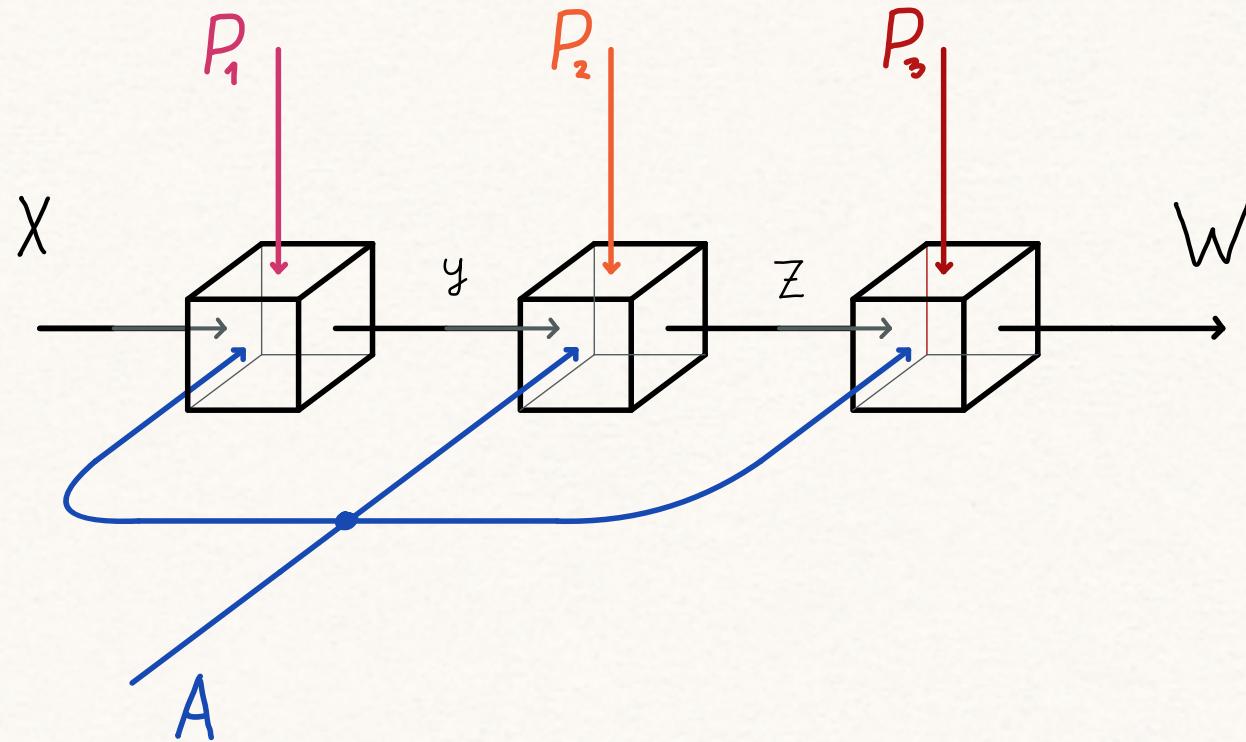


GCNNs AS PARAMETRIC COKLEISLI MORPHISMS

BRUNO GAVRANOVIC, MATTIA VILLANI



CHILL TALK?

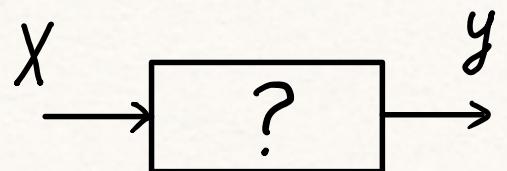
GOAL:

