

Duboko učenje 1 međuispit

1. Razmatramo klasifikacijsku konvolucijsku mrežu s arhitekturom:
- konvolucijski sloj bez nadopunjavanja: dvije jezgre dimenzija $k \times k$, bez pomaka, korak 1;
 - aktivacija ReLU;
 - sažimanje maksimumom 2×2 s korakom 2;
 - pretvaranje u vektor;
 - potpuno povezani sloj dimenzije 2 te aktivacijom softmax.

Poznata je ulazna slika X i parametri W potpuno povezanog sloja (nema pomaka!):

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 3 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad W = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

2×8 8×8

Zadaci:

- Odredite nepoznati hiperparametar k u konvolucijskom sloju, takav da omogući validan unaprijedni prolaz kroz model ako je poznato da vrijedi $k > 1$.
- Odredite izlaze mreže ako su parametri konvolucijske jezgre zadani na sljedeći način. Težina prve jezgre u gornjem lijevom kutu jednaka je 1, u gornjem desnom -1, a 0 inače. Težine druge jezgre u gornjem lijevom i gornjem desnom kutu jednake su 1, a 0 inače.
- Odredite gradijente negativne log-izglednosti po izlazima konvolucijskog sloja, ako je točna oznaka za zadanu sliku $y = 0$.

Napomena: Kod sloja sažimanja maksimumom, u slučaju više jednakih kandidata izabrati onoga s manjim indeksom.

2. Potrebno je napraviti model za detekciju znaka plus u binarnoj ulaznoj slici veličine 4×4 . Plusom smatramo isječak izgleda $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$. Plus se može naći u bilo kojem dijelu slike. Ako se plus nalazi u slici, model na izlazu mora predvidjeti 1, inače mora predvidjeti 0. Zadaci:

- Konstruirajte potpuno povezani model koji može izvoditi zadani zadatak.
- Konstruirajte konvolucijski model koji može izvoditi zadani zadatak.
- Prikažite rad konstruiranih modela na slikama:
 - intenzitet 1 u gornjem lijevom pikselu, svi ostali intenziteti 0
 - plus u donjem desnom kutu, svi ostali intenziteti 0
- Izračunajte broj parametara za predložene modele i usporedite ih.
- Objasnite koja je od ovih arhitektura prikladnija za zadani problem.

3. Točnost konvolucijskog klasifikacijskog modela evaluiramo na podskupovima T_m , T_v , V_m i V_v . Podskupovi T_m i T_v sadrže slike za treniranje na kojima se nalaze po veličini isključivo mali (m) ili veliki (v) objekti. Analogno, podskupovi V_m i V_v sadrže slike iz skupa za validaciju podijeljene na isti način.

slučaj	točnost (%)			
	T_m	T_v	V_m	V_v
I.	98.4	98.5	52.3	53.0
II.	99.0	50.1	93.5	47.2
III.	50.4	50.2	49.3	49.0

Razmatramo rezultate evaluacije naučenih modela u tri odvojena slučaja kao što je prikazano u tablici. Nijedan od slučajeva ne pruža zadovoljavajuću točnost modela, stoga bismo u svakome od njih htjeli poduzeti neke promjene i naučiti nove poboljšane instance modela. Dostupne promjene su sljedeće:

- (a) povećanje broja kanala u konvolucijskim slojevima `[[1, ...]]`,
- (b) povećanje broja epoha treniranja modela `[[1, ...]]`,
- (c) povećanje prostorne dimenzije svih konvolucijskih jezgara
- (d) povećanje regularizacijskog hiperparametra λ koji kažnjava L2 normu parametara modela
- (e) dodavanje slojeva sažimanja `[[1, ...]]`.

Svaku od mogućih promjena uparite s točno jednim od slučajeva. Komentirajte svaki od slučajeva. Objasnite zašto ste određenu promjenu pridružili baš tome slučaju i zašto očekujete poboljšanje točnosti modela.

4. Razmatramo aktivacijsku funkciju softmax.

- (a) Pokažite da za vektor s i konstantu k vrijedi sljedeće: $\text{softmax}(s) = \text{softmax}(s + k)$.
- (b) Implementirajte funkciju `softmax_k(logits, k)` u numpy-u koja se oslanja na prethodno opisano svojstvo i logitima prije izračuna softmaxa dodaje argument k . Funkcija treba omogućiti rad s minigrupom podataka, odnosno pretpostavlja da je argument `logits` numpy polje dimenzija $N \times C$, gdje je N veličina minigrupe, a C broj razreda. Kakve oblike (engl. shape) argumenta k podržava vaša funkcija?
- (c) Po potrebi modificirajte implementaciju funkcije `softmax_k` tako da se ona može koristiti za ostvarivanje numerički stabilnog softmaxa; pokažite tu mogućnost na primjeru poziva funkcije pod pretpostavkom da je zadana matrica `logits`. Objasnite.

5. U modulu `layers.py` želimo dodati gubitak za učenje višezadačnog modela koji nam omogućuje da jednoj slici istovremeno pridijelimo više razreda. Primjerice, u sustavu za klasifikaciju slika koji prepoznaje mačke, pse i hrčke, ulazna slika koja sadrži i psa i mačku trebala bi imati oznaku `[1 1 0]`. Predložite implementaciju razreda `MultiLabelLossWithLogits` s metodama `forward` i `backward_inputs`. Metode na izlazu vraćaju iznos gubitka odnosno gradijente gubitka po ulazima, a kao argumente primaju logite i točne oznake. Uputa: tretirajte svaki izlaz kao neovisni binarni klasifikator. Bonus: interpretirajte vaš gubitak sa stanovišta teorije vjerojatnosti.