Využitie samoorganizácie v čiastočne riadenom učení hlbokých neurónových sietí

Sabína Samporová a Kristína Malinovská
Katedra aplikovanej informatiky, FMFI,
Univerzita Komenského v Bratislave
Mlynská dolina, 84248 Bratislava
Email: {samporova1, rebrova1}@uniba.sk

Abstrakt

Kvalitný a rozsiahly súbor dát je pri trénovaní hlbokých neurónových sietí veľmi dôležitý. Vytvorenie takéhoto súboru dát s označeniami je ale veľmi náročné, preto sa hľadajú modely na využitie neoznačených dát, popri trénovaní na označených dátach. Čiastočne riadené učenie sa zaoberá skúmaním týchto modelov. Zamerali sme sa na model Mean Teacher a jemu podobné, založené na dvoch hlbokých sieťach, ktoré dostanú rovnaký vstup s rôznymi augmentáciami. Cieľom Mean Teacher modelov je trénovanie na označených dátach a zároveň snaha o konzistenciu označovania medzi odpoveďami na označené aj neoznačené dáta. Spravidla sa na určenie chyby konzistencie používa štvorec chýb označení. Náš prístup je využiť samoorganizáciu neurálnych reprezentácií na poslednej konvolučnej vrstve, čím by sa mohla zlepšiť úspešnosť modelu, ako aj vhľad do modelu v zmysle vysvetliteľnej umelej inteligencie.

1 Úvod

Hlboké neurónové siete sú v súčasnosti pravdepodobne najpoužívanejšími a najskúmanejšími modelmi v strojovom učení s aplikáciami v mnohých rôznych oblastiach. Trénovanie takýchto modelov si vyžaduje veľké množstvo adekvátne označených trénovacích dát, no zvyčajne je dobre označených dát z reálneho sveta málo. Paradigma čiastočne riadeného učenia (semi-supervised learning) rieši tento problém prostredníctvom rôznych techník. Mnohé z nich používajú pri učení odchýlku medzi výstupmi siete pre dve rôzne augmentácie toho istého vstupu, čo sa nazýva regularizácia na základe konzistencie (consistency regularization). Jedným zo známych predstaviteľov tohto druhu čiastočne riadeného učenia je model Mean Teacher model (MT), ktorý navrhli Tarvainen a Valpola (2017). V tomto príspevku predstavujeme návrh na adaptáciu tohto modelu s použitím samoorganizácie.

2 Mean teacher model

Mean Teacher model (Tarvainen a Valpola, 2017) pozostáva z dvoch hlbokých sietí s rovnakou architektúrou, ale samostatnými trénovateľnými parametrami, teda váhami. Prvá z nich je nazývaná študent a označujeme ju ako θ a druhá učiteľ θ'. MT model je vhodný na riešenie problému klasifikácie. Na trénovanie model využíva nie len označené, ale aj neoznačené dáta, teda také, ktoré nemajú priradenú príslušnosť do niektorej z tried.

Dôležitým komponentom modelu je použitie takzvaných augmentácii vstupných obrázkov ako je napríklad náhodná translácia či rotácia a rôzne druhy slabého šumu (Gaussovský, zaušumenie farieb, atď.). Technika augmentácie dát je bežne používaná a spravidla vylepšuje aj klasické učenie s učiteľom (Krizhevsky a spol., 2012). Pri čiastočne riadenom učení zohráva dôležitú úlohu, ale na samotnú kompenzáciu absencie označení nestačí.

MT na trénovanie študentského modelu používa 2 druhy stratových funkcií (loss functions), Supervised loss $S(\theta)$ definovanú v (1) a takzvanú Consistency loss $J(\theta)$ definovanú v (2). $S(\theta)$ môžeme určiť len pre dáta, ktoré majú označenie a je to krížová entropia (cross entropy) predikcie siete pre vstup x_j, na ktorý aplikujeme augmentáciu η a príslušného označenia y_j.

$$S(\theta) = \frac{1}{m} \sum_{j}^{m} \left[-\log \; P_f(y_j | x_j; \theta, \eta) \right], \qquad (1)$$

Ako $J(\theta)$, čiže chyba konzistencie, je použitá stredná kvadratická chyba (MSE) predikcie študentského a učiteľského modelu, ktoré sa trénujú súčasne pre rovnaký vstup, na ktorý aplikujeme rôzne augmentácie η a η'. Tento komponent chybovej funkcie nevyžaduje pre svoj výpočet žiadne označenia y čiže hovoríme o učení bez učiteľa a môžeme ho aplikovať aj na dáta bez označení.