PROVIDE A CATEGORICAL FRAMEWORK

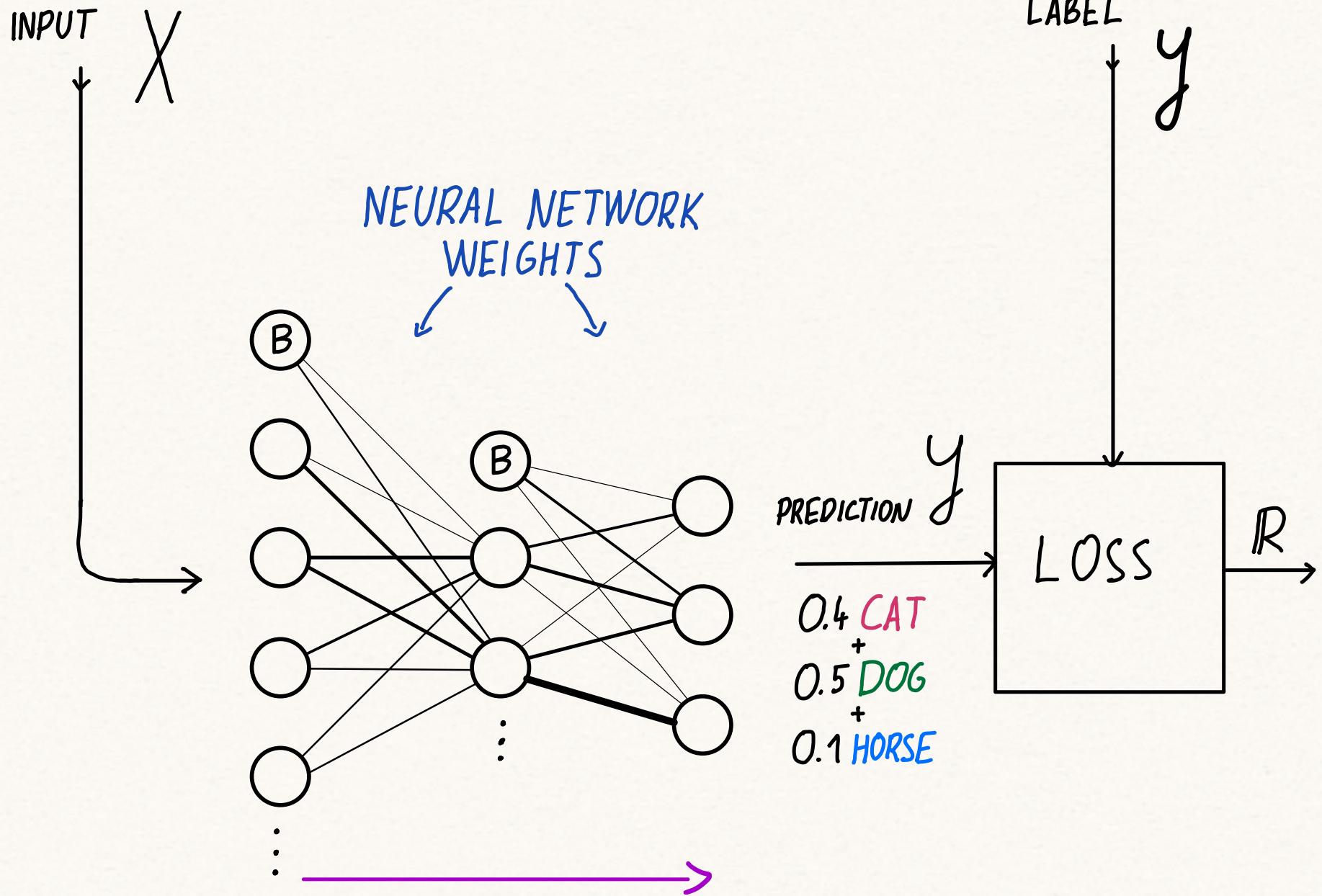
-

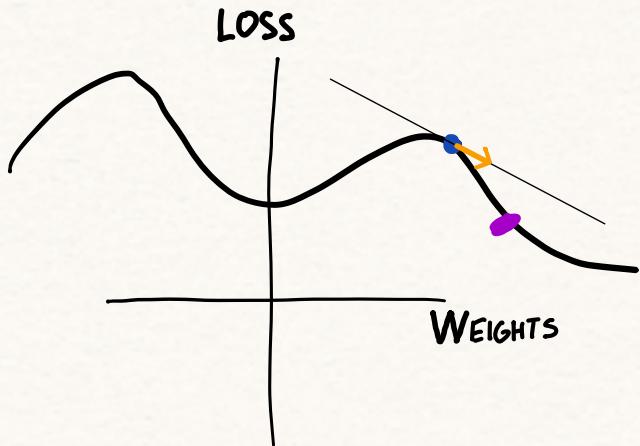
FOR DEEP LEARNING

SUPERVISED LEARNING WITH NEURAL NETWORKS IN ONE SLIDE:



