

2021 DeepSleep Paper Review

Layer Normalization (2016)

Presenter : Haram Lee

1. 개념

이미지 출처 : Ioffe, Sergey, and Christian Szegedy,
"Batch normalization: Accelerating deep network training
by reducing internal covariate shift.", ICML 2015
<https://arxiv.org/pdf/1502.03167.pdf>

Remind for Batch normalization

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_1 \dots x_m\}$;

Parameters to be learned: γ, β

Output: $\{y_i = \text{BN}_{\gamma, \beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m} \sum_{i=1}^m x_i \quad // \text{ mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_{\mathcal{B}})^2 \quad // \text{ mini-batch variance}$$

$$\hat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \quad // \text{ normalize}$$

$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv \text{BN}_{\gamma, \beta}(x_i) \quad // \text{ scale and shift}$$

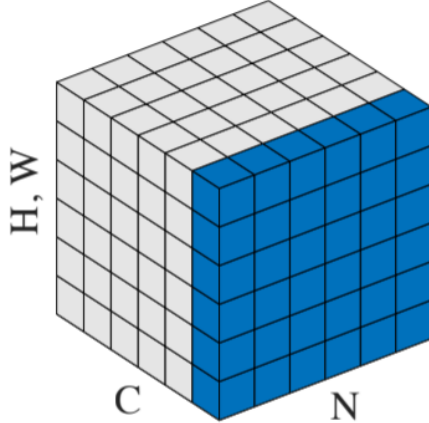
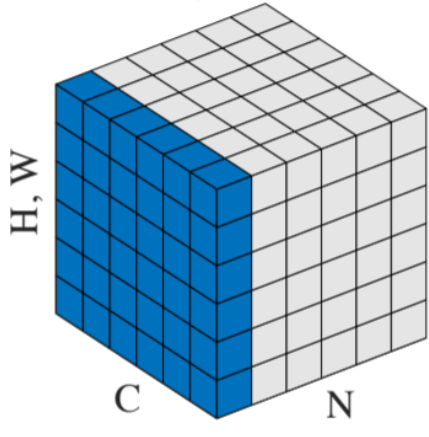
↘ 새로운 파라미터 γ, β

$$z = g(Wu + b)$$



$$z = g(\text{BN}(Wu))$$

1. 개념

정규화	설명	그림	Shape 비교
Batch Norm	각 feature별로, 한 mini-batch에 대한 summed input의 평균과 분산을 구한다.		<ul style="list-style-type: none"> fully-connected networks <ul style="list-style-type: none"> x : $N \times D$ $\mu, \sigma, \gamma, \beta$: $1 \times D$ convolutional networks <ul style="list-style-type: none"> x : $N \times C \times H \times W$ $\mu, \sigma, \gamma, \beta$: $1 \times C \times 1 \times 1$
Layer Norm	각 training case별로, 한 레이어의 모든 뉴런에 대한 summed input의 평균과 분산을 구한다.		<ul style="list-style-type: none"> fully-connected networks <ul style="list-style-type: none"> x : $N \times D$ $\mu, \sigma, \gamma, \beta$: $N \times 1$ convolutional networks <ul style="list-style-type: none"> x : $N \times C \times H \times W$ $\mu, \sigma, \gamma, \beta$: $N \times 1 \times 1 \times 1$

1. 개념

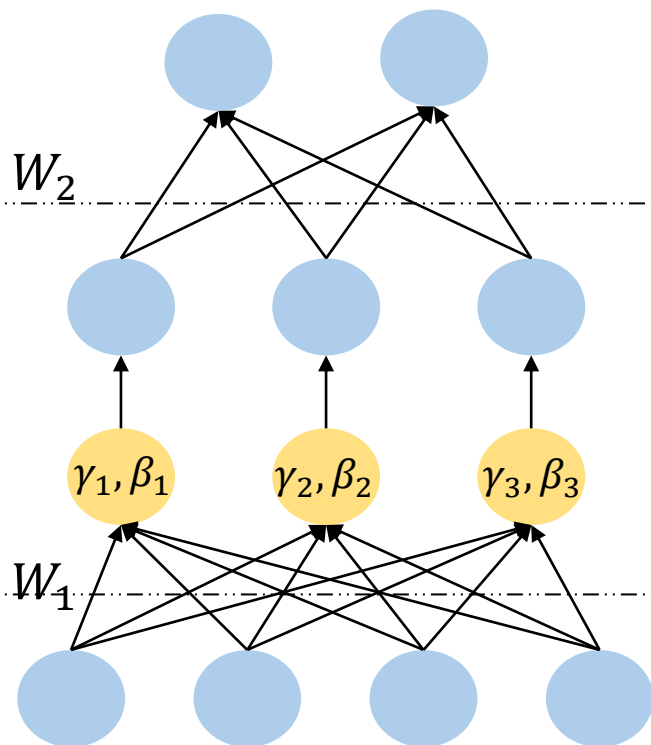
[MLP에서의 비교]

