnost između z_i i z_j za originalnu sliku.



Slika 1. Arhitektura metode SimCLR. Slika je preuzeta iz [1].

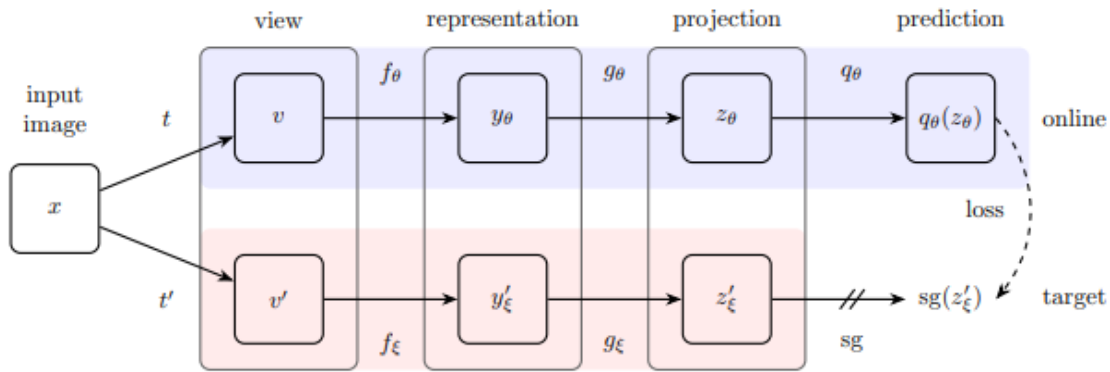
Na početku se generiraju mini-grupe veličine N iz našeg skupa slika. Na svaku sliku iz mini-grupe se primijene dvije transformacije t i t' iz obitelji transformacija τ – dobije se skup od $2N$ izmijenjenih slika. Svaki par izmijenjenih slika provuče se kroz par enkodera $f(\bullet)$ arhitekture ResNet s jednako postavljenim težinama i na taj način dobijemo vektore h_i i h_j – reprezentacije slika. Reprezentacije h_i i h_j tada ulaze u mrežu $g(\bullet)$, tzv. *projection head* – višeslojni perceptron s jednim skrivenim slojem - koja ih preslikava u prostor gdje se može primijeniti kontrastivni gubitak. Na taj način se dobivaju reprezentacije $z_i = g(h_i) = W^{(2)}\sigma(W^{(1)}h_i)$, gdje je σ nelinearna ReLU aktivacijska funkcija skrivenog sloja. Za izračunavanje kontrastivnog gubitka, nasumično se odabire N slika iz skupa podataka na koje se primijene nasumično odabrane transformacije te se na taj način dobije $2N$ izmijenjenih primjera. Odabere se jedan par – pozitivni primjer, a ostalih $2(N - 1)$ primjera se nazivaju negativnim primjerima. Za svaki od negativnih primjera izmjeri se sličnost s pozitivnim primjerom primjenom funkcije softmax, tj. računa se vjerojatnost da su izmijenjene slike pozitivnog primjera najbližije. Funkcija gubitka računa se kao negativna log-vrijednost dobivene vjerojatnosti. Opisani kontrastivni gubitak naziva se NT-Xent gubitak.

4. Metoda BYOL (Bootstrap Your Own Latent)

Postavlja se pitanje: je li moguće naučiti vizualne reprezentacije bez uvođenja negativnih primjera? Metoda BYOL pokazuje da – jest. Prvi korak u metodi je jednak kao i u metodi SimCLR – generiraju se mini-grupe veličine N i na svaki primjer iz mini-grupe se primijene dvije transformacije t i t' kako bi se dobile izmijenjene inačice primjera. Svaka od izmijenjenih inačica šalje se na ulaz jedne od dviju neuronskih mreža – *online* i *ciljnu* mrežu. Online mreža definira se težinama θ i sastoji se od tri faze: enkodera f_θ , projektora g_θ i prediktora q_θ . Ciljna mreža ima jednaku arhitekturu, no koristi drugačije težine ξ koje su definirane kao prosječno eksponencijalno kretanje parametara θ . Točnije, nakon svakog koraka u učenju osvježavaju se na sljedeći način:

$$\xi \leftarrow \tau \xi + (1 - \tau) \theta,$$

gdje je $\tau \in [0, 1]$ stopa propadanja.