TASK: FIND A FUNCTION $X \rightarrow y$ THAT BEST FITS
A DATASET: List $X \times y$





GRADIENT DESCENT

~

„OPTIMIZER”

- NN IS COMPUTATION PARAMETERIZED BY WEIGHTS
- BACKPROPAGATION OF CHANGES
- PARAMETER UPDATE - „OPTIMIZERS”

WE ALREADY HAVE A FRAMEWORK:

Categorical Foundations of Gradient-Based Learning

G.S.H. CRUTTWELL, Mount Allison University, Canada

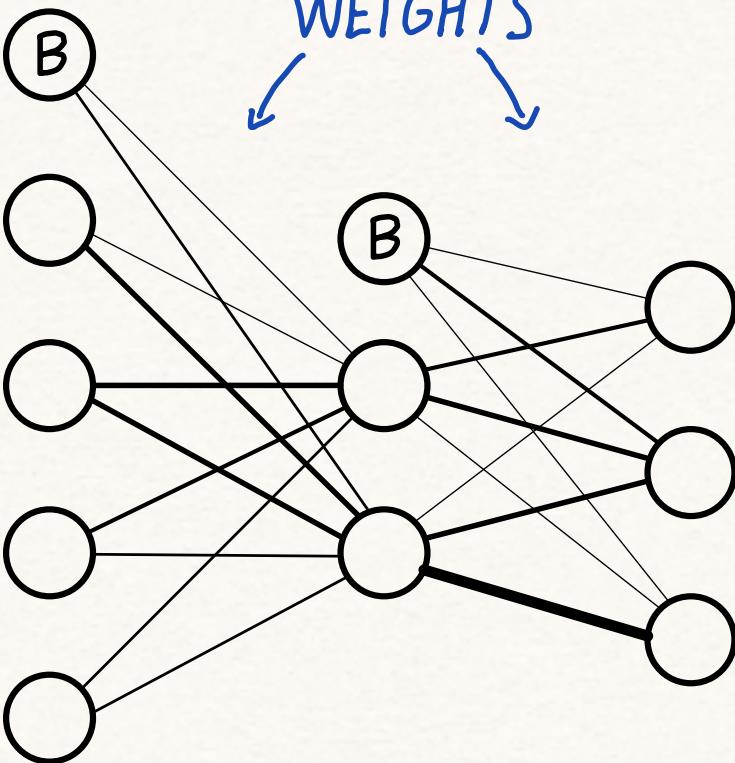
BRUNO GAVRANOVIC and NEIL GHANI, University of Strathclyde, UK

PAUL WILSON and FABIO ZANASI, University College London, UK

We propose a categorical semantics of gradient-based machine learning algorithms in terms of lenses, parametrised maps, and reverse derivative categories. This foundation provides a powerful explanatory and unifying framework: it encompasses a variety of gradient descent algorithms such as ADAM, AdaGrad, and Nesterov momentum, as well as a variety of loss functions such as MSE and Softmax cross-entropy, shedding new light on their similarities and differences. Our approach to gradient-based learning has examples generalising beyond the familiar continuous domains (modelled in categories of smooth maps) and can be realized in the discrete setting of boolean circuits. Finally, we demonstrate the practical significance of our framework with an implementation in Python.

