TypeCNN

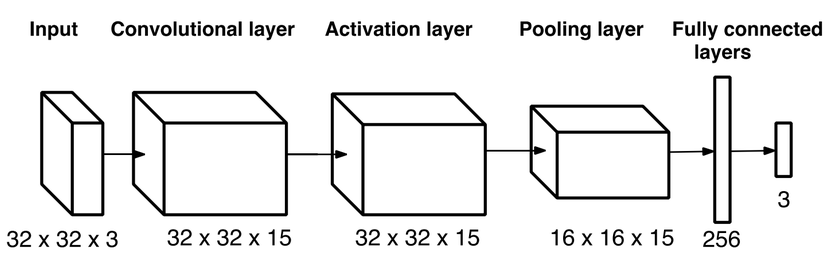
Knihovna pro práci s konvolučními neuronovými sítěmi podporující různé datové typy

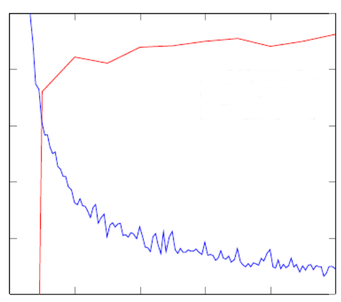
Petr Rek, xrekpe00@stud.fit.vutbr.cz / rekp@email.cz

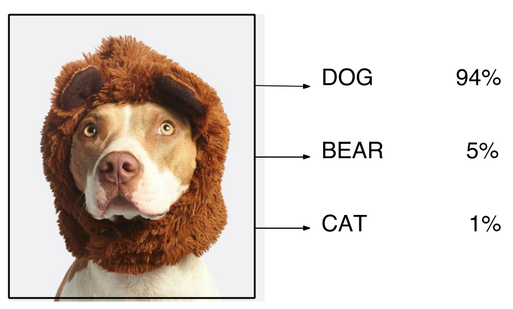
26

Excel@FIT 2018

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY







Knihovna TypeCNN

Dostupná z **github.com/Rekpet/TypeCNN**

- Konvoluční neuronová síť:
  - Výpočetní vrstvy – Konvoluční, Poolingová (average i max), Plně propojená, Drop-out, Aktivační,
  - Aktivační funkce - ReLU, Leaky ReLU, Sigmoid, Tanh, Softmax,
  - Ztrátové funkce – Mean squared error, Cross-entropy,
  - Optimalizátory – SGD, Momentum, Nestorov momentum, Adagrad, Adam.
- Podpůrné prostředky:
  - XML schéma pro specifikaci CNN,
  - Modul perzistance (načítání a ukládání),
  - Konzolové i programové rozhraní pro CNN i NN,
  - Parsery pro formáty IDX, Binary a PNG,
  - Podpora až tří datových typů v rámci jedné vrstvy, nezávislé mezi vrstvami,
  - Datový typ FixedPoint – reprezentace v pevné řádové čárce s definovatelnou šířkou celé a desetinné části.

Dosažené výsledky

Datové sady

Knihovna byla testována na datových sadách MNIST (99.31%) a CIFAR-10 (62.20%), což vzhledem k použitým architekturám demonstruje schopnost knihovny provádět trénování CNN.

Porovnání s jinými knihovnami

Knihovna byla porovnána s jinými knihovnami. Z pohledu kvality trénování je srovnatelná s těmi používanými v praxi. Z pohledu času se nachází ve středu mezi menšími (projekt jednoho člověka) a většími (používané v praxi) projekty.

Aproximace založená na pevné řádové čárce

Bylo ukázáno, že NN a CNN lze provádět v reprezentaci s pevnou řádovou čárkou. Pro 16+ bitů s nízkým či žádným vlivem na přesnost. Pro méně bitů již záleží na konkrétní úloze a architektuře sítě.

Cíl

Vyvinout knihovnu v jazyce C++, které bude rozumně rychlá a obsahovat rozumné množství funkcionality a pomocných prostředků.

Knihovna bude navíc obsahovat tři na sobě nezávislé typové aliasy – pro váhy, pro inferenci a pro trénování.