$$J(\theta) = \frac{1}{n} \sum_{i}^{n} \left\| f(x_i, \theta', \eta') - f(x_i, \theta, \eta) \right\|^2, \qquad (2)$$

Celková hodnota stratovej funkcie, ktorá sa použije pri učení je potom vyjadrená ako vážený súčet $S(\theta)$ a $J(\theta)$

$$Loss(\theta) = S(\theta) + w_t J(\theta), \tag{3}$$

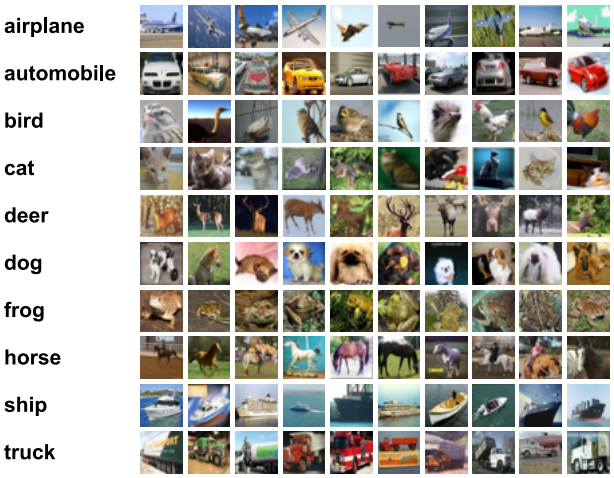
kde w_t je váha chyby konzistencie, ktorá je na začiatku pomerne malá ale v priebehu trénovania rastie. Na základe stratovej funkcie je študentský model θ je trénovaný spätným šírením chyby metódou najmenšieho gradientu.

Učiteľský model nemá rovnaký spôsob trénovania, ale funguje ako exponenciálny kĺzavý priemer (exponential moving average, EMA) študentského modelu. Táto stratégia sa nazýva Temporal Ensembling (Laine a Aila, 2016) čiže skladanie modelu v čase a predstavuje akýsi ďalší druh regularizácie. Úprava váh modelu učiteľa je vykonaná pomocou pravidla (4), kde θ'_t je označenie pre váhy učiteľského modelu v čase t, a θ_t sú váhy študenta v čase t a α je hyperparameter - rýchlosť učenia.

$$\theta'_t = \alpha\theta'_{t-1} + (1-\alpha)\theta_t \tag{4}$$

3 Dataset a baseline

Ako dataset sme zvolili štandardný CIFAR10 dataset (Krizhevsky a spol., 2009), ktorý pozostáva zo 60 tisíc farebných obrázkov s rozmermi 32×32 pixelov. Obrázky sú označené a sú z 10 rôznych tried (lietadlá, autá, vtáky, mačky, jelene, psy, žaby, kone, lode, nákladné autá) ilustrovaných na Obr. 1.



Obr. 1: CIFAR10: desať náhodne zvolených obrázkov[1].

Pri vývoji modelu sme vychádzali priamo zo zverejneného modelu a kódu od autorov článku, Tarvainena a Vapolu[2], kde sú zverejnené aj optimálne parametre pre klasifikáciu datasetu CIFAR10, ktorý dosahuje presnosť $93.72\% \pm 0.15$. Tento výsledok budeme brať ako základ pre porovnanie úspešnosti nášho modelu a so zapojením princípu samoorganizácie očakávame zlepšenie úspešnosti v zmysle presnosti klasifikácie.

4 Využitie samoorganizácie: náš model

Pri tvorbe nášho modelu sme uvažovali, ako formulovať chybu konzistencie, ktorá by zahŕňala viac informácie ako v doterajšom modeli. Keďže chyba konzistencie v MT je vyjadrená vzdialenosť vektorov, s rozmerom rovným počtu tried, ide o pomerne málo informácie. Tuna a spol.(2021) vo svojom modeli Binary Mean Teacher modelovali binárnu klasifikáciu prítomnosti objektu na obrázku. V svojom modeli teda nemohli vyjadriť chybu konzistencie pri odozve z intervalu $(0, 1)$ pre dve triedy (oproti 10 triedam v CIFAR10) a teda vyjadrili $J(\theta)$ ako MSE na poslednej konvolučnej vrstve použitej architektúry, čiže ešte pred plne prepojenou časťou siete.[3]

