Revision

Backpropagation

- We need backpropagation to train:
  - A: Linear regression
  - B: Polynomial regression
  - C: Perceptron
  - D: Multi Layer Perceptron
  - E: U-Net
  - F: Transformer
  - G: Generative Adverserial Networks

Backpropagation

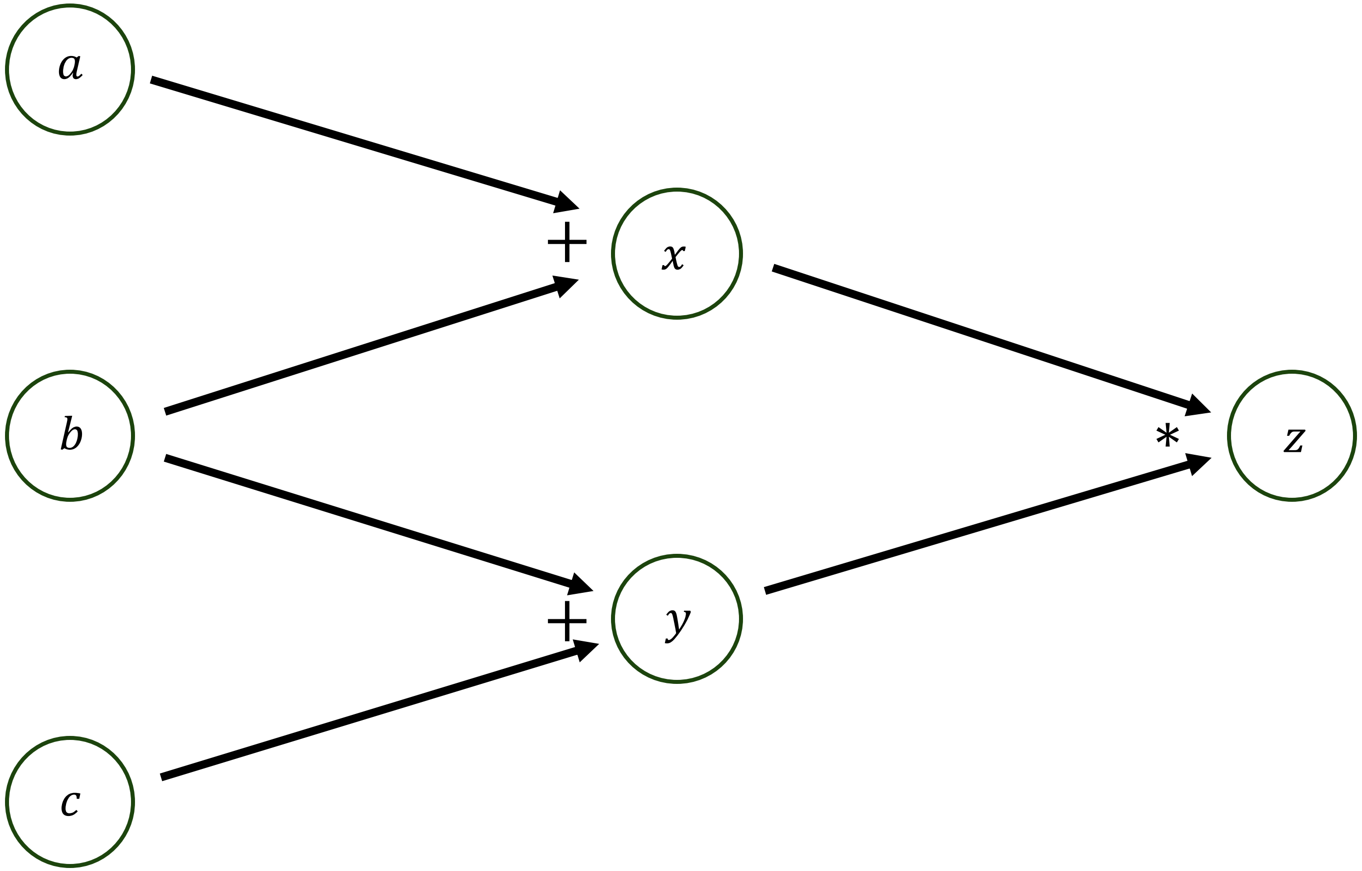
- We need backpropagation to train:
    - A: Linear regression
    - B: Polynomial regression
    - C: Perceptron
    - **D: Multi Layer Perceptron**
    - **E: U-Net**
    - **F: Transformer**
    - **G: Generative Adverserial Networks**