PARA CONSTRUCTION

NEURAL NETWORK
WEIGHTS



BASIC NN LAYER

$$f: \mathbb{R}^k \rightarrow \mathbb{R}^{k'} \\ f(X) = \sigma(XW)$$

$$\begin{matrix} 1 & X &] \\ & k & \\ & & k' \end{matrix} \quad \begin{matrix} W \\ [\end{matrix}$$

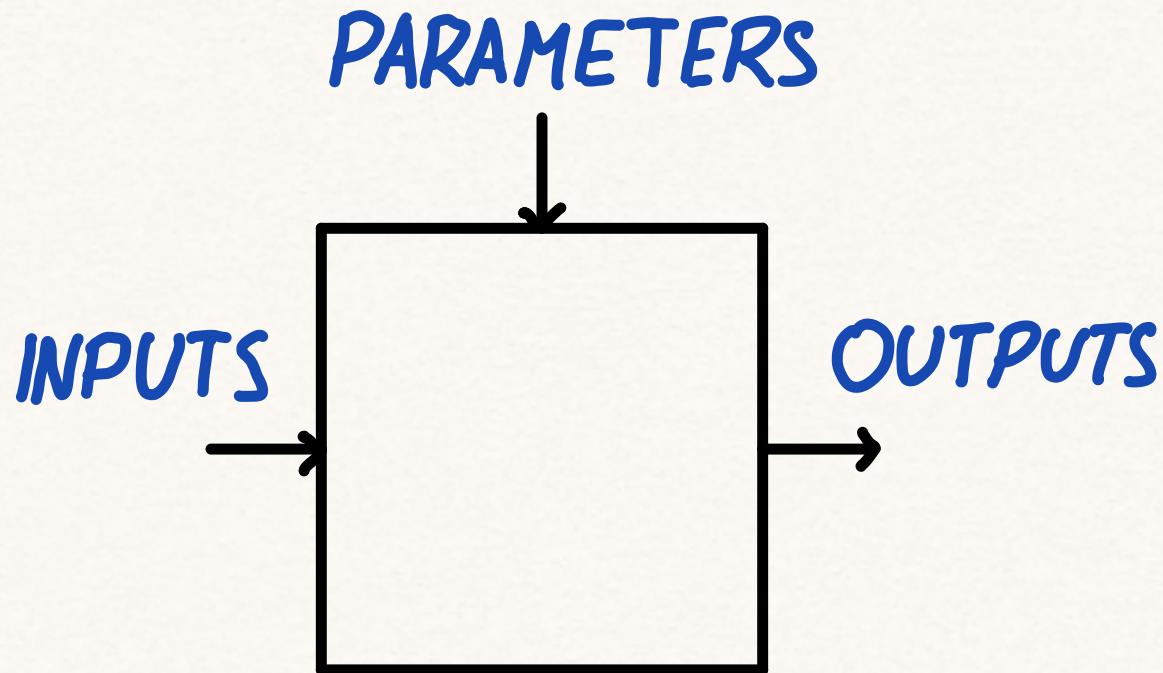
k - NUMBER OF INCOMING FEATURES
 k' - NUMBER OF OUTGOING FEATURES

EACH LAYER HAS ITS OWN WEIGHT MATRIX

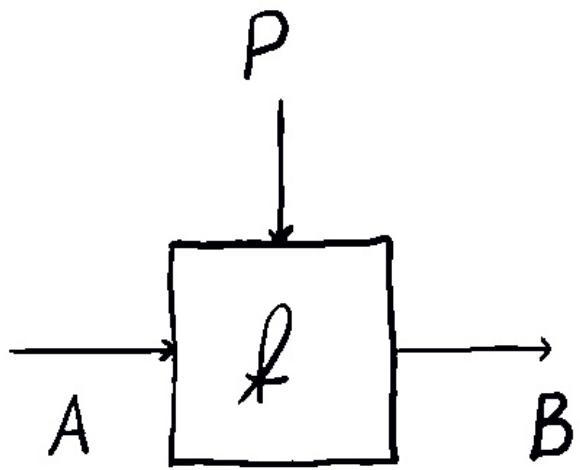
FIX A MONOIDAL CATEGORY $(\mathcal{C}, \otimes, I)$.

