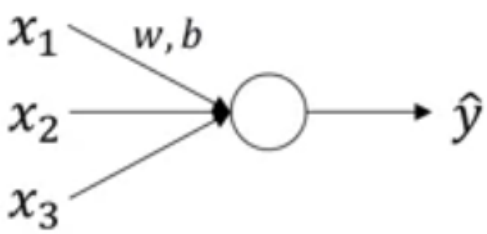
Нормализация в глубинном обучении

Чёлушкин Максим 172

Нормализация

- Уменьшает время обучения сети
- Уменьшает ковариантный сдвиг
- Помогает поддерживать значимость признаков с разными значениями

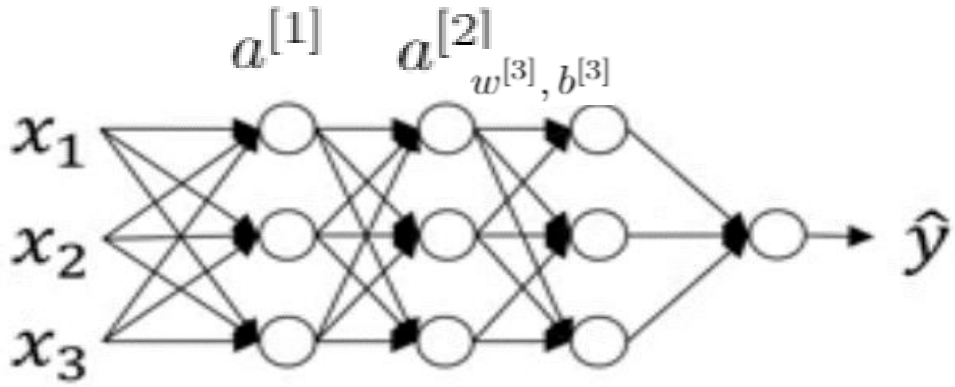
Batch нормализация



$$\mu = \frac{1}{m}\sum_i (x_i)$$

$$\sigma^2 = \frac{1}{m}\sum_i (x_i - \mu)^2$$

$$x = \frac{x-\mu}{\sigma}$$



$$z^{[l]} = z^{(1)} \ldots z^{(m)}$$

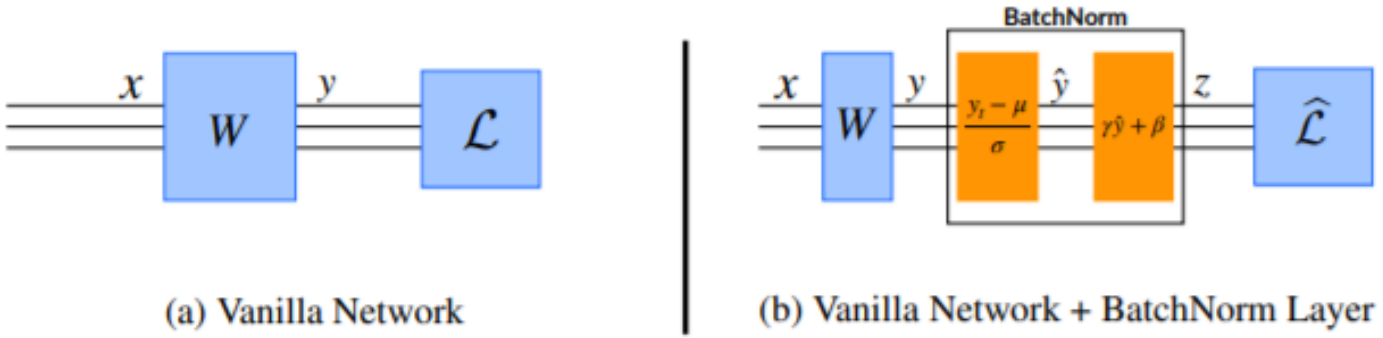
$$\mu = \frac{1}{m}\sum_i z^{(i)}$$

$$\sigma^2 = \frac{1}{m}\sum_i (z^{(i)} - \mu)^2$$

$$z^{(i)}_{norm} = \frac{z^{(i)} - \mu}{\sqrt{\sigma^2 - \epsilon}}$$

$$\tilde{z}^{(i)} = \gamma z^{(i)}_{norm} + \beta$$

Batch нормализация



(a) Vanilla Network

(b) Vanilla Network + BatchNorm Layer

$$X \xrightarrow{w^{[1]}b^{[1]}} z^{[1]} \xrightarrow[BN]{\beta^{[1]}\gamma^{[1]}} \tilde{z}^{[i]} \rightarrow a^{[1]} \xrightarrow{w^{[2]}b^{[2]}} z^{[2]}$$

$$X^{\{1\}} \xrightarrow{w^{[1]}b^{[1]}} z^{[1]} \xrightarrow[BN]{\beta^{[1]}\gamma^{[1]}} \tilde{z}^{[i]} \rightarrow a^{[1]} \xrightarrow{w^{[2]}b^{[2]}} z^{[2]}$$

$$X^{\{2\}} \xrightarrow{w^{[1]}b^{[1]}} z^{[1]} \xrightarrow[BN]{\beta^{[1]}\gamma^{[1]}} \tilde{z}^{[i]} \rightarrow a^{[1]} \xrightarrow{w^{[2]}b^{[2]}} z^{[2]}$$