Backpropagation

Draw a computation graph for $z = (a + b) * (b + c)$

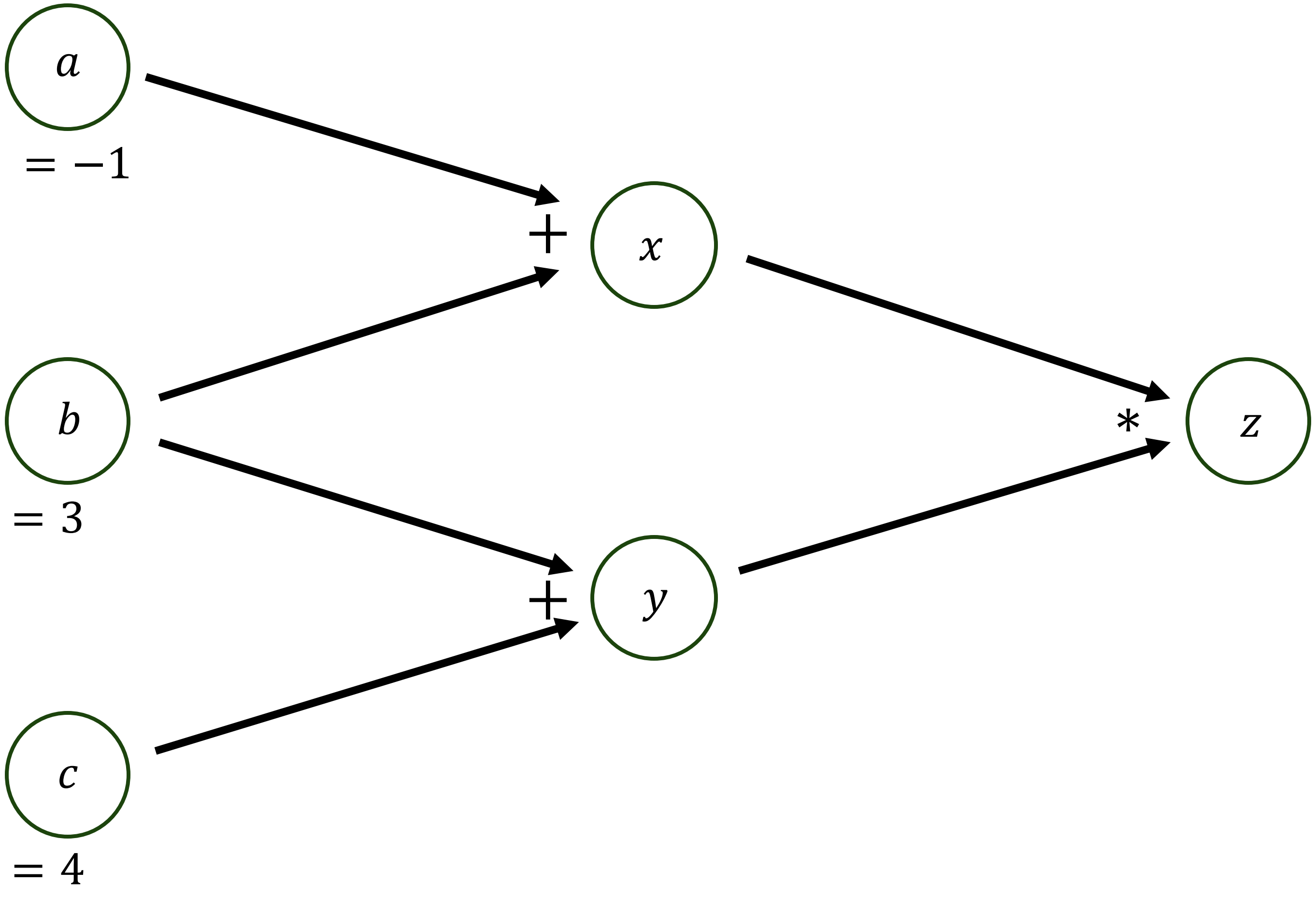
Computation Graph

$$z = \underbrace{(a+b)}_{x} * \underbrace{(b+c)}_{y}$$



Forward Pass

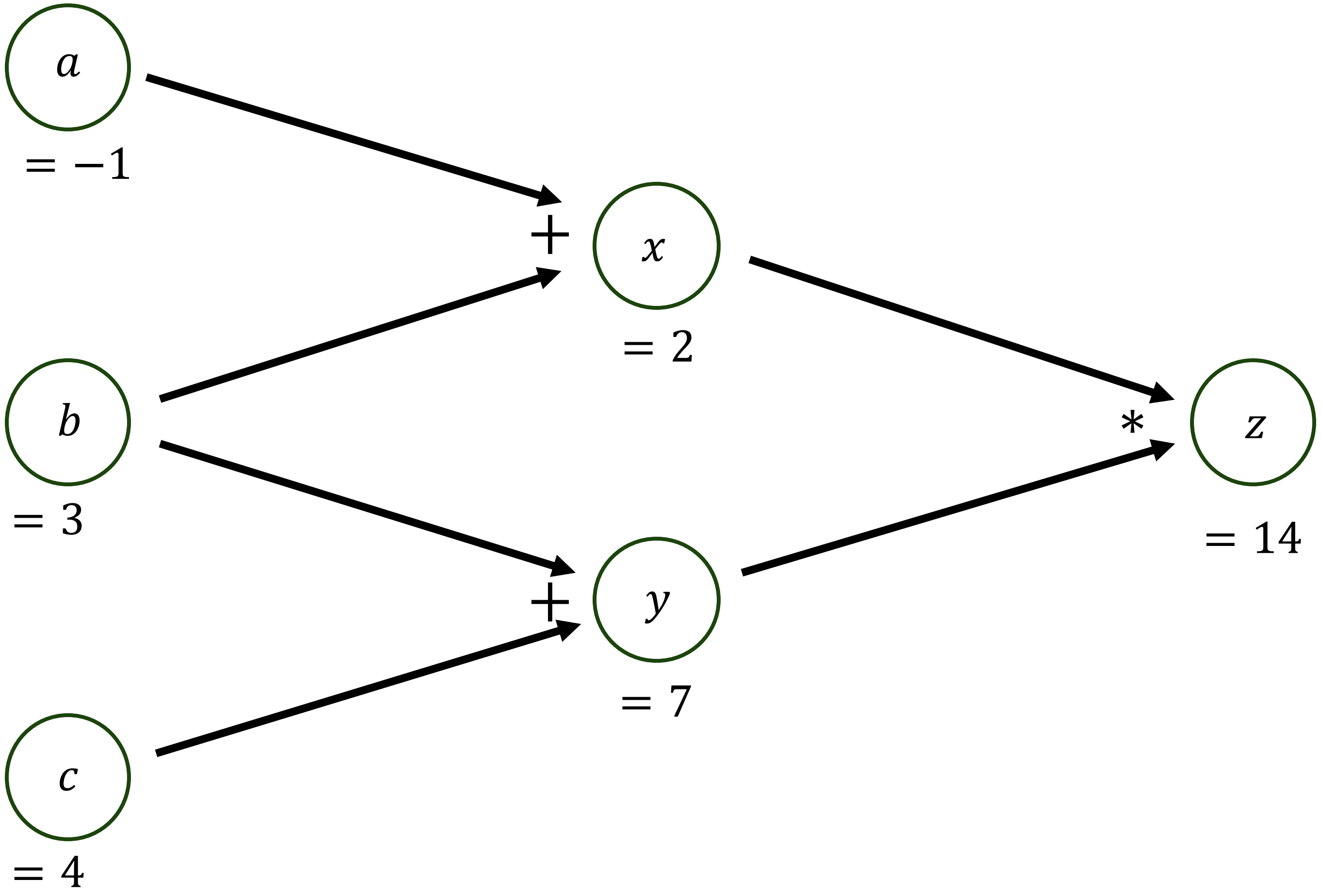
$$z = \underbrace{(a+b)}_{x} * \underbrace{(b+c)}_{y}$$



