Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer

Main Idea:

- To propose a way to improve model capacity, training time and model quality through a conditional computation approach that alternates between dense LSTM and MoE blocks.

Approach:

- Introduces a new neural network component (a new block/layer) which consists of:
    - n experts, each a feed-forward neural network
    - a trainable gating network, which selects a sparse combination of these experts to process each input token given.
- The gating network presented is an improvement over the standard approach, which trains a weight matrix to give score to an input x and pass that to a softmax (gating output $G(x) = Softmax(W_g * x)$. The gating mechanism proposed is called noisy top-k routing, which adds noise and sparsity:
    - Gaussian noise is added before taking the softmax to help with load balancing between experts during training.
        - $H(x) = (W_g * x) + StandardNormal() * SoftPlus((W_{noise} * x)_i)$
    - Sparsity is added by taking only the top k scores given by the gating mechanism.
        - $G(x) = Softmax(KeepTopK(H(x), k)$
        - If not in the top k, H(x) becomes -inf so it is not considered in the final output.

- To balance expert utilization, an auxiliary term to the loss is added, which works by being computed at a batch level.
    - For each expert and the training batch X, take the expert's importance in the batch:
        - $Importance(X) = \sum_{x \in X} G(x)$
        - Importance(X)e = sum of all the expert's G(x) for the batch
    - The term Limportance is added to the loss (which will be computed at the batch level) to encourage all experts to have equal importance:
        - $L_{importance}(X) = W_{importance} * V(importance(X)$
        - $W_{importance}$ is a hand-tuned scaling factor and V is the coefficient of variation squared.
- The final network consists of alternating LSTM blocks with these new MoE blocks.

My takeaways:

- This approach means that for the first time MoE was used as a network component and not as the network itself, providing a method to integrate it with dense layers.
- Introduced top-k routing.
- Experiments showed that experts tend to become specialized on syntax and semantics, which is an important follow-up to the findings of the "Learning Factored Representations…" paper which hinted that different experts specialize in different clusters of the data.

- This paper also provides advancements in load balancing, crafting an auxiliary loss term for load balancing that seems much more effective than the previous method of pausing the training of highly utilized experts.