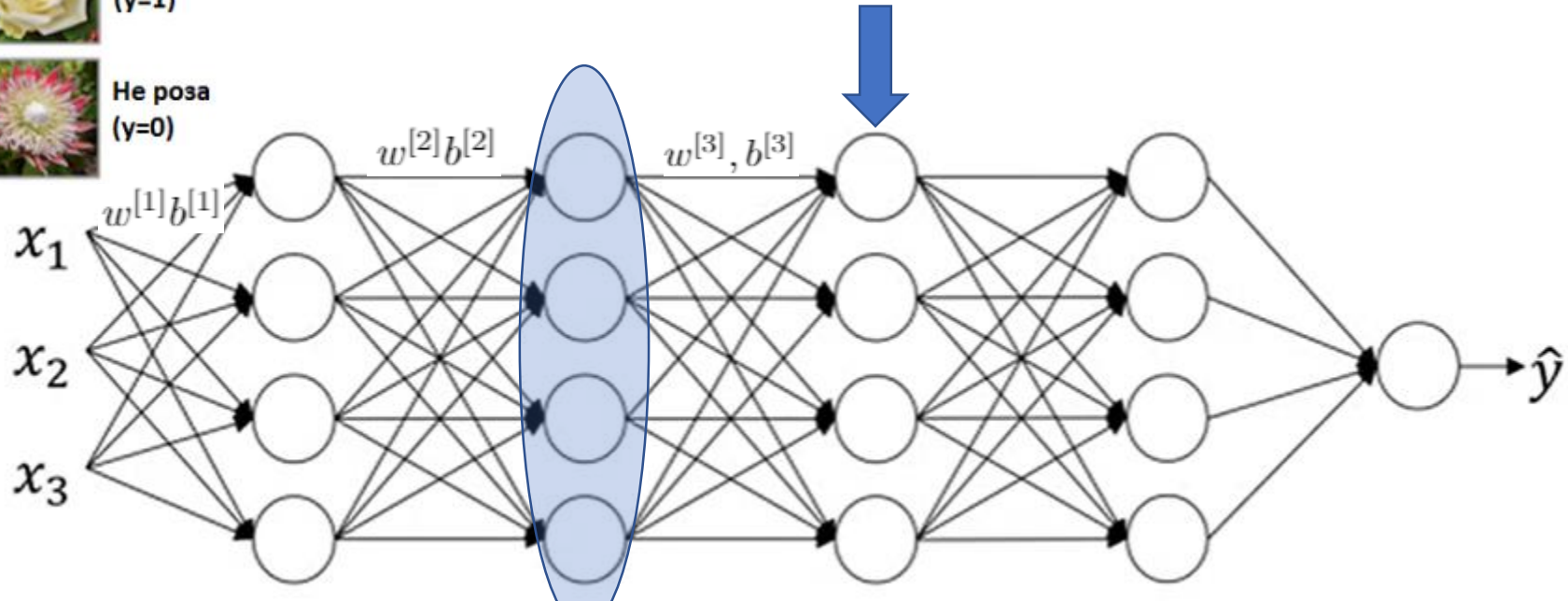
Ковариантный сдвиг

Роза
(y=1)

Не роза
(y=0)

Роза
(y=1)

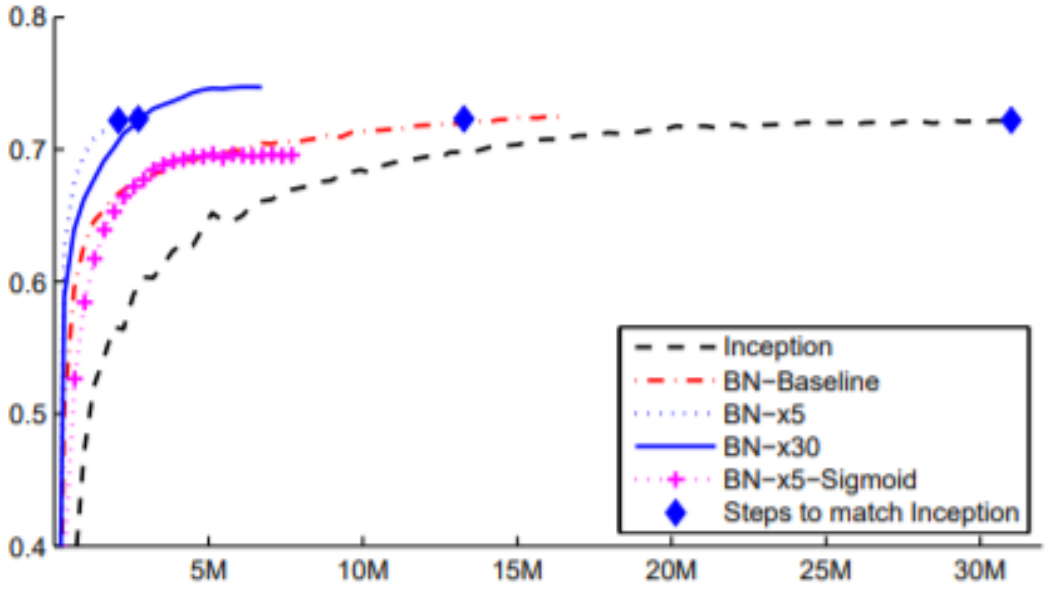
Не роза
(y=0)



Batch нормализация как регуляризатор

- Среднее и дисперсия считаются по mini-batch
- Это добавляет шум к значениям z
- Схоже с работой регуляризации dropout

Batch нормализация



Model	Steps to 72.2%	Max accuracy
Inception	$31.0 \cdot 10^6$	72.2%
BN-Baseline	$13.3 \cdot 10^6$	72.7%
BN-x5	$2.1 \cdot 10^6$	73.0%
BN-x30	$2.7 \cdot 10^6$	74.8%
BN-x5-Sigmoid		69.8%