Forward Pass

$$z = \underbrace{(a+b)}_{x} * \underbrace{(b+c)}_{y}$$

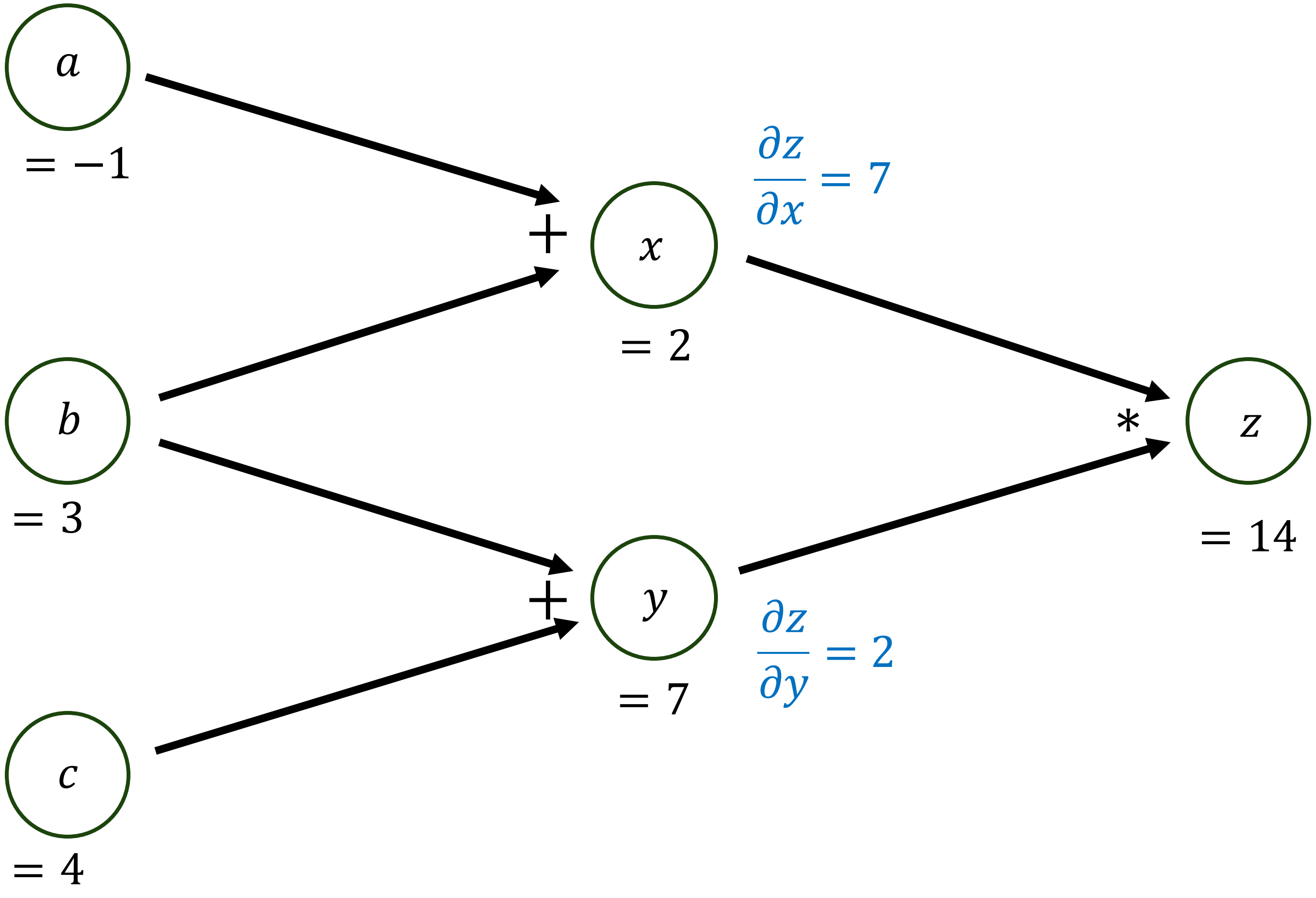
$a = -1 \qquad b = 3 \qquad c = 4$



Backward Pass

$$z = \underbrace{(a+b)}_{x} * \underbrace{(b+c)}_{y}$$

