Theory and Method for Neuron Sparsification using Variational Dropout

1 Method

1.1 Prior Work: Variational Dropout

Variational Dropout is a technique for sparsifying the weights of neural networks by introducing a centered Gaussian prior for the weights $p(W)$. The goal is to approximate the posterior distribution $q(W)$ using a variational distribution, which is parameterized and optimized by maximizing the Evidence Lower Bound (ELBO):

$$\mathcal{L}(q, \theta) = \mathbb{E}_q \log p(Y|W, X) - KL(q||p) \tag{1}$$

This method encourages many weights to shrink towards zero, allowing for weight sparsification.

1.2 Extending to Neuron Sparsification

While Variational Dropout successfully sparsifies weights, it doesn't directly reduce the number of neurons. To extend this technique, I modify the prior distribution to encourage neuron-level sparsification. Instead of assigning a different scale to each weight, I apply the same scale to all weights connected to a given output neuron.

$$p(W|\sigma) = \prod_{ij} \mathcal{N}(W_{ij}|0, \sigma_i^2) \tag{2}$$

This formulation encourages entire neurons to either remain active (non-zero weights) or become inactive (weights close to zero).

1.3 Mixture of Gaussian Prior

To improve flexibility, the prior distribution is extended to a mixture of two Gaussian:

$$p(W|\theta) = \prod_{ij} \left(p_i \mathcal{N}(W_{ij}|0, {\sigma_1}_i^2) + (1 - p_i)\mathcal{N}(W_{ij}|0, {\sigma_2}_i^2)\right) \tag{3}$$

This allows for a more nuanced control over the weights connected to a neuron, enabling a combination of large and small weights while still promoting neuron sparsification.

1.4 Optimization Procedure

The optimization goal is to maximize the ELBO with respect to both the variational distribution q and the prior parameters θ. The ELBO consists of two terms: the expected log-likelihood and the KL divergence between the variational posterior and the prior.

For optimizing with respect to θ, the relevant term is:

$$-KL(q||p) = -\mathbb{E}_q \log \frac{q(W)}{p(W|\theta)} = \mathbb{E}_q \log p(W|\theta) + const \tag{4}$$

Thus, we need to maximize $\mathbb{E}_q \log p(W|\theta)$ with respect to θ. As derived in the next section, the parameters σ_1, σ_2, and p_i can be estimated using an Expectation-Maximization (EM) algorithm for Gaussian mixtures, treating the observed weights as $\tilde{W}_{ij} = \sqrt{\mu_{ij}^2 + s_{ij}^2}$. After updating θ, the ELBO can be optimized with respect to q.

1.5 Neuron Sparsification Criterion

Neuron sparsification is achieved when the dropout rate of all weights connected to a neuron is high. A weight is considered irrelevant if its dropout rate exceeds 0.99. Consequently, a neuron is irrelevant if all its connected weights are irrelevant.

This approach is applied to fully connected and convolutional layers. The same methodology can be generalized to other types of layers such as Multi-Head Attention.

2 KL Minimization for Neuron Sparsification

To optimize the objective w.r.t. the parameters $\theta = (\sigma_1, \sigma_2, p)$, we need to maximize the KL-term:

$$-KL(p||q) = \mathbb{E}_q \log q(W) - \mathbb{E}_q \log p(W|\theta) = \mathbb{E}_q \log p(W|\theta) + const \tag{5}$$

Therefore, we solve the optimization problem:

$$\mathbb{E}_q \log p(W|\theta) \to \max_\theta \tag{6}$$

2.1 Latent Variable Introduction

We introduce latent variables z_{ij}, indicating the assignment of each weight W_{ij} to one of the Gaussian components. This results in:

$$p(W, z|\theta) = \prod_{ij} \left(p_i \mathcal{N}(W_{ij}|0, \sigma_1^2)\right)^{z_{ij}} \left((1-p_i)\mathcal{N}(W_{ij}|0, \sigma_2^2)\right)^{1-z_{ij}} \tag{7}$$

We derive a lower bound for the objective function by incorporating a variational distribution $r(z)$:

$$\begin{gathered}\mathbb{E}_q \log p(W|\theta) = \mathbb{E}_q \mathbb{E}_r \left(\log p(W, z|\theta) - \log r(z)\right) + \mathbb{E}_q KL(r(z)\|p(z|W, \theta)) \geq \\ \mathbb{E}_r \left(\mathbb{E}_q \log p(W, z|\theta) - \log r(z)\right) = \mathbb{E}_r \left(\log p(\tilde{W}, z|\theta) - \log r(z)\right) = \mathcal{L}_{GM}(\theta, r)\end{gathered} \tag{8}$$

where $\tilde{W}_{ij} = \sqrt{\mu_{ij}^2 + s_{ij}^2}$. Now we can note that it is the same lower bound as for the Gaussian Mixture model where observed data is $\tilde{W}$.

2.2 Gaussian Mixture Model Optimization

To solve this optimization problem, we apply the Expectation-Maximization (EM) algorithm, which is commonly used for Gaussian mixture models. After each EM step, the lower bound $\mathcal{L}_{GM}(\theta, r)$ will converge to the log-likelihood $\log p(\tilde{W}|\theta)$:

$$p(\tilde{W}|\theta) = \prod_{ij} \left(p_i \mathcal{N}(\tilde{W}_{ij}|0, \sigma_1^2) + (1-p_i)\mathcal{N}(\tilde{W}_{ij}|0, \sigma_2^2)\right) \tag{9}$$

2.3 Conclusion

In the optimization step, the resulting lower bound $p(\tilde{W}|\theta)$ is used to replace the original expectation $\mathbb{E}_q \log p(W|\theta)$. This allows us to effectively optimize the variational dropout framework for neuron sparsification.