$\text{Para}(\mathcal{C})$ IS A BICATEGORY WHERE

$$\text{Para}(\mathcal{C})(A, B) := \sum_{P : \mathcal{C}} \mathcal{C}(P \otimes A, B)$$



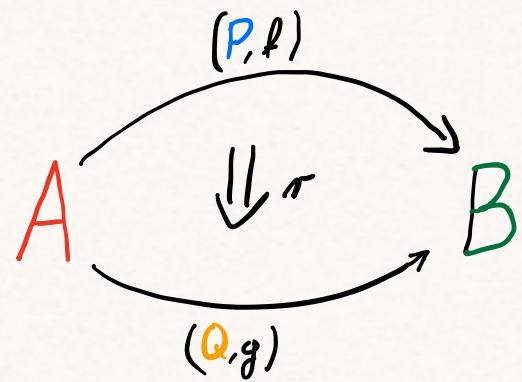
COMPOSITION TENSORS THE PARAMETERS



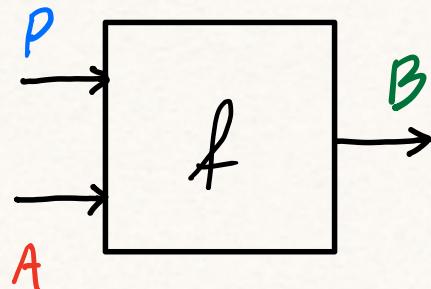
GRAPHICAL LANGUAGE

TEXTUAL
NOTATION

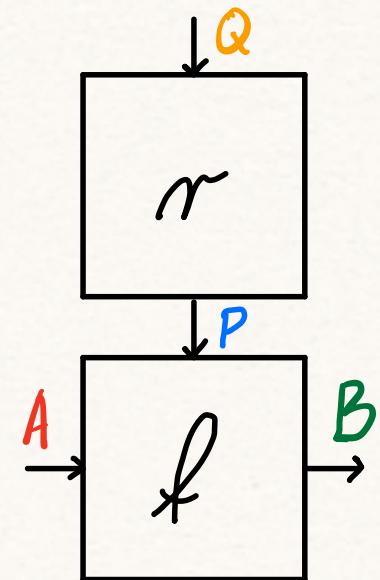
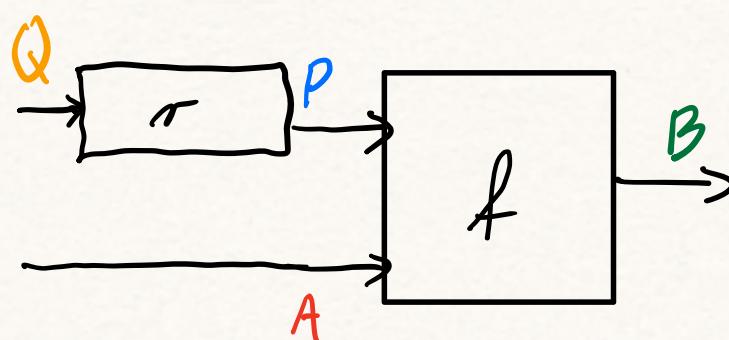
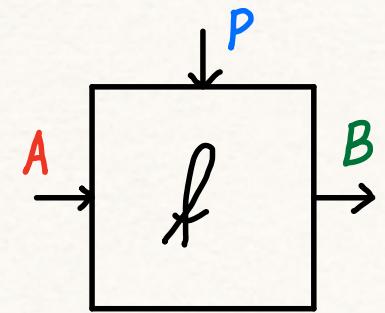
$$f: P \otimes A \longrightarrow B$$