$a = -1 \qquad b = 3 \qquad c = 4$

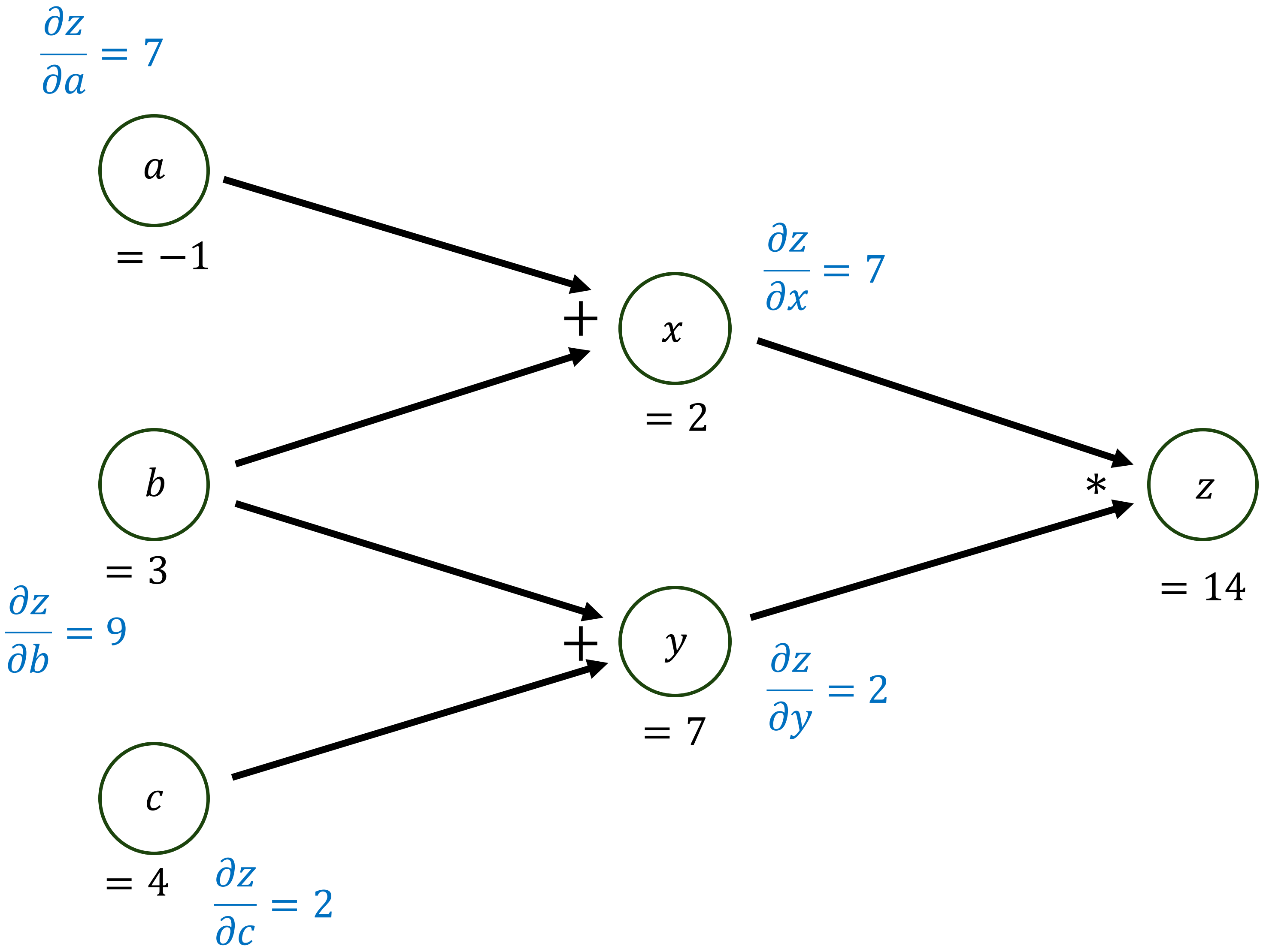


a
$= -1$
b
$= 3$
c
$= 4$
$+$
x
$= 2$
$\frac{\partial z}{\partial x} = 7$
$+$
y
$= 7$
$\frac{\partial z}{\partial y} = 2$
$*$
z
$= 14$

