

Počítačové videnie - Úloha - CIFAR-10

Viktor Kocur

1.12.2018

1 Prostredie a dataset

Praktickú časť úlohy budete robiť v pythonovských knižniciach Keras a Tensorflow. Stiahnete ich jednoducho cez pip. Pozor, TF zatiaľ nespodporuje Python 3.7. Ak máte GPU nvidia, môžete si stiahnuť TF s podporou pre GPU. Inštrukcie nájdete na stránke tensorflow-u.

Celý projekt budete robiť na CIFAR-10 datasete. Ten nahráte nasledovne:

```
from keras.datasets import cifar10
(x, y), (x_test, y_test) = cifar10.load_data()
```

CIFAR-10 obsahuje 10 tried obrázkov. Obrázky sú farebné s rozmermi 32*32. Na cvičení sme pracovali s datasetom MNIST, na ňom sa dá veľmi jednoducho dosiahnuť viac ako 99 % presnosť. Na rozdiel od neho však jednoduché metódy dosahujú na CIFAR-10 presnosť len 40-60%. Preto sa nenechajte odradiť ak máte horšie výsledky, než na aké ste zvyknutí.

2 Cieľ úlohy

Cieľom úlohy je, aby ste sa oboznámili z procesom tréovania konvolučných neurónových sietí. Ako výstup z tejto úlohy bude pdf súbor kde popíšete ako ste postupovali a aké ste dosiahli výsledky. Navyše budete musieť odovzdať kódy k skriptom, ktoré ste na tréovanie použili.

Budete si môcť vybrať z podúloh, ktoré chcete riešiť. Každá podúloha súvisí s nejakým aspektom tréovania. V nasledujúcich podsekcích bude popísané čo sa očakáva ako výstup pre jednotlivé podúlohy a taktiež ich bodové ohodnotenie. Na konci je tabuľka, kde je všetko zosumarizované. Súčet bodov, ktoré sa dajú získať ja väčší ako 15, preto niektoré podúlohy môžete vynechať. Zópar z nich však bude povinných, ak tie nespravíte budú vám strhnuté body, aj keby ste mali za nepovinné úlohy body navyše. Ak vám bude vychádzať viac ako 15 bodov dostanete aj tak iba 15.

2.1 Základné tréovanie a inferencia - Povinné 2b

V pdf súbore popíšte jednoduchú konvolučnú sieť, rozdeľte si dáta na tréovacie a validačné a natrénujte ho. Model potom spustíte na testovacie dáta a v pdfku napíšete výslednú presnosť. Priložte k odovzdaniu skripty, ktoré ste použili na tréovanie.

2.2 Architektúry - 6 x 1b

Otestujte výsledky pre rôzne architektúry. Môžete meniť hĺbku siete, šírku, aktivácie, vrstvy. Tu používajte vrstvy ako Dense, Conv2D, Flatten a poolingové vrsvy. Vrstvy ako Dropout a BatchNorm patria do iných podúloh. Pre každú architektúru dajte do pdfka diagram generovaný kerasom (ak sa vám to nepodarí spravte to textovo) a porovnajte výsledky pre jednotlivé architektúry. Kludne môžete mať aj skoro rovnaké architektúry, len zmeníte aktiváciu, alebo pridáte počet filtrov atď.

2.3 Optimizátory a losses - 6 x 0.5b

Skúste použiť iné loss funkcie a optimizátory. Pre jednu architektúru vytvorte graf kde bude vidieť ako sa vyvíja validačná a tréningová chyba a presnosť medzi epochami pre jednotlivé optimizátory a loss funkcie.