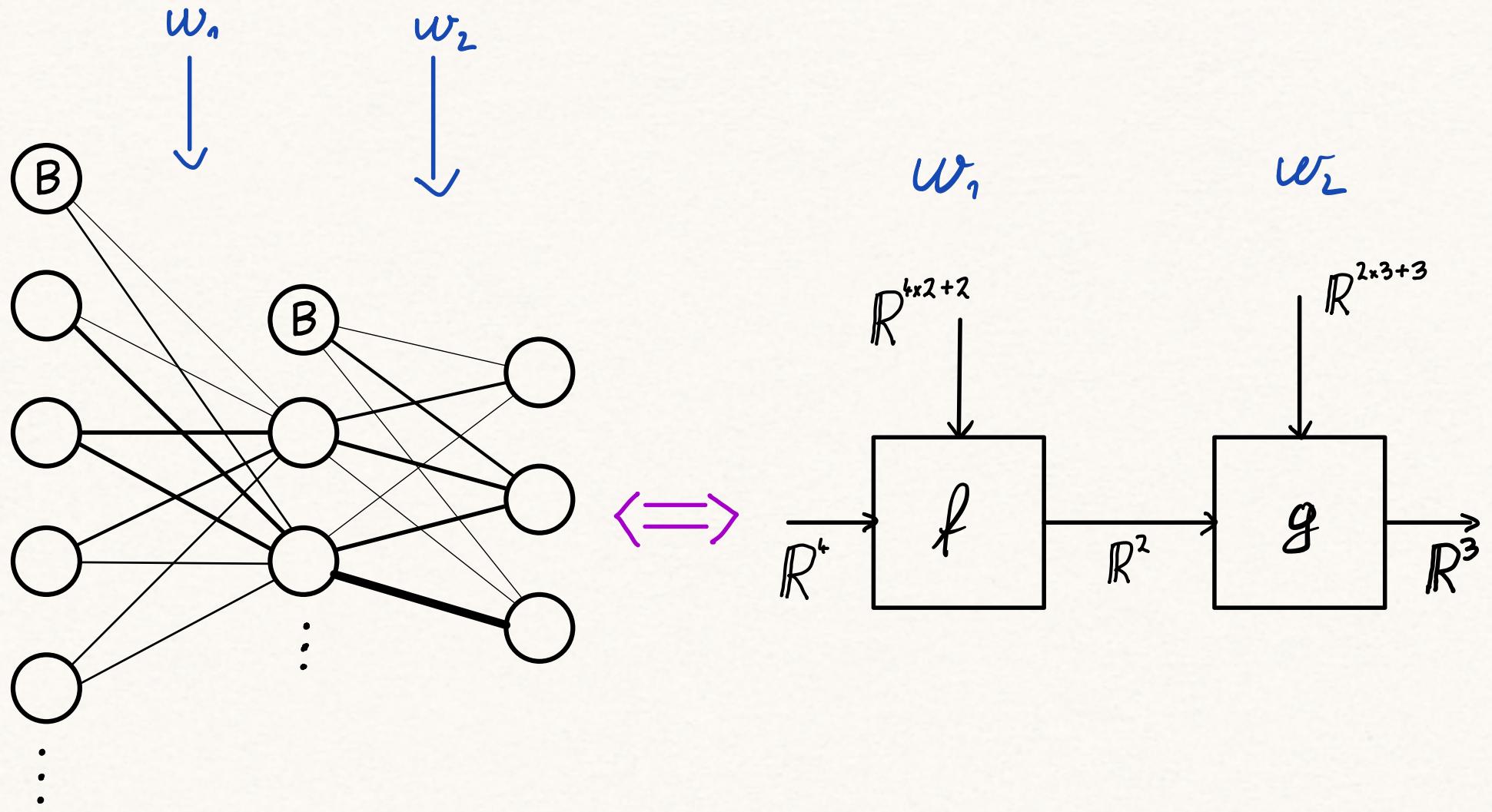
STANDARD
STRING DIAGRAM



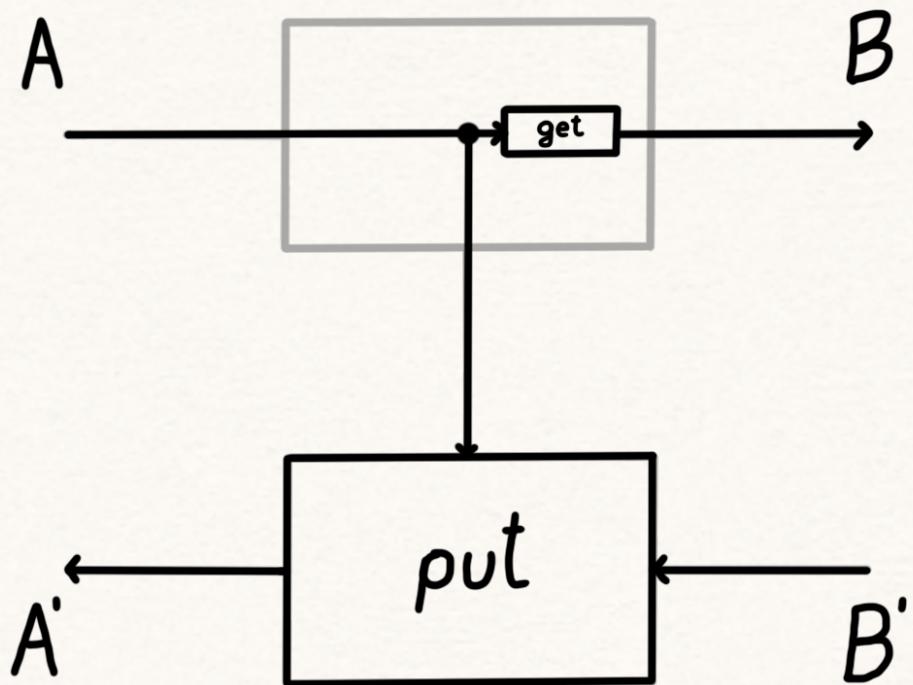