Backward Pass

$$z = \underbrace{(a+b)}_{x} * \underbrace{(b+c)}_{y}$$

$a = -1 \qquad b = 3 \qquad c = 4$



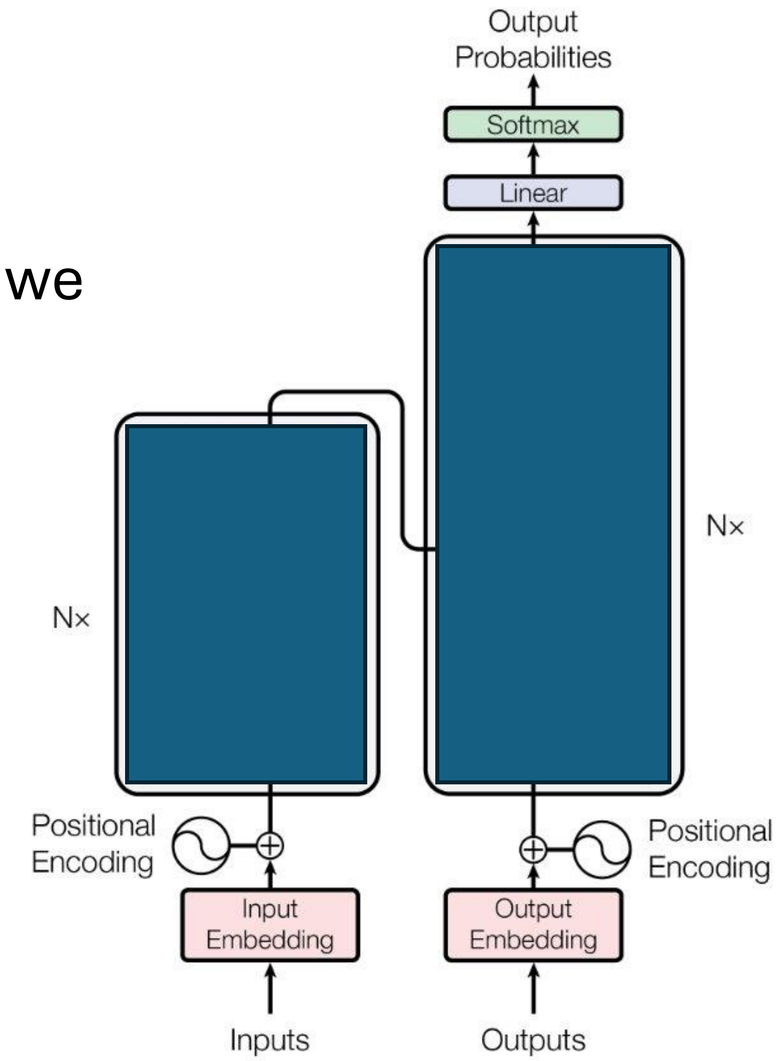
$\frac{\partial z}{\partial a} = 7$
a
$= -1$
b
$= 3$
$\frac{\partial z}{\partial b} = 9$
c
$= 4$
$\frac{\partial z}{\partial c} = 2$
$+$
x
$= 2$
$\frac{\partial z}{\partial x} = 7$
$+$
y
$= 7$
$\frac{\partial z}{\partial y} = 2$
$*$
z
$= 14$

Transformers

Where in the transformer do we use:

- Self-Attention
- Masked Self-Attention
- Cross-Attention
- Masked Cross-Attention

?