Slika 2. Arhitektura metode BYOL. Preuzeto iz [2].

Nadalje, izmijenjene slike se, zajedno s parametrima, dovode na ulaz enkoderima kako bi se dobile reprezentacije y_θ i y_ξ . Reprezentacije se, kao u metodi SimCLR, provode kroz višeslojni

perceptron zbog projekcije u manji prostor. Zadnja faza, prediktor, primijenjuje se samo na online mrežu te vraća predviđanje $q_{\theta}(z_{\theta})$ od z'_{ξ} te L2-normalizirani z'_{ξ} na sljedeći način:

$$\overline{q_{\theta}(z_{\theta})} \approx q_{\theta}(z_{\theta}) / \|q_{\theta}(z_{\theta})\|_2,$$

$$\overline{z'_{\xi}} \approx z'_{\xi} / \|z'_{\xi}\|_2.$$

Uz ove vrijednosti definira se gubitak kao srednja kvadratna greška između normaliziranih predviđanja i izlaza iz ciljne mreže. Tijekom svakog koraka učenja, radi se stohastička optimizacija kako bi se minimizirao gubitak $L^{\text{BYOL}}_{\theta, \xi}$ ovisno o θ :

$$L^{\text{BYOL}}_{\theta, \xi} = L_{\theta, \xi} + L'_{\theta, \xi},$$

gdje je $L'_{\theta, \xi}$ gubitak dobiven prolazom v' kroz online mrežu i v kroz ciljnu mrežu,

$$\theta \leftarrow \text{optimizer}(\theta, \nabla_{\theta} L^{\text{BYOL}}_{\theta, \xi}, \eta),$$

gdje je *optimizer* optimizator i η je stopa učenja.

5. Eksperimentalni rezultati

Skup podataka CIFAR-10 u eksperimentu se koristio za učenje modela. Učenje modela izvršeno je na platformi Google Colab koristeći grafičku jedinicu NVIDIA Tesla K80. Tijekom eksperimenta, mjerila se uspješnost modela ovisno o broju epoha i veličini mini-grupe, a svi parametri oba modela bili su podešeni kao u izvornim radovima [1] i [2]. Ocjenjivanje modela izvršilo se linearnom evaluacijom.

Radi usporedbe, broj epoha i veličine mini-grupa bile su jednake kod oba modela tijekom eksperimenta. Mjerila se uspješnost učenja kroz 20, 40, 60, 80 i 100 epoha te s veličinom mini grupe od 64, 128 i 256.

Broj epoha	SimCLR (Top 1)	BYOL (Top 1)
20	26.56	27.84
40	28.34	29.01
60	29.42	29.63
80	33.17	31.22
100	34.93	34.78

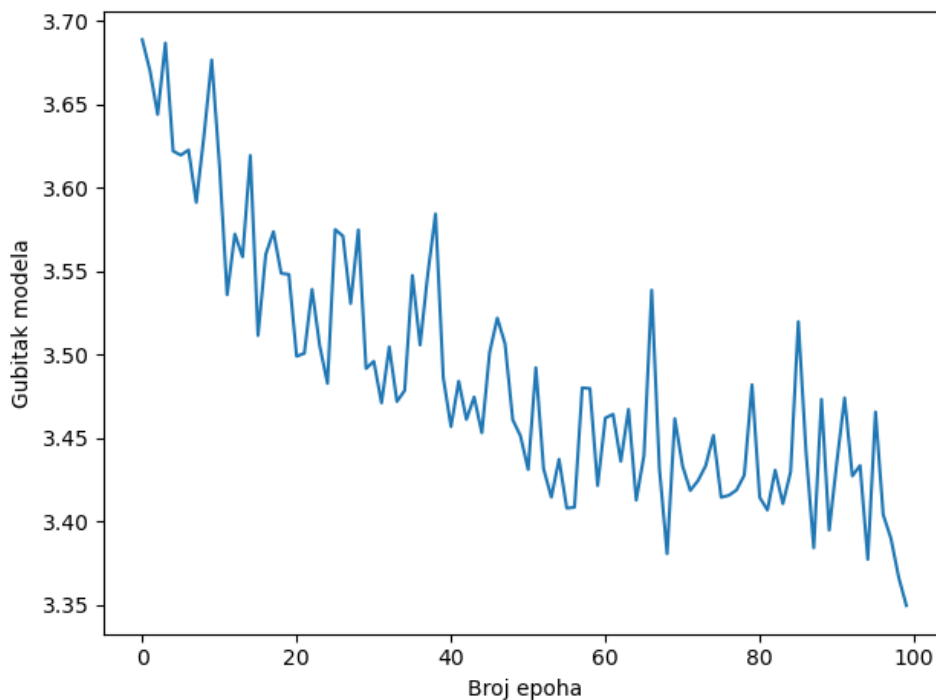
Tablica 1. Top-1 točnost koristeći linearnu evaluaciju u ovisnosti o broju epoha učenja.