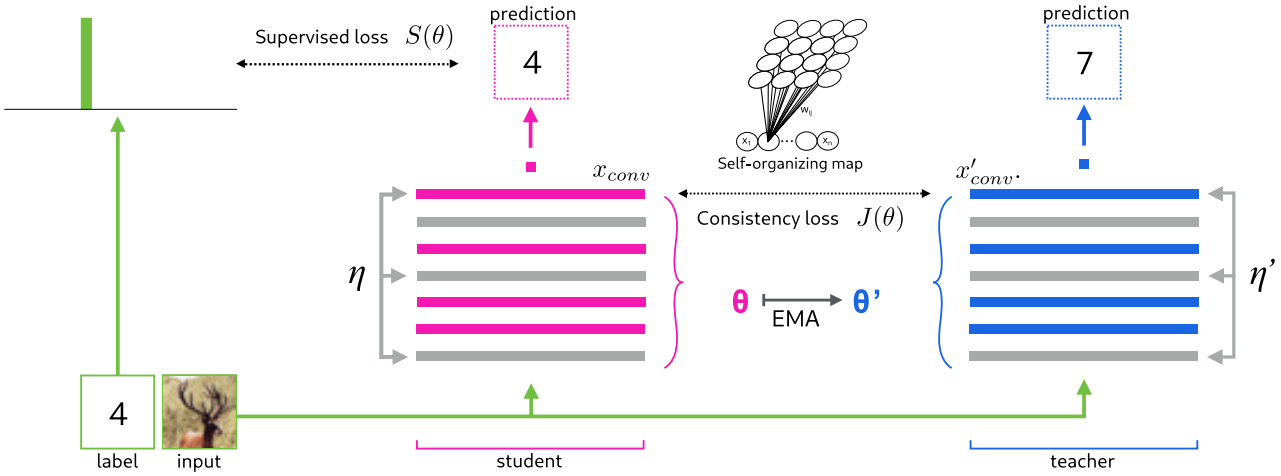
Túto reprezentáciu využijeme aj my a nazývame x_{conv}. Predpokladáme, že reprezentácia vstupu po prechode všetkými konvolučnými vrstvami, bude vhodnejšia pre zachytenie rozdielov a podobností medzi obsahom obrazu, než jeho samotná klasifikácia. Náš navrhovaný model ilustrujeme na Obr. 2. Ďalej navrhujeme, že pridanie konceptu samoorganizácie môže zlepšiť trénovanie tým, že bude lepšie odzrkadľovať vzdialenosť dát, než pôvodný spôsob počítania $J(\theta)$. Samoorganizáciu zapojíme do počítania $J(\theta)$ tak, že využijeme samoorganizujúcu sa mapu (SOM) (Kohonen, 1990), ktorú natrénujeme na rozpoznávanie reprezentácií x_{conv} zo študentského aj učiteľského modelu počas danej epochy, keď sa model učí. Potom túto SOM použijeme na počítanie Consistency loss $J(\theta)$.

Samoorganizujúce sa mapy majú vstupnú vrstvu veľkosti vstupného vektora plne prepojenú s nasledujúcou vrstvou, ktorú tvorí mapa neurónov konkrétneho tvaru, napríklad mriežka. V každom neuróne máme parameter siete, vektor veľkosti vstupu, ktorý reprezentuje niečo ako prototyp. Vyjadrením euklidovskej vzdialenosti medzi vstupným neurónov a váhami neurónov vyjadríme víťaza v zmysle najbližšieho neurónu, ktorého váhu potom pri učení posilníme daným vstupom a zároveň posilníme aj jeho okolie. Výsledkom učenia sa SOM je topografická mapa vstupného priestoru, ktorá zachytáva štruktúru a podobnosti v dátach nelineárnym spôsobom.

Označme reprezentácie z poslednej konvolučnej vrstvy študentského a učiteľského modelu ako x_{conv} a x'_{conv}. V SOM nájdeme víťazný neurón (winner neuron, best matching unit) pre x_{conv} a x'_{conv}, označme c a $c\prime$. Keďže neuróny SOM sú definované ich váhami a

[1] https://www.cs.toronto.edu/~kriz/cifar.html
[2] github.com/CuriousAI/mean-teacher/

[3] Podobná stratégia sa bežne používa pri transfer learning (Weiss a spol., 2016), kde sa použije model, ktorý už je dobre natrénovaný na všeobecnej úlohe a jeho plne prepojená vrstva alebo vrstvy sa nahradia novou architektúrou, ktorá modeluje inú úlohu.