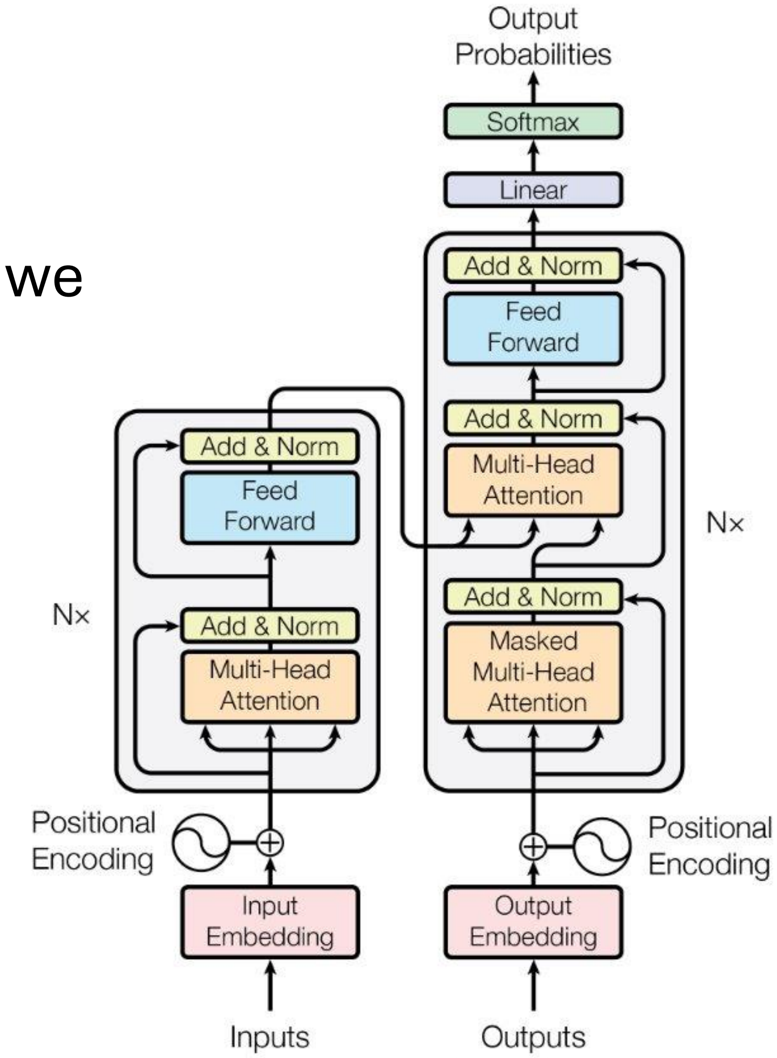


Transformers

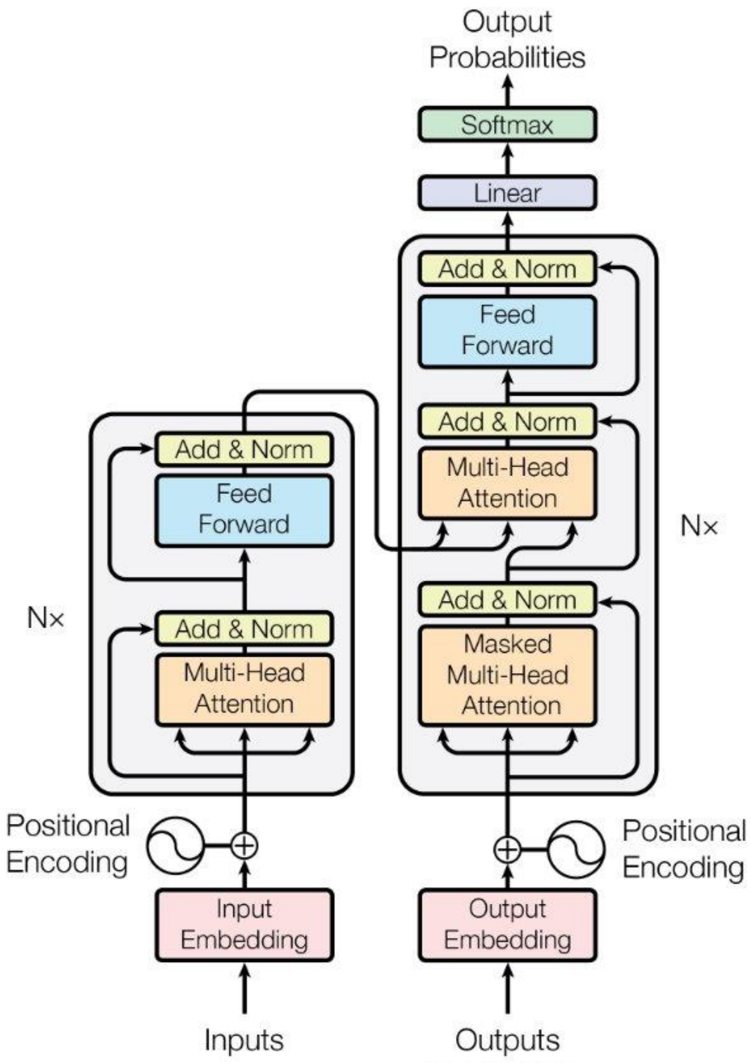
Where in the transformer do we use:

- Self-Attention
- Masked Self-Attention
- Cross-Attention
- Masked Cross-Attention

?



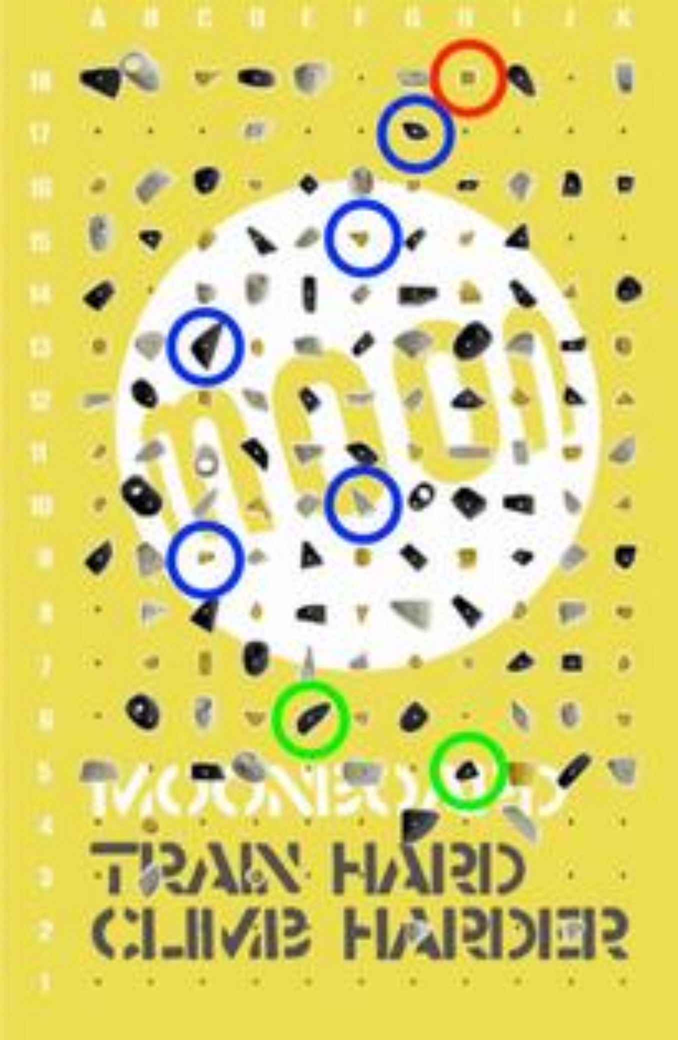
Transformers



Why are the outputs going into the model?
Is this different during training and testing?