output

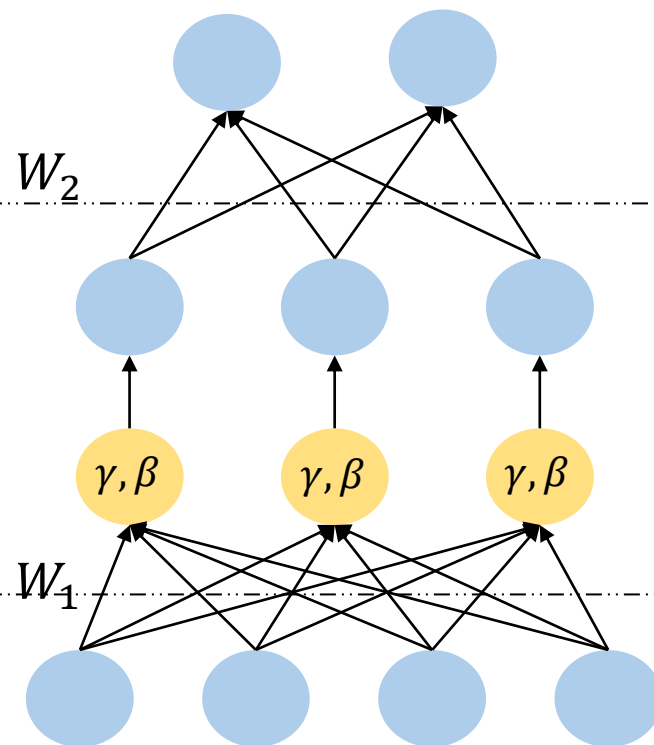
Hidden
layer

Normalization
Layer

input



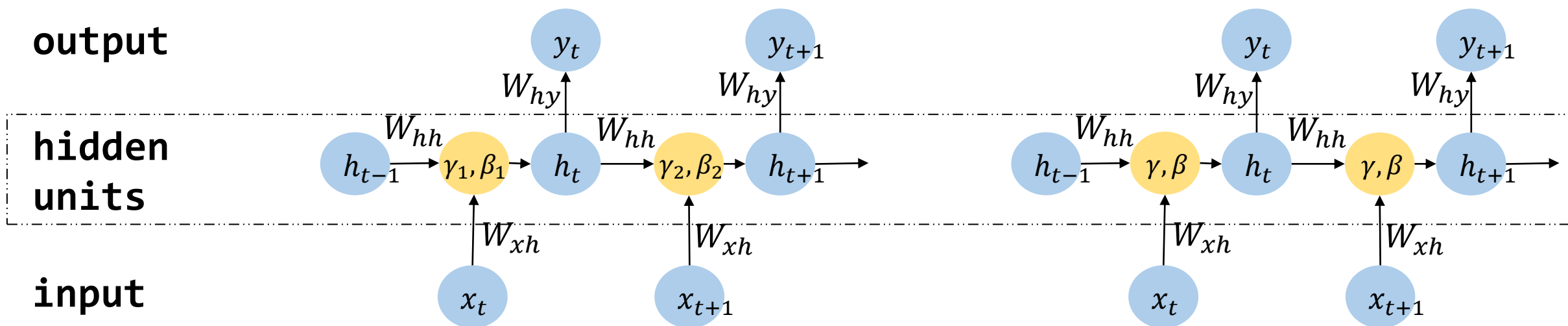
Batch normalization



Layer normalization

1. 개념

[RNN에서의 비교]



Batch normalization

Layer normalization

2. 특징

Batch normalization 의 한계

- RNN의 recurrent neuron에 대한 입력은 종종 시퀀스 길이에 따라 달라지므로, RNN에 적용하기 어렵다.
- μ 와 σ 는 batch size에 따라 달라지기 때문에 batch size 조정에 제약이 있을 수 있다.
- online learning task나 batch size가 작아야 하는 극도로 큰 분산 모델에는 적용할 수 없다.

Layer normalization 의 효과

- RNN 에 적용하였을 때 잘 동작하고, 학습 속도와 성능의 일반화를 향상시킬 수 있다.
- batch size에 제약이 없다.
- batch size=1인 pure online learning에서 사용할 수 있다.

