Research Report on SEResNet

Architecture Overview

Squeeze-and-Excitation Networks (SENets) are an enhancement of standard convolutional neural networks (CNNs) that introduce a new architectural unit called the Squeeze-and-Excitation (SE) block. **The primary motivation behind SENets is to improve the representational power of a network by explicitly modeling the interdependencies between the channels of its convolutional features**. This is achieved by adaptively recalibrating channel-wise feature responses, allowing the network to emphasize informative features and suppress less useful ones. SENets can be integrated into existing architectures, such as ResNet, by inserting SE blocks into their modules, resulting in variants like SE-ResNet (SEResNet)

Detailed Explanation of the Squeeze-and-Excitation Block

Motivation

Traditional CNNs learn spatial and channel-wise features, but the relationships between channels are typically implicit and entangled with spatial correlations. The SE block aims to explicitly model these channel-wise dependencies, enabling the network to focus on the most informative features for a given input

How it works

The SE block operates in two main steps:

- **Squeeze (Global Information Embedding)**:
  The output feature maps from a convolutional transformation are aggregated across their spatial dimensions using global average pooling. This produces a vector of channel-wise statistics, summarizing the global spatial information for each channel.

$$z_c = \frac{1}{H * W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j)$$

  where u_c is the feature map for channel c, and H, W are its height and width.

- **Excitation (Adaptive Recalibration)**:
  The aggregated channel descriptors are passed through a small bottleneck of two fully connected layers with a ReLU activation in between, followed by a sigmoid activation. This produces a set of weights (one per channel) that represent the importance of each channel for the current input.

$$s = \sigma(W_2 \,.\, \delta(W_1 z))$$

- **Feature Recalibration**:
  The original feature maps are then rescaled (channel-wise multiplication) by these weights, enhancing or suppressing each channel according to its learned importance.

$$\hat{x}_c = s_c \,.\, u_c$$

Integration into ResNet

In SE-ResNet, the SE block is inserted into the residual module. Specifically, the SE block is applied to the output of the non-identity branch (i.e., after the convolutional transformations and before the addition with the skip connection). This recalibrated output is then summed with the identity branch, as in the standard residual block

Comparison: Standard ResNet vs. SEResNet

Aspect	Standard ResNet	SEResNet (SE-ResNet)
Channel Attention	None (implicit only)	Explicit, via SE blocks
Residual Block	Convolutions + identity skip connection	Convolutions + SE block + identity skip connection
Parameters	Baseline	$\sim 10\%$ more (mostly in SE blocks' FC layers)
Computation	Baseline (e.g., 3.86 GFLOPs for ResNet-50)	Slightly higher (e.g., 3.87 GFLOPs for SE-ResNet-50)
Performance (ImageNet Top-5 Error, ResNet-50)	7.48%	6.62% (0.86% absolute improvement)
Performance (ImageNet Top-1 Error, ResNet-50)	24.80%	23.29% (1.51% absolute improvement)
Flexibility	Standard residual structure	SE blocks can be inserted at any depth or module

*Performance numbers are from single-crop evaluations on ImageNet

Improvement in Accuracy and Performance

- **Explicit Channel Modeling**:
  By learning to recalibrate channel-wise feature responses based on global context, SE blocks help the network focus on the most relevant features for each input, improving discriminative power
- **Dynamic Feature Selection**:
  The excitation mechanism introduces input-dependent dynamics, allowing the network to adaptively emphasize or suppress features, which is especially beneficial in deeper layers where class-specific features are more important
- **Minimal Overhead**:
  The computational and parameter increase is modest compared to the performance gains. For example, SE-ResNet-50 achieves nearly the same accuracy as a much deeper ResNet-101, but with half the computational cost
- **General Applicability**:
  SE blocks have been shown to improve a wide range of architectures (e.g., VGG, Inception, ResNeXt, MobileNet, ShuffleNet) and tasks (classification, detection, scene recognition), demonstrating that the approach is broadly effective and not limited to a specific architecture or dataset
- **Empirical Results**:
  SE-ResNet models consistently outperform their non-SE counterparts across various depths and tasks. For instance, SE-ResNet-50 outperforms ResNet-50 by 0.86% in top-5 error and 1.51% in top-1 error on ImageNet, with only a 0.26% increase in computational cost

Inference

After running the model on test dataset, we get an accuracy of 85.81%. The loss curve is shown as below: