GroupNorm

A Preprint

Abstract

Authors suggest GroupNorm as an alternative to BatchNorm as is has significant problems when it comes to computer vision such as small batch size which is used for memory purposes but using BatchNorm error increases rapidly.

1 Introduction

A family of feature normalization methods, including BN, LN, IN, and GN, perform the following computation:

$$\hat{x_i} = \frac{1}{\sigma_i}(x_i - \mu_i)$$

where $x-$ feature, in case of 2D images $i = (i_N, i_C, i_H, i_W)$ where N - batch axis, C - channels, H, W - height and width respectively

μ_i, σ_i defined as

$$\mu_i = \frac{1}{m}\sum_{k \in S_i} x_k \qquad \sigma_i = \sqrt{\frac{1}{m}\sum_{k \in S_i}(x_k - \mu_i)^2 + \epsilon} \tag{1}$$

ϵ - small constant, $\mathcal{S}_i$ is the set of pixels in which the mean and std are computed(m is the size of the set).

$$\text{In BatchNorm} \qquad \mathcal{S}_i = \{k | k_C = i_C\} \tag{2}$$

$$\text{In LayerNorm} \qquad \mathcal{S}_i = \{k | k_N = i_N\} \tag{3}$$

$$\text{In InstanceNorm} \qquad \mathcal{S}_i = \{k | k_N = i_N, k_C = i_C\} \tag{4}$$

In BatchNorm μ and σ are computated along (N, H, W) axis which basically means

$$\mu_i = \frac{1}{N \cdot H \cdot W} \cdot \sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{n,c,h,w}$$

In LayerNorm μ and σ are computated along (C, H, W)

In InstanceNorm μ and σ are computated along (H, W)

And also! all these methods learn linear transformation o compensate for the possible lost of representational ability:

$$y_i = \gamma \hat{x_i} + \beta_i$$

2 GroupNorm

Formally, GroupNorm is defined as

$$\mathcal{S}_i = \{k | k_N = i_N, [\frac{k_C}{C \cdot G}] = [\frac{i_C}{C \cdot G}]\} \quad \text{where G - number of groups (predefined hyperparemeter =32)}$$

The main point in GroupNorm is that When the batch size is small in different normalizations, these statistics become noisy and unreliable estimates of the true underlying data distribution. This instability leads to increased error, but in GroupNorm it computes its normalization statistics independently of the batch size. For each sample in a batch, GN divides the channels into a predefined number of groups (G). Then, for each group within that single sample, it calculates the mean and variance using only the features belonging to that group and that sample, across their spatial dimensions (H, W). Because the calculation is confined to a single sample (and its groups of channels), the batch size (N) doesn't influence the statistics themselves. This makes GN's performance stable even with very small batch sizes