U tablici 1. prikazan je odnos broja epoha i točnosti modela s podešenom veličinom mini-grupa od 64. Pokazalo se da u prosjeku BYOL daje nešto bolje rezultate od SimCLR-a, no i dalje su rezultati relativno slični – aritmetička sredina točnosti SimCLR-a kroz epohe u tablici jest 30.28, dok je taj broj za BYOL 30.50. Oba modela pogoduje veći broj epoha te nakon osamdesete epohe se bilježi veći rast točnosti modela SimCLR.

Veličina mini-grupe	SimCLR (Top 1)	BYOL (Top 1)
64	34.93	34.78
128	36.74	36.52
256	39.13	38.63

Tablica 2. Top-1 točnost koristeći linearnu evaluaciju u ovisnosti o veličini mini-grupe.

U tablici 2. prikazan je odnos veličine mini-grupa i točnosti modela s podešenim brojem epoha za učenje od 100. Veličina mini-grupe također utječe na točnost: pokazalo se da s udvostručenjem točnost primjetno raste.



Slika 3. Prikaz kretanja gubitka tijekom učenja modela SimCLR.

Na slici 3. prikazan je gubitak modela SimCLR tijekom 100 epoha učenja s veličinom mini-grupe od 256. Vidi se da kroz epohe gubitak oscilira između vrijednosti, no amplituda oscilacija se s vremenom učenja smanjuje te vrijednost gubitka konvergira.

6. Zaključak

Iako su za samopodržano učenje uobičajenije kontrastivne metode, metoda BYOL pokazuje rezultate otprilike jednake metodi SimCLR, u nekim slučajevima čak i bolje – posebno u učenju s manjim brojem epoha. Vrlo reprezentativni rezultati bi se mogli postići dužim učenjem modela, većim mini-grupama te finim ugađanjem, što se može vidjeti u [1] i [2]. Od novijih radova, metoda DetCon [3] pokazuje izvrsne rezultate, a ona se temelji upravo na

metodama obrađanima u ovom radu. Ovakav način označavanja primjera s razlogom je trenutno zanimljiva tema i može se očekivati da će zbilja pridonijeti u razvoju područja umjetne inteligencije, a i šire.

7. Literatura

- [1] T. Chen, S. Kornblith, M. Norouzi, G. Hinton: *A Simple Framework for Contrastive Learning of Visual Representations*
- [2] J.B. Grill, F. Strub, F. Altche, P. H. Richemond: *Bootstrap Your Own Latent A New Approach to Self-Supervised Learning*
- [3] O.J. Henaff, S. Koppula, J. B. Alayrac: *Efficient Visual Pretraining with Contrastive Detection*
- [4] MetaAI: *Demystifying a key self-supervised learning technique: Non-contrastive learning* (<https://ai.facebook.com/blog/demystifying-a-key-self-supervised-learning-technique-non-contrastive-learning/>)
- [5] S. Thalles: *Self-Supervised Learning and the Quest for Reducing Labeled Data in Deep Learning* (<https://sthalles.github.io/self-supervised-learning/>)
- [6] Udaljeni repozitorij simclr: <https://github.com/google-research/simclr>
- [7] Udaljeni repozitorij solo-learn: <https://github.com/vturrisi/solo-learn>
- [8] Udaljeni repozitorij BYOL-Pytorch: <https://github.com/reshinthadithyan/BYOL-Pytorch>