Obr. 2: MT model

tie vlastne reprezentujú polohu tohto neurónu v mnohorozmernom priestore. Potom našim cieľom je, aby c a $c\prime$, predstavujúce vstupné reprezentácie, ktoré vznikli z rovnakého obrázku, boli v tomto priestore čo najbližšie. Preto našu novú chybu konzistencie definujeme ako súčet vzdialenosti víťazov a tiež vzdialenosti víťaza od konvolučnej reprezentácie vstupu. Vzdialenosť dvoch bodov a, b v tomto priestore označíme $d(a, b)$ a teda chybovú funkciu $J(\theta)$ vieme zapísať ako:

$$J(\theta) = d(c, c\prime) + d(x_{conv}, c) + d(x'_{conv}, c\prime) \qquad (5)$$

Našu intuíciu sme overili jednoduchým experimentom. Z nášho baseline MT modelu natrénovaného na CIFAR10 sme vybrali x_{conv} reprezentácie vstupných obrázkov z validačnej sady a použili ich ako trénovacie dáta pre SOM. Na Obr. 3 zobrazujeme rozloženie tried na natrénovanej mape, kde vidno klastre pre jednotlivé kategórie aj bez použitia SOM, ako chybovej funkcie pri učení. Očakávame, že ak zapojíme nami navrhovanú chybu konzistencie bude táto organizácia lepšia a zároveň očakávame aj zlepšenie samotnej úspešnosti modelu v klasifikačnej úlohe.

Potenciál nášho modelu môže byť aj lepší vhľad do vnútorných reprezentácií neurónových sietí, s možným využitím v doméne vysvetliteľnej umelej inteligencie, keďže nám umožní porovnať odpovede modelu pre neoznačené dáta s prototypmi naučených tried a umožní nám teda dáta klasifikovať a skúmať presnosť modelu v zmysle klasifikácie nových vstupov. Akýsi vhľad môžeme pozorovať už v našom experimente na Obr. 3, kde vidíme prekryv medzi dvoma triedami, ktoré ale znamenajú jelene a kone, čiže to, že sú na mape zaznačené na rovnakom mieste môžeme chápať ako prirodzenú vlastnosť podobnosti týchto tried.

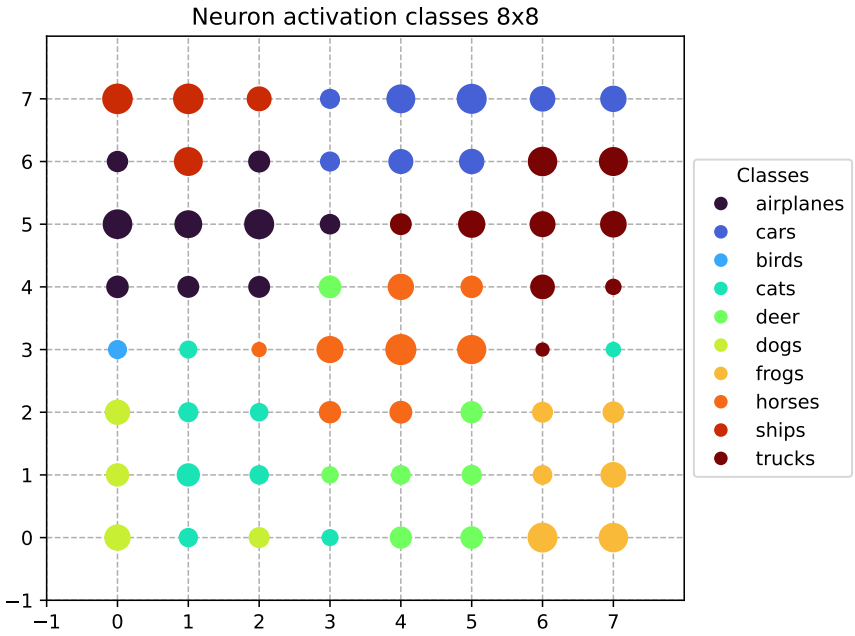
5 Implementácia a predbežné výsledky experimentu

