Training Day-95 Report:

Training Day Report:

What is Batch Normalization?

Batch normalization is a deep learning technique used to improve the training process of neural networks by normalizing the inputs of each layer. It ensures that the data being fed into a layer has a consistent mean and variance, which accelerates convergence and stabilizes learning.

Key Concepts of Batch Normalization

1. **Normalization of Inputs:**
   - Batch normalization standardizes the output of the previous layer by subtracting the batch mean and dividing by the batch standard deviation.
   - This ensures that the input to the next layer has a mean of 0 and a standard deviation of 1.
2. **Learnable Parameters (Scale and Shift):**
   - Unlike traditional normalization, batch normalization introduces two additional parameters, **γ (scale)** and **β (shift)**, which are learned during training.
   - This allows the network to retain the flexibility to adjust the normalized outputs.
3. **Integration During Training:**
   - Batch normalization is applied before or after the activation function within each layer.
   - It works differently during training and inference:
     - **Training:** Uses the mean and variance of the current batch.
     - **Inference:** Uses a running mean and variance calculated during training.

Advantages of Batch Normalization

- **Faster Convergence:** Reduces internal covariate shift, allowing the model to train faster.
- **Higher Learning Rates:** Mitigates the risk of divergence, enabling the use of larger learning rates.

- **Regularization Effect:** Reduces overfitting by introducing a slight noise through mini-batch statistics.
- **Stabilizes Learning:** Handles variance in feature distributions, making the model more robust.

Mathematical Representation:

For each feature xix_i in a batch:

x^i=xi−μσ2+ϵ\hat{x}_i = \frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}}

Where:

- μ\mu: Mean of the batch.
- σ2\sigma^2: Variance of the batch.
- ϵ\epsilon: Small constant to prevent division by zero.

The normalized output is then scaled and shifted using:

yi=γx^i+βy_i = \gamma \hat{x}_i + \beta

Where γ\gamma and β\beta are learnable parameters.

Applications of Batch Normalization

- Image classification tasks (e.g., CNNs in computer vision).
- Sequence modeling tasks (e.g., RNNs and LSTMs).
- Deep networks prone to vanishing or exploding gradients.

Batch normalization has become a standard component in most modern neural network architectures, significantly enhancing their performance and training efficiency.