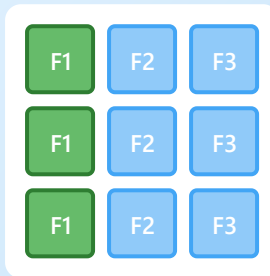


Layer Normalization (Layer Norm)

Batch Norm vs Layer Norm

Batch Norm



Across Batch ↓

Layer Norm



Across Features →

Normalizes across **feature dimension** per sample

✓ Independent of Batch Size

Works with batch size = 1
Each sample normalized independently

Key Features

Independent of batch size during training/inference

Each sample normalized **independently**

No dependence on **batch statistics**

Essential for **RNNs and Transformers** (sequential models)

✓ Standard in NLP

Used in modern NLP architectures:
BERT, GPT, and Transformers



Layer Normalization Formula

$$\mu = (1/H) \sum x_i$$

Mean: Average value of all features in the layer

$$\sigma^2 = (1/H) \sum (x_i - \mu)^2$$

Variance: Average of squared deviations from the mean

$$\hat{x} = (x - \mu) / \sqrt{(\sigma^2 + \epsilon)}$$

Normalization: Transform to mean 0 and variance 1

$$y = \gamma \odot \hat{x} + \beta$$

Scale & Shift: Adjust with learnable parameters

γ (**gamma**): Scale parameter (learnable)

β (**beta**): Shift parameter (learnable)

ϵ (**epsilon**): Small constant for numerical stability (e.g., 10^{-5})

H : Number of hidden units in the layer

What is Internal Covariate Shift?

Internal Covariate Shift refers to the phenomenon where the distribution of inputs to each layer continuously changes during neural network training. As the parameters of previous layers are updated, the statistical characteristics of the inputs received by the current layer change with each iteration.

Problems that Arise

- Training instability: Each layer must constantly adapt to changing input distributions
- Slow convergence: Requires smaller learning rates
- Gradient problems: Risk of vanishing/exploding gradients
- Worse in deep networks: Cumulative effect increases with depth

Layer Normalization Solution

- Stabilizes input distribution: Normalizes each sample independently
- Faster training: Enables larger learning rates
- Improved gradient flow: Stabilizes gradients during backpropagation
- Consistent representations: Model can learn stable features



Why Layer Norm is Used in Transformers

1 Independent Sequence Processing

Transformers process each input sequence independently. Rather than sharing statistics across batches like Batch Normalization, normalization needs to be applied independently for each sample (sentence).

2 Variable-Length Sequence Handling

Sentences vary in length, and each token carries different meanings. Layer Norm normalizes across the feature dimension of each token, operating stably regardless of sequence length.

3 Self-Attention 안정화

Self-Attention 메커니즘에서 각 토큰은 시퀀스의 모든 토큰을 참조합니다. Layer Normalization은 이러한 attention 연산 전후에 적용되어 표현을 일관되게 유지하고 학습 수렴을 빠르게 합니다.

4 배치 크기 독립성

추론(inference) 시 배치 크기가 1일 수 있습니다. Layer Norm은 배치 크기에 의존하지 않으므로 학습과 추론에서 동일하게 동작하여 안정적입니다.



주요 NLP 모델에서의 Layer Norm 적용

BERT

GPT-2/3/4

T5

RoBERTa

BART