Evolution Strategies for Approximating Gradients

CMPUT 328

Nilanjan Ray

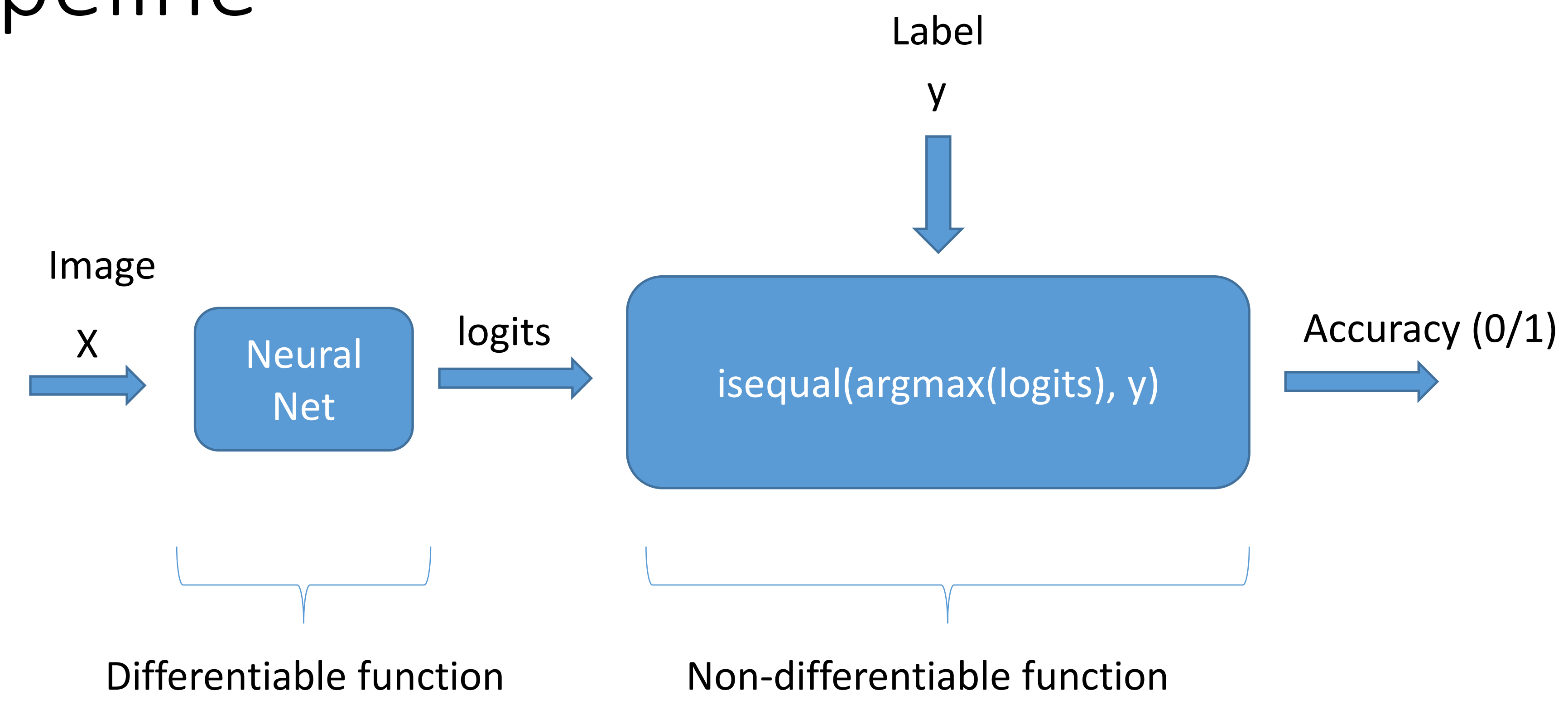
Differentiable Programming

- Our familiar example is neural networks: parameterized differentiable function
- Software packages use chain rules to compute derivatives to backpropagate for learning parameters
- Differentiable pipeline can be more general than neural net
- In fact, any program where each computation is a differentiable function can make use of auto derivative
- PyTorch and Eager mode of TensorFlow use this kind of “auto-diff”
- This framework is called “differentiable programming”
- Differentiable programming is more general than deep learning or neural network

More general differentiable programming?

- What if you have a function that is non-differentiable within a processing pipeline?
- How do we compute derivative across a non-differentiable function?
- We can apply a type of sampling technique called “evolution strategies” (ES) to overcome this kind of obstacles
- ES has been applied as a powerful alternative to reinforcement learning recently (https://arxiv.org/abs/1703.03864)
- I will show here how we can use ES in an image classification pipeline, where a part of the pipeline is non-differentiable

Motivating example: An image classification pipeline



Goal: maximize accuracy by optimizing parameters of the neural net

We cannot apply chain rule or auto-diff, because a part of the pipeline is non-differentiable

Evolution strategies to approximate gradient

- Apply ES to compute approximate gradient of logits
- Push gradient of logits backward to neural net for chain-rule or auto-diff
- A hybrid method: apply chain rule to differentiable parts and apply ES to non-differentiable parts

Algorithm for approximate gradient computation by ES

- Initialize g to a zero vector
- For n=1, 2,…, N
    - Sample a vector q_n from a zero-mean Gaussian of variance t
    - Compute accuracy as f_n = isequal(argmax(logits + q_n), y)
    - Update gradient vector: $g = g + f_n q_n$
- Return g / (tN) as approximate gradient of logits

Disadvantages

- ES is not sample efficient: if logits is a very long vector, N needs to be a really large integer, leading to slow or poor optimization
- How to choose the variance parameter t?