Нормализация весов

- **Инициализация** параметров
- **Репараметризация** нейронной сети

Инициализация

- Начинаем с случайными w, b
- Для каждого узла считаем значения до активации
- Считаем среднее и дисперсию
- Пересчитываем параметры
- Пересчитаем выход каждого нейрона

$$t = w \cdot x + b$$

$$\mu[t], \sigma^2[t]$$

$$w = \frac{1}{\sigma[t]} w, b = \frac{b - \mu[t]}{\sigma[t]}$$

$$y = g(\frac{t - \mu[t]}{\sigma[t]}) = g(w \cdot x + b)$$

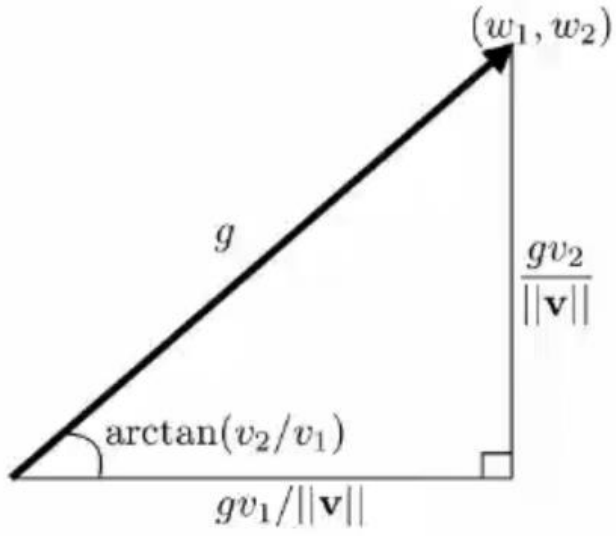
Репараметризация

- Представим веса в виде функции от новых параметров

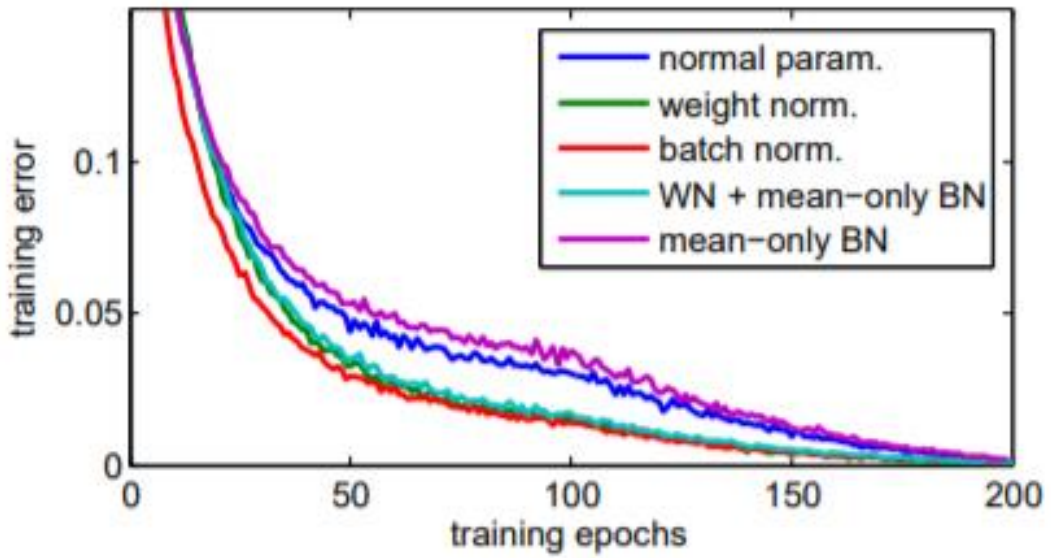
$$\mathbf{w} = \frac{g}{||\mathbf{v}||}\mathbf{v}$$

- Обучаем сеть с новыми параметрами

- Разделяет длину и направление вектора весов



Нормализация весов



Model	Test Error
Maxout [6]	11.68%
Network in Network [17]	10.41%
Deeply Supervised [16]	9.6%
ConvPool-CNN-C [26]	9.31%
ALL-CNN-C [26]	9.08%
our CNN, mean-only B.N.	8.52%
our CNN, weight norm.	8.46%
our CNN, normal param.	8.43%
our CNN, batch norm.	8.05%
ours, W.N. + mean-only B.N.	**7.31%**

