Learning slope for DeepZ relaxation

Xinyuan Huang **Shaohua Li** **Yiqun Liu**

1 Method

We treat slopes as learnable parameters. We use the idea of translating logical constraints into loss function introduced in DL2[1], and update slopes by optimizing the loss function. For true label j, and last layer neurons $n_i, 0 \leq i \leq 9$

$$n_i = a_0^i + \sum_k a_k^i \epsilon_k, 0 \leq i \leq 9 \tag{1}$$

$$n_j - n_i = a_0^j - a_0^i + \sum_k (a_k^j - a_k^i)\epsilon_k \tag{2}$$

For each label i, we choose $\epsilon_k \in \{-1, +1\}$ depending on the sign of $(a_k^j - a_k^i)$ to get the lower bounds lb_i of $n_j - n_i$. To verify the case, our logic is $\wedge_{i \neq j} lb_i \geq 0$, this can be translated into following loss:

$$\text{loss} = \sum_{i, i \neq j} max(0, -lb_i) \tag{3}$$

We also introduce a loss mask $[m_0, m_1, ..., m_9], m_i \in \{1, 0\}$ to block out labels which already have been proved to be not larger than true label, namely in some round, we find that for label t, $lb_t \geq 0$, then we set m_t to 0. So the real loss during verification is:

$$\text{loss} = \sum_{i, i \neq j} max(0, -lb_i \times m_i) \tag{4}$$

Our main criterion of verifying a case is in an optimization round, if it holds that $\forall 0 \leq i \leq 9, lb_i \times m_i \geq 0$, then we successfully verify this case. We also use an auxiliary criterion, we keep updating the lower bound of true label j by assigning -1 or +1 to ϵ_k depending on the sign of a_k^j, and increasing the lower bound of j if possible. We also keep updating the upper bound of other labels by decreasing the upper bound of other labels in a similar manner. If we find that the lower bound of true label is not smaller than upper bound of other labels, we successfully verify this case.

2 Implementation

We run a round of DeepZ original implementation to initialize slopes. For relu neurons whose range don't cross 0, we initialize their slopes as 0.5, although we don't use their slopes in this initialization round.

Then we begin iterations of optimization. We switch between two optimization strategies: (1) Optimize all relu layers' slopes together (2) Only optimize the last relu layer's slope, freeze other relu layers' slopes. The switch condition is when loss doesn't decrease for specified number of rounds.

References

[1] Dana Drachsler-Cohen Timon Gehr Ce Zhang Martin Vechev Marc Fischer, Mislav Balunovic. Dl2: Training and querying neural networks with logic. In *International Conference on Machine Learning*, 2019.