Součástí bude vyhodnocení knihovny na datových sadách typických pro konvoluční neuronové sítě, porovnání knihovny s jinými volně dostupnými a také případová studie využívající typové nezávislosti – bude ověřen vliv použití reprezentace s pevnou řádovou čárkou na přesnost konvoluční neuronové sítě. Tato reprezentace je vhodná například pro mobilní a vestavěná zařízení, kdy je podstatná rychlost anebo spotřeba.

Typová nezávislost a použitá aproximace

Knihovna obsahuje celkem tři na sobě nezávislé typové aliasy:

- **WeightType** – datový typ vah při jejich použití během inference a ukládání na disk,
- **ForwardType** – datový typ pro provádění inference, včetně veškerých mezivýpočtů,
- **BackwardType** – datový typ pro trénování (gradienty, přesnější uložení vah, …).

Za aliasy lze dosadit libovolný datový typ podporující danou množinu operací (aritmetické, logické, …), včetně uživatelem definovaných.

V rámci experimentů je použita vlastní implementace datového typu s pevnou řádovou čárkou FixedPoint<F, P>, kde F je počet bitů před řádovou čárkou a P počet bitů za řádovou čárkou. Tato reprezentace je vhodná pro použití v zařízeních, které kladou důraz na spotřebu či rychlost. Přináší však problémy související s přetékáním na menších bitových šířkách a nízkou rozlišitelností.

Typickým postupem je natrénování sítě na datovém typu s plovoucí řádovou čárkou a poté převedení do této reprezentace a dotrénování.

Příklad reprezentace

4 bity celé části, 4 bity desetinné části

Nejmenší hodnota = -8 Největší hodnota = 7.9375

Experimenty s MNIST

Název knihovny	Doba trénování	Úspěšnost
SimpleCNN	749 s	97.18%
TinyDNN	152 s	98.15%
Keras (TensorFlow)	743 s	98.18%
TypeCNN	530 s	98.22%

Tabulka 1: Porovnání knihoven při trénování CNN po dobu 10 epoch na datové sadě MNIST

Datový typ	Úspěšnost	
	Před dotrénováním	Po dotrénování
float	97.61%	97.84%
FixedPoint<14,14>	97.61%	97.85%
FixedPoint<8,8>	97.70%	97.88%
FixedPoint<12,4>	74.80%	97.71%
FixedPoint<6,2>	20.84%	95.09%
FixedPoint<4,4>	10.28%	94.66%
FixedPoint<3,5>	11.65%	84.95%

Tabulka 2: Neuronová síť trénována 10 epoch na datovém typu float, poté dotrénována s daným typem pro inferenci a váhy

Datový typ	Úspěšnost
float	97.61%
FixedPoint<14,14>	97.53%
FixedPoint<8,8>	97.76%
FixedPoint<12,4>	96.79%
FixedPoint<5,3>	92.45%
FixedPoint<4,4>	77.30%

Tabulka 3: Trénování neuronové sítě od počátku na daném datovém typu

Datový typ	Úspěšnost	
	Před dotrénováním	Po dotrénování
float	98.23%	98.44%
FixedPoint<14,14>	94.84%	98.45%
FixedPoint<8,8>	94.03%	98.50%
FixedPoint<12,4>	14.56%	97.80%
FixedPoint<6,2>	10.12%	26.36%

Tabulka 4: Konvoluční neuronová síť trénována po dobu 10 epoch na datovém typu float, poté dotrénována s daným typem pro inferenci a váhy

Datový typ	Úspěšnost	
	Před dotrénováním	Po dotrénování
FixedPoint<8,8>	94.03%	98.50%
FixedPoint<2,6>	94.28%	98.47%
FixedPoint<3,5>	90.77%	98.40%
FixedPoint<4,4>	79.24%	97.26%
FixedPoint<2,2>	9.80%	88.39%

Tabulka 5: Konvoluční neuronová síť trénována po dobu 10 epoch na datovém typu float, poté dotrénována s datovým typem FixedPoint<8,8> pro inferenci a daným typem pro váhy