Нормализация слоев(Layer Norm)

- Пусть есть mini-batch, где каждое наблюдение содержит K эл-ов
- Считаем среднее и дисперсию по каждому наблюдению
- После нормализуем наблюдение
- Делаем сдвиг

$$B = \{x_1, x_2, \ldots, x_m\},$$

$$\{x_{i,1}, x_{i,2}, \ldots, x_{i,K}\}$$

$$\mu_i = \frac{1}{K}\sum_{k=1}^{K} x_{i,k}$$

$$\sigma_i^2 = \frac{1}{K}\sum_{k=1}^{K}(x_{i,k} - \mu_i)^2$$

$$\hat{x}_{i,k} = \frac{x_{i,k} - \mu_k}{\sqrt{\sigma_k^2 + \epsilon}}$$

$$y_i = \gamma\hat{x}_i + \beta \equiv \mathrm{LN}_{\gamma,\beta}(x_i)$$

Нормализация слоев для RNN

- Нормализация никак не затрагивает другие наблюдения.

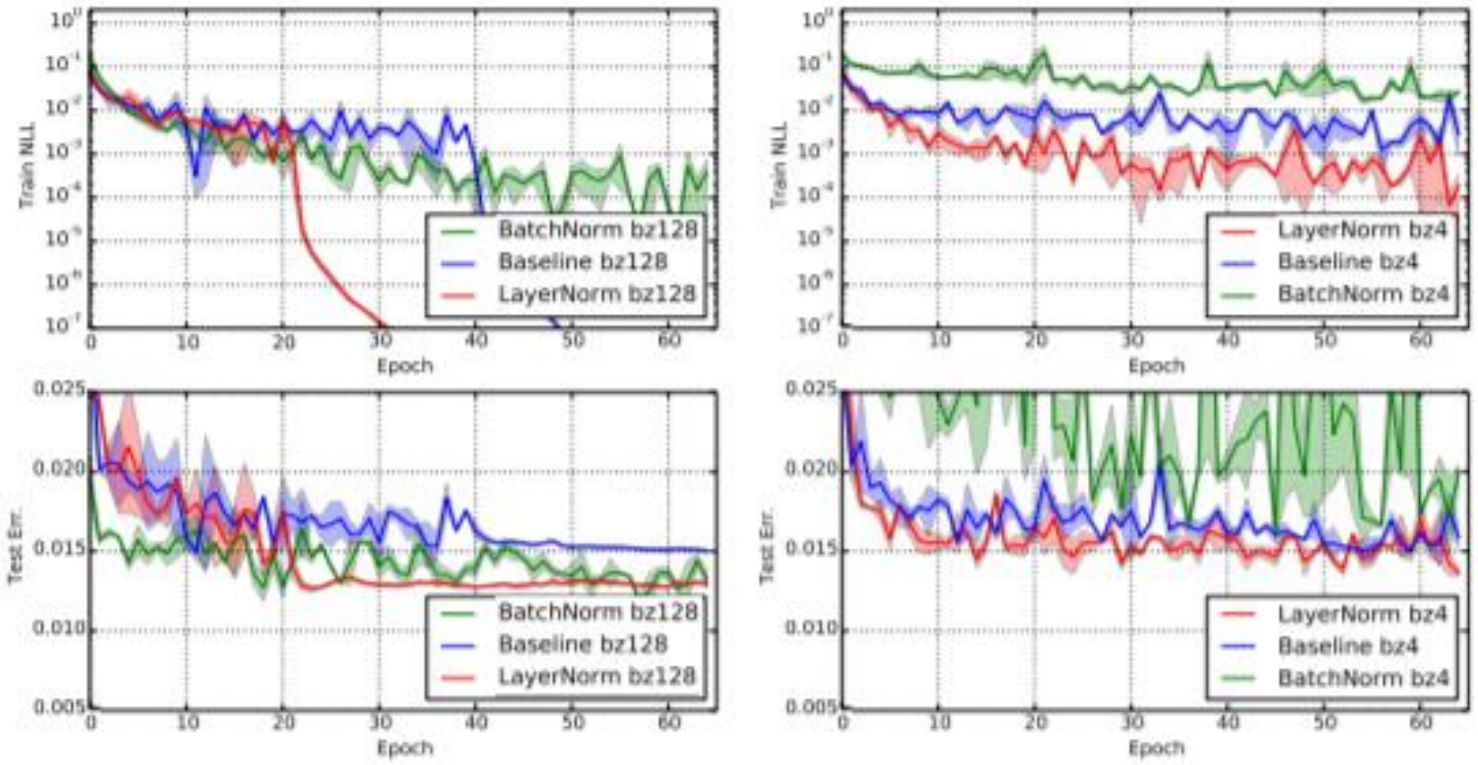
x^t - текущий вход

h^{t-1} - вектор предыдущих состояний

$a^t = W_{hh}h^{t-1} + W_{xh}x^t$ - сумма входов

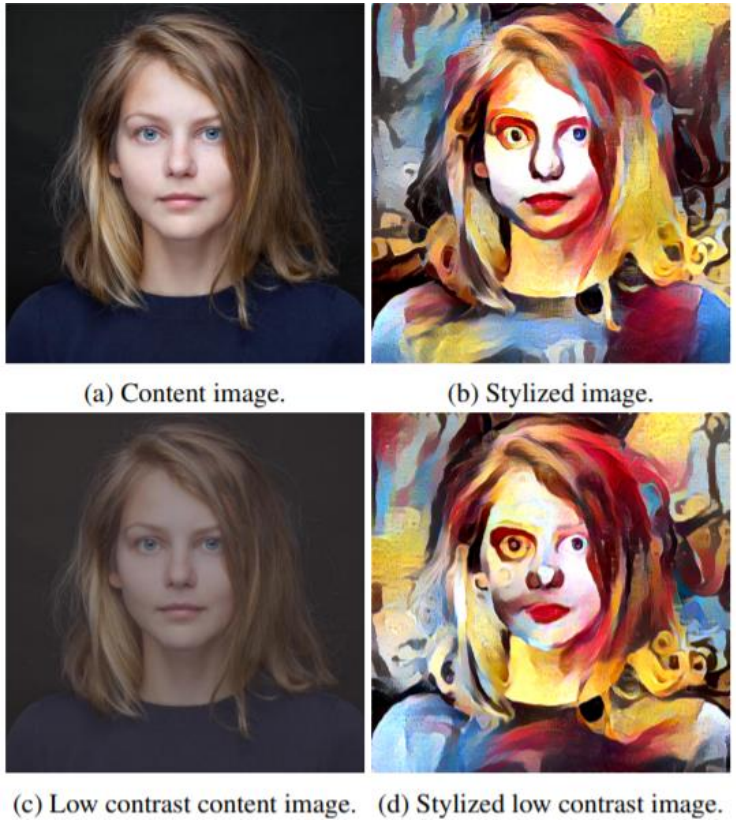
$$\mathbf{h}^t = f\left[\frac{\mathbf{g}}{\sigma^t} \odot (\mathbf{a}^t - \mu^t) + \mathbf{b}\right] \qquad \mu^t = \frac{1}{H}\sum_{i=1}^{H} a_i^t \qquad \sigma^t = \sqrt{\frac{1}{H}\sum_{i=1}^{H}(a_i^t - \mu^t)^2}$$

Нормализация слоев



Instance Norm

- Передача стиля
- Проблема: результат сети зависит от контрастности картинки



(a) Content image. (b) Stylized image.

(c) Low contrast content image. (d) Stylized low contrast image.