Moonboard

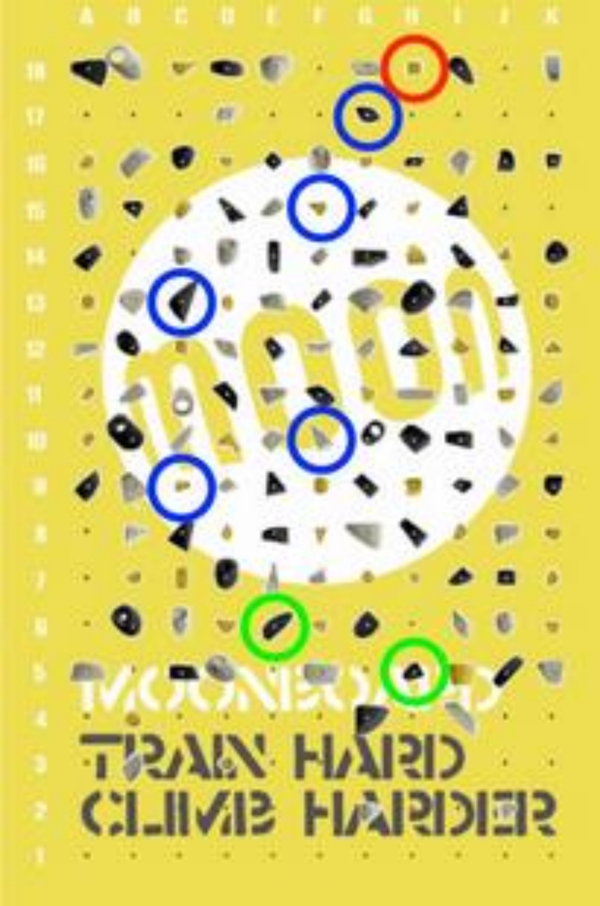
friction labs.com



[18H, 17G, 15F, 13C, 10F, 9C, 6E, 5H], Grade: 5

Grade: 1 -> easiest
Grade 16 -> hardest

Moonboard

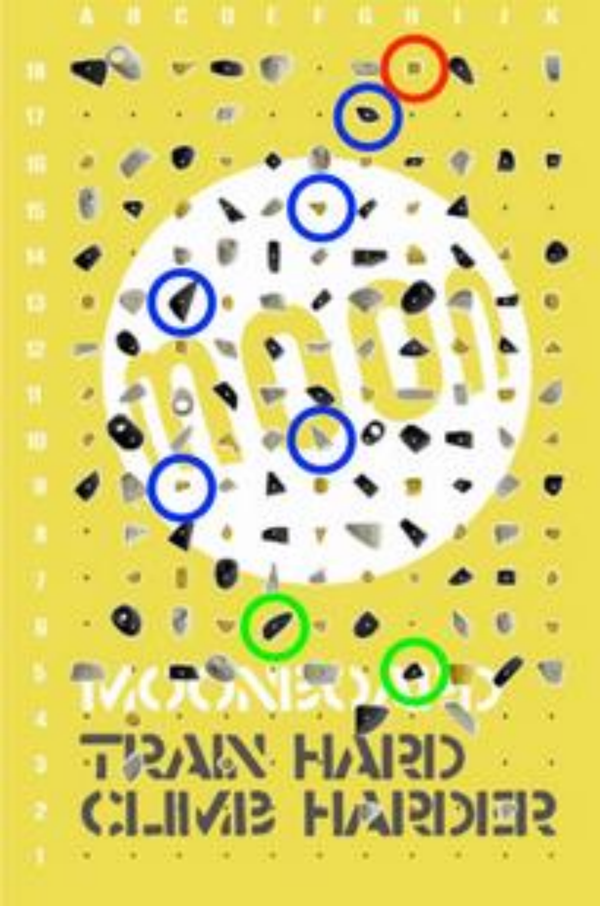


frictionlabs.com

Training Data:
[18H, 17G, 15F, 13C, 10F,
9C, 6E, 5H], Grade: 5
....

Design a neural network that takes a **route as input and outputs a grade.**

Moonboard



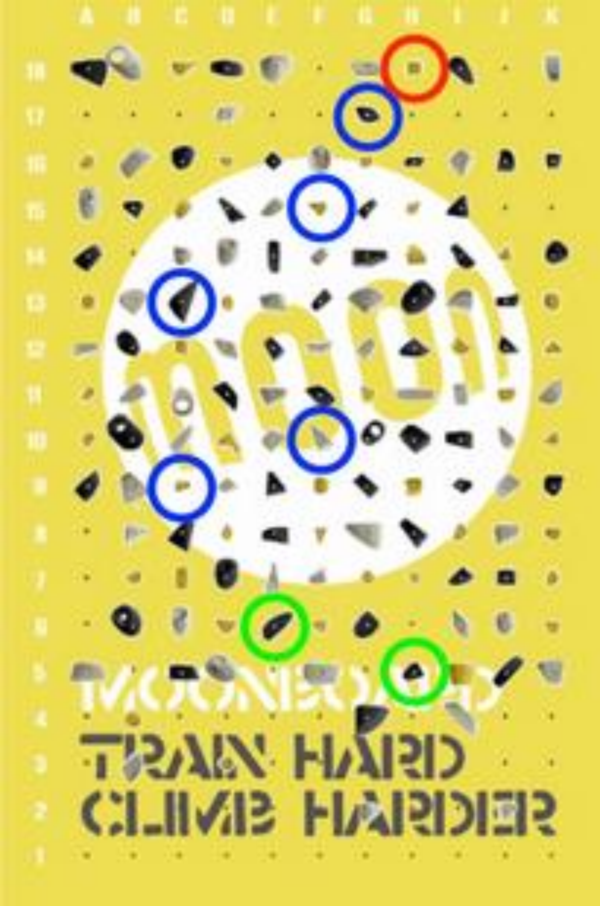
frictionlabs.com

Training Data:
[18H, 17G, 15F, 13C, 10F,
9C, 6E, 5H], Grade: 5
....

Design a neural network that takes a **route as input and outputs a grade.**

Which type of network should we use?
Which type of loss function?