V našej implementácii sme vychádzali z pôvodného kódu autorov modelu MT a prebrali architektúru, ktorú natrénovali na najlepšiu presnosť a zapojili sme do nej samoorganizujúcu sa mapu. Trénovanie prebiehalo tak, že sa v rámci jednej epochy pri doprednom prechode zapamätali reprezentácie x_{conv} z oboch modelov, pomocou spätného šírenia chyby sa upravil študentský model a pomocou EMA sa upravil učiteľský model a následne po úprave váh modelov sa zapamätané x_{conv} použili na natrénovanie SOM. SOM loss sme začali používať na úpravu váh až od druhej epochy, keďže v prvej epoche nebola SOM ešte natrénovaná.

Keďže architektúra Tarvainen a Valpola (2017) bola veľmi veľká, približne 30 miliónov parametrov, nemali sme dostatočné zdroje na trénovanie takéhoto modelu. Rozhodli sme sa teda pre účely experimentovania ďalej pracovať s o niečo menšou architektúrou modelu MT. Vybrali sme si architektúru, ktorú používa Muhammad Sarmad[4]. Tá obsahuje o niečo viac než 3 milióny parametrov, ale je možné ju s použitím našich zdrojov natrénovať.

Aktuálne ešte stále prebieha optimalizácia hyperparametrov, s dôrazom na ladenie nami navrhnutej stratovej funkcie. V budúcnosti plánujeme experimentovať aj s hyperparametrami samoorganizujúcej mapy ako aj jej topológiou. Pri učení SOM plánujeme zapojiť mechanizmus na kompenzáciu toho, že feature vektory, ktoré dostáva SOM na vstupe sa neustále vyvíjajú. Napríklad, že pri výbere víťaza uprednostníme vzdialenejší neurón a/alebo taký, ktorý je málo saturovaný (teda málo krát zvíťazil), s cieľom zapojiť čo najviac neurónov SOM a získať čo najviac vyrovnanú organizáciu na mape.

[4] github.com/iSarmad/MeanTeacher-SNTG-HybridNet



Obr. 3: Distribúcia tried na mape: pre každý neurón siete zobrazujeme, koľko krát bol neurón víťazom pre najpočetnejšie zastúpenú triedu označenú farebne, táto početnosť je zobrazená veľkosťou značky.

6 Záver

Paradigma čiastočne riadeného učenia je zaujímavý nástroj na vysporiadanie sa s nedostatkom adekvátne označených dát a využitím neoznačených dát na zlepšovanie úspešnosti hlbokého modelu. Ukázalo sa, že trieda modelov MT je účinná na takéto situácie. Keďže pôvodná chyba konzistencie, ktorú model využíva, je málo informovaná, navrhli sme ju nahradiť samoorganizujúcou mapou, pretože očakávame, že by mohla vernejšie zachytiť štruktúru a podobnosť dát a tak lepšie určiť ich konzistenciu, ako aj zlepšiť výkon modelu MT a jemu podobných. Takto navrhnutá chybová funkcia by mohla mať využitie aj v iných čiastočne riadených modeloch a obohatiť tieto modely o známy a biologicky relevantný spôsob učenia bez učiteľa.

Poďakovanie

Tento príspevok vznikol v Centre pre kognitívnu vedu na KAI FMFI UK v Bratislave, s podporou grantu VEGA 1/0373/23 a KEGA 022UK-4/2023. Za podporu tiež ďakujeme Slovenskej spoločnosti pre kognitívnu vedu SSKV[5].

[5] https://cogsci.fmph.uniba.sk/sskv/

Literatúra

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480.

Krizhevsky, A., Hinton, G. a spol. (2009). Learning multiple layers of features from tiny images. Technická správa, University of Toronto, Toronto.

Krizhevsky, A., Sutskever, I. a Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. V *Advances in Neural Information Processing Systems*, vol. 25, str. 1097–1105.

Laine, S. a Aila, T. (2016). Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*.

Tarvainen, A. a Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S. a Garnett, R. (zost.), V *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc.

Tuna, M., Malinovská, K., Farkas, I., Kraus, S. a Krsek, P. (2021). Semi-supervised learning in camera surveillance image classification. V *2021 IEEE 17th International Conference on Intelligent Computer Communication and Processing (ICCP)*, str. 155–162.

Weiss, K., Khoshgoftaar, T. M. a Wang, D. (2016). A survey of transfer learning. *Journal of Big data*, 3(1):9.