'2D'
STRING DIAGRAM



EXAMPLE

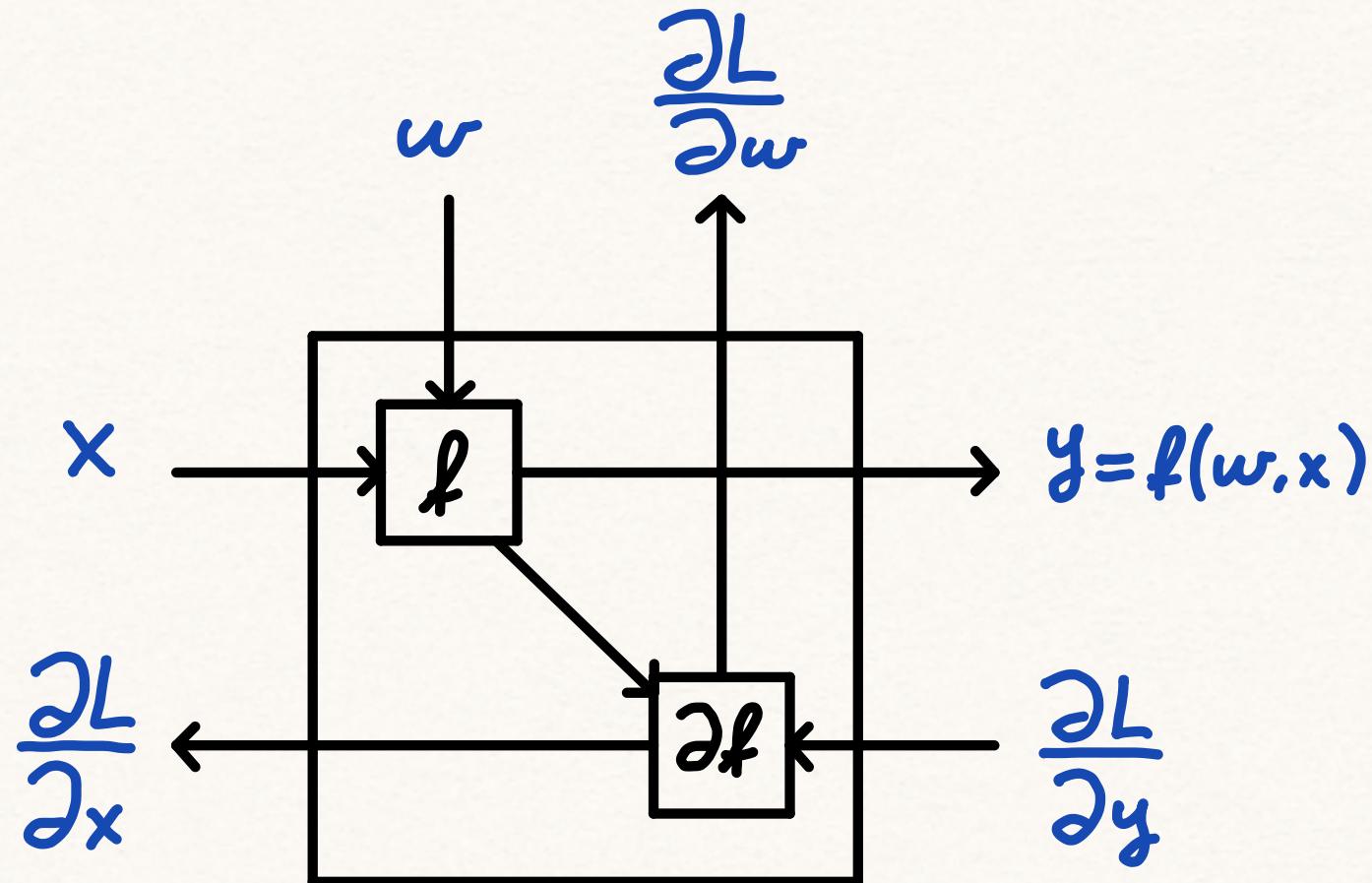


LENS