Moonboard



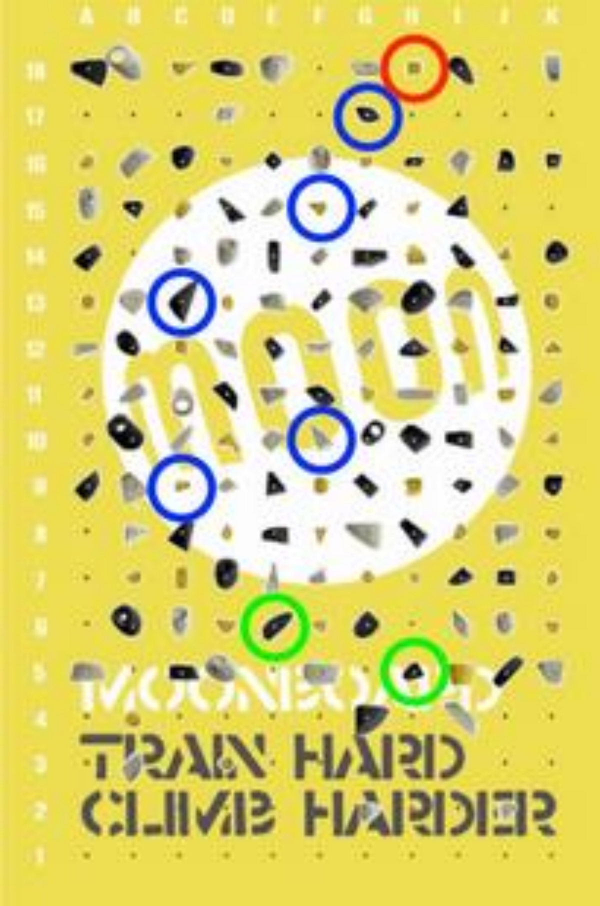
frictionlabs.com

Training Data:
[18H, 17G, 15F, 13C, 10F,
9C, 6E, 5H], Grade: 5
....

Design a neural network that takes a **route as input and outputs a grade.**

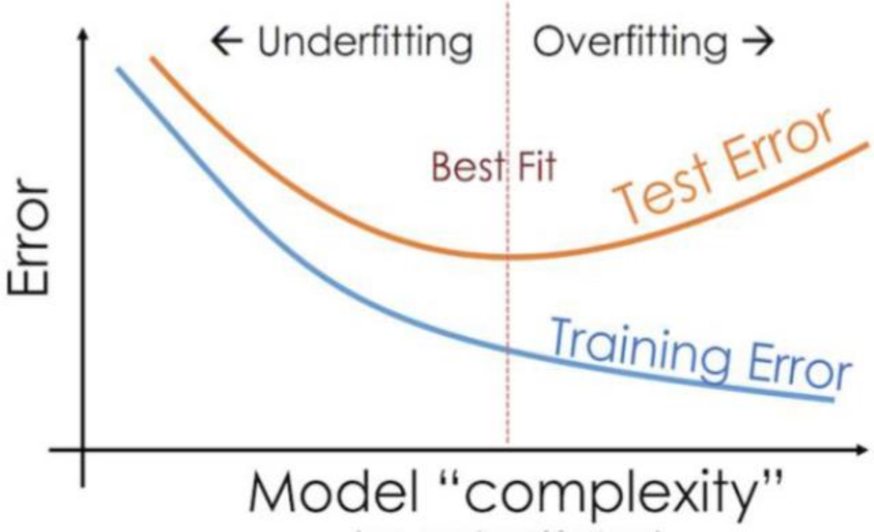
How can stop our network from overfitting?

Moonboard



frictionlabs.com

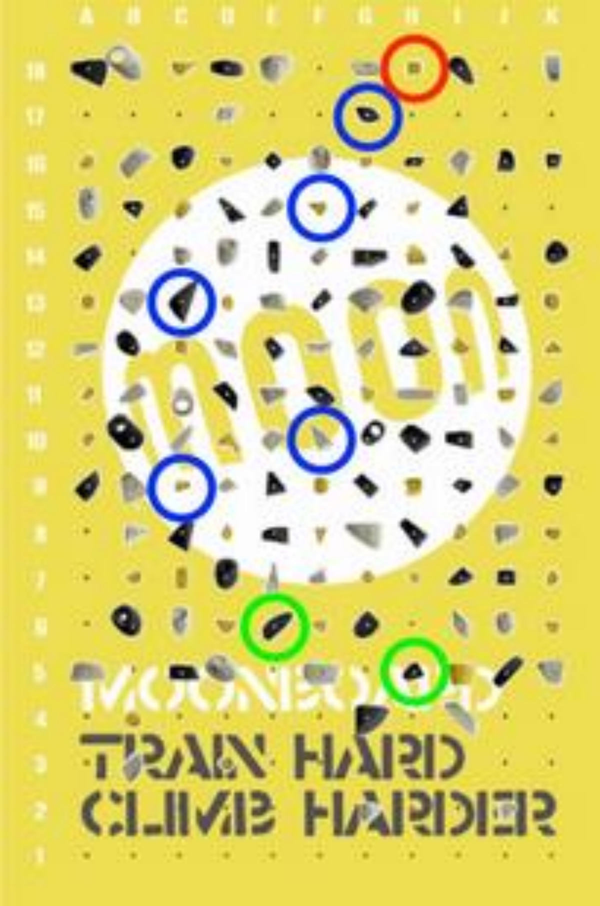
Training Data:
[18H, 17G, 15F, 13C, 10F, 9C, 6E, 5H], Grade: 5
....



Design a neural network that takes a **route as input and outputs a grade.**

How can stop our network from overfitting?

Moonboard



frictionlabs.com

Training Data:
[18H, 17G, 15F, 13C, 10F,
9C, 6E, 5H], Grade: 5
....

Design a neural network that takes a **grade as input and outputs a route.**