2.4 Normalizácia 0.5b, Augmentácia 2 x 1b, BatchNorm 1b

Vyberte si jednu architektúru a vyskúšajte aké dosiahnete výsledky ak použijete na vstupné dáta normalizáciu, rôzne augmentácie (v texte vysvetlite aké a prečo dávajú zmysel), alebo v sieti použijete BatchNorm vrstvu. V texte porovnajte výsledky. Nezabudnite ich porovnať aj s modelom bez úpravy vstupných dát. Na normalizáciu a augmentáciu použite <https://keras.io/preprocessing/image/>

2.5 Regularizácia 3 x 0.5b

Otestujte regularizáciu pomocou Dropout vrstvy, alebo L1/L2 regularizácie <https://keras.io/regularizers/>. V texte popíšte aké parametre ste regularizovali, prípadne kam ste dali Dropout (môžete tiež pridať diagram) a porovnajte výsledky voči modelu bez regularizácie.

2.6 Regularizácia - maličký dataset 2 b

Vytvorte si tréningový dataset ktorý má strašne málo príkladov napr. 3-6 obrázka na triedu. Skúste na ňom natrénovať sieť s L2 regularizáciou a bez nej. Porovnajte rozdiel.

2.7 Najlepší tréningový postup a inferencia - Povinné 2b (+2b pre najlepší model)

Kompletne popíšte, čo ste počas tréningu vášho najlepšieho modelu použili. Pridajte aj diagram. Model pošlite (mailom, alebo cez cloud) spolu s ostatnými súborami. V prípade, že ste použili normalizovali vstup, tak pošlite aj samostatný súbor na testovanie tohoto modelu (bude tam `model.eval()`).

3 Štruktúra

Dajte pdfku nejaku rozumnú štruktúru. Napríklad každá sekcia je jedna podúloha. Do nadpisu sekcie dajte aj to koľko bodov za ňu očakávate. Architektúram/modelom dajte nejaké rozumné meno, aby sa dali identifikovať ďalej v texte a kóde.

3.1 Skripty

Kód ktorý ste pri práci použili odošlite spolu s pdfkom. Preferoval by som napríklad, ak by ste mali aj viacero tréningových cyklov v jednom skripte. Ideálne celá úloha v jednom skripte, ktorý nakreslí všetky grafy, diagramy a vypíše výsledky. V skriptoch vždy označte komentárom model podľa názvu z pdfka a takisto označte v pdfku ako sa volá skript kde sa realizuje to čo popisujete.

3.2 Model

Súčasťou úlohy je uploadnúť aspoň jeden model. Ak bude príliš veľký uploadnite ho niekam napr. google drive a napíšte do pdfka link, ideálne aj do mailu.

3.3 Vizualizácia

Väčšinu vizualizácii môžete robiť rovno pomocou kerasu. Pozrite si <https://keras.io/visualization/> Tam nájdete ako kresliť grafy pre loss, accuracy a ako robiť diagramy k sietiam. V `keras.utils.plot_model` použijete `show_shapes=True` (pozn. ak toto nespravíte, tak ako keby ste ten diagram v texte ani nemali)

4 Odovzdávanie

V prílohe mi na kocurvik@gmail.com pošlite pdfko, link na model a zip so skriptami. Deadline je 16.12. 18:00. Za úlohu môžete dostať max 15 bodov aj keď budete mať hotových viac vecí. Ako bolo už spomenuté, nemusíte urobiť všetko. Vyberte si čo vás zaujalo. Z celkového hodnotenia sa odčítavajú iba body ktoré chýbajú v povinných podúlohách. Za každý deň meškania sa strhnú 3 body (tj. zníži sa maximálny počet bodov).

5 Tabuľka

Podúloha	Povinná	Body	Výstup
Základný tréning	áno	2	test acc
Architektúry	nie	6 x 1	diagram + porovnanie test acc
Optimizátory	nie	6 x 0.5	graf vývoja tréningovej/val loss a acc
Normalizácia	nie	0.5 + 1	porovnanie test acc medzi sebou a oproti baseline
Augmentácia	nie	2 x 1	
BatchNorm	nie	1	
Regularizácia	nie	3 x 0.5	porovnanie test acc medzi sebou a oproti baseline
Regularizácia + malý dataset	nie	2	porovnanie test acc
Najlepší model	áno	2 (+2)	podrobný popis + model + test acc + testovací skript
Spolu		21 (+2)	ku všetkému treba popis + skripty