3. 분석

(1) 불변 속성

	Weight matrix re-scaling	Weight matrix re-centering	Weight vector re-scaling	Dataset re-scaling	Dataset re-centering	Single training case re-scaling
Batch norm	Invariant	No	Invariant	Invariant	Invariant	No
Weight norm	Invariant	No	Invariant	No	No	No
Layer norm	Invariant	Invariant	No	Invariant	No	Invariant

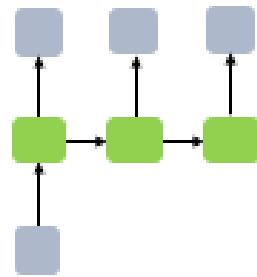
Table 1: Invariance properties under the normalization methods.

(2) Normalization scalar(σ)는 암묵적으로 learning rate를 낮추고 학습을 더 안정적으로 만든다.

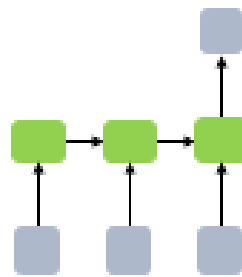
- weight norm이 커질 경우, weight space 방향의 곡률이 원만해진다.
- 이를 통해, 학습 중에 weight norm이 큰 weight vector의 방향을 변경하기가 더 어렵다.
- 따라서 정규화 방법은 weight vector에 암묵적인 "early stopping" 영향을 미치며 수렴을 위한 학습을 안정화 하는 데 도움이 된다.

4. 실험

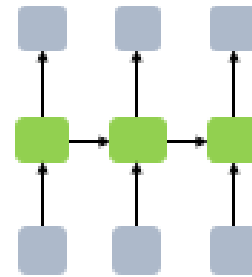
- 1) Image-sentence ranking
- 2) Question-answering
- 3) Contextual language modelling
- 4) Generative modelling
- 5) Handwriting sequence generation
- 6) MNIST classification



일 대 다(one-to-many)



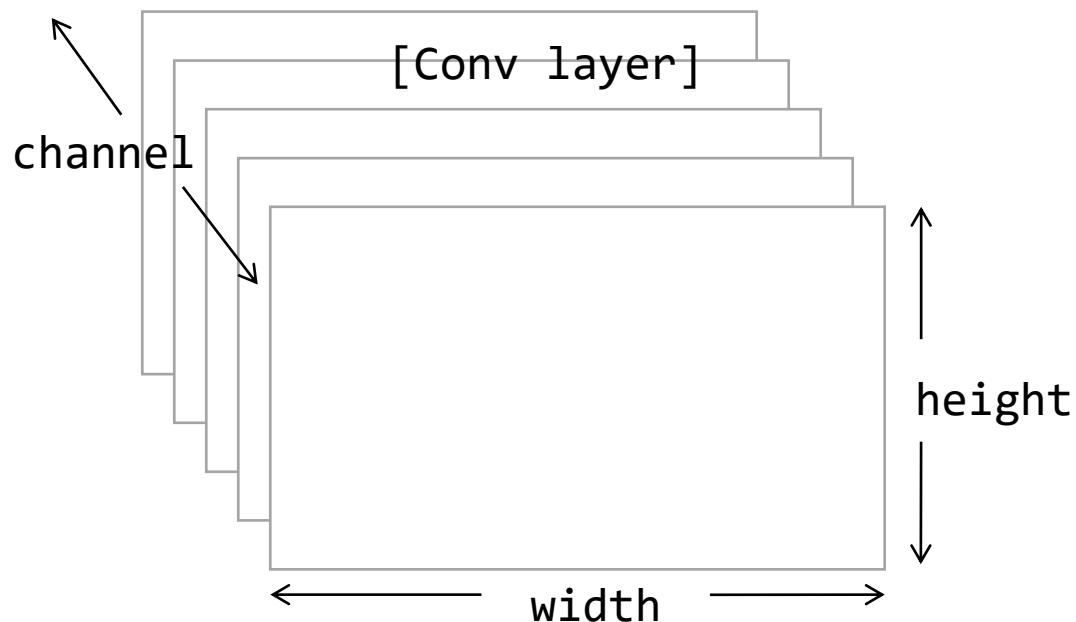
다 대 일(many-to-one)



다 대 다(many-to-many)

5. 결과

- 대체로 수렴 속도가 빨라지고, 베이스라인 모델보다 성능이 좋아졌다.
- CNN에서, 베이스라인 모델에 비해 속도 개선은 되지만, 성능은 Batch Normalization을 적용한 것이 더 좋다.
- 실험을 통해 경험적으로, 특히 long sequence와 small mini-batch를 가지는 RNN이 Layer normalization의 가장 큰 혜택을 받는다.
- (+추가) Transformer 등의 모델에서 사용한다.



Q & A