Instance Norm

- Вместо нормализации по каждому примеру, нормализируем по каждому каналу.

$$y_{tijk} = \frac{x_{tijk} - \mu_{ti}}{\sqrt{\sigma_{ti}^2 + \epsilon}}, \quad \mu_{ti} = \frac{1}{HW}\sum_{l=1}^{W}\sum_{m=1}^{H} x_{tilm}, \quad \sigma_{ti}^2 = \frac{1}{HW}\sum_{l=1}^{W}\sum_{m=1}^{H}(x_{tilm} - mu_{ti})^2$$

Group Norm

- Считаем среднее и дисперсию по группам каналов
- Является комбинацией Layer Norm и Instance Norm

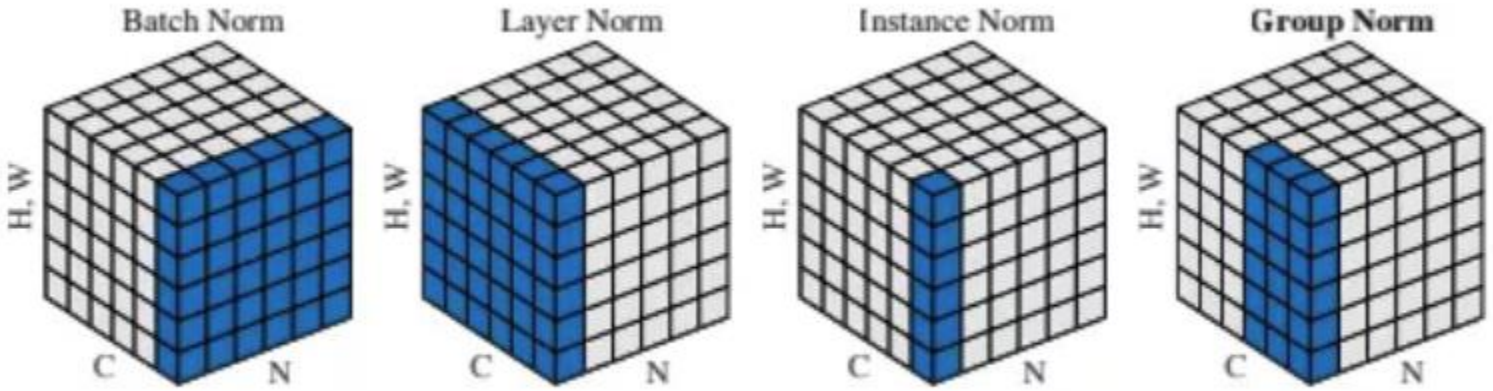
Group Norm

$$\mu_i = \frac{1}{m}\sum_{k\in\mathcal{S}_i} x_k, \quad \sigma_i = \sqrt{\frac{1}{m}\sum_{k\in\mathcal{S}_i}(x_k - \mu_i)^2 + \epsilon},$$

$$\mathcal{S}_i = \{k \mid k_N = i_N, \lfloor \frac{k_C}{C/G} \rfloor = \lfloor \frac{i_C}{C/G} \rfloor\}.$$

$$\mathbf{i} = (\mathbf{i}\mathrm{N}, \mathbf{i}\mathrm{C}, \mathbf{i}\mathrm{H}, \mathbf{i}\mathrm{W})$$

N – Batch
C – Channel
H – Height
W - Width



A visual comparison of various normalization methods

Вопросы

- Какую проблему Batch Norm решает Weight Norm
- Что такое проблема ковариантного сдвига и как Batch Norm ее решает?
- Как Group Norm свести к Instance Norm и Layer Norm?

Источники

- Batch Normalization - https://arxiv.org/pdf/1502.03167.pdf
- Weight Normalization - https://arxiv.org/pdf/1602.07868.pdf
- Layer Normalization - https://arxiv.org/pdf/1607.06450.pdf
- Instance Normalization - https://arxiv.org/pdf/1607.08022.pdf
- Group Normalization - https://arxiv.org/pdf/1803.08494.pdf