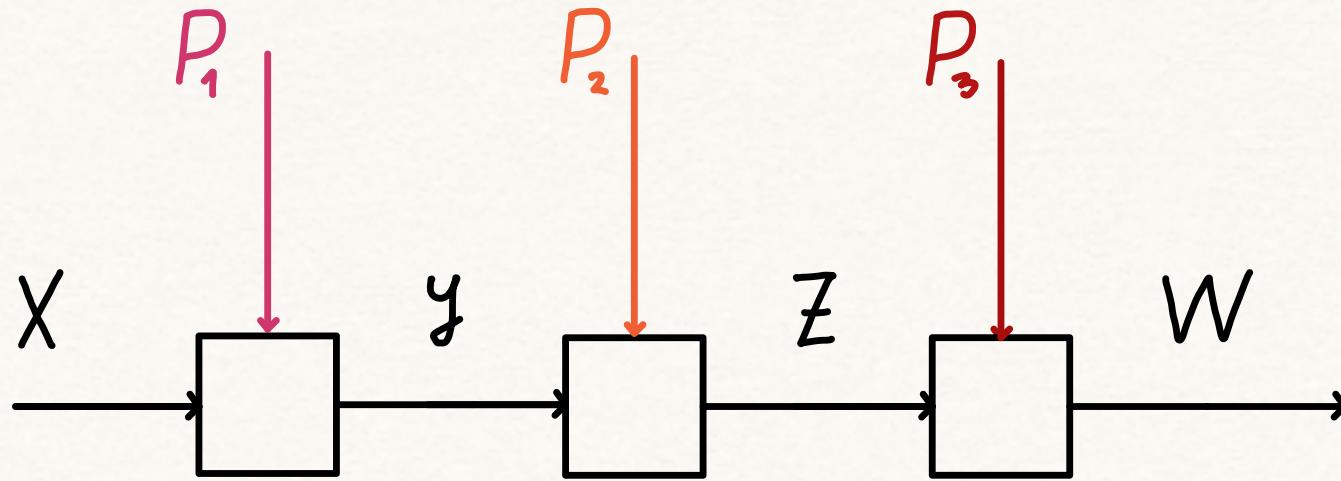
$$e \xrightarrow{R} \text{Lens}_A(e)$$

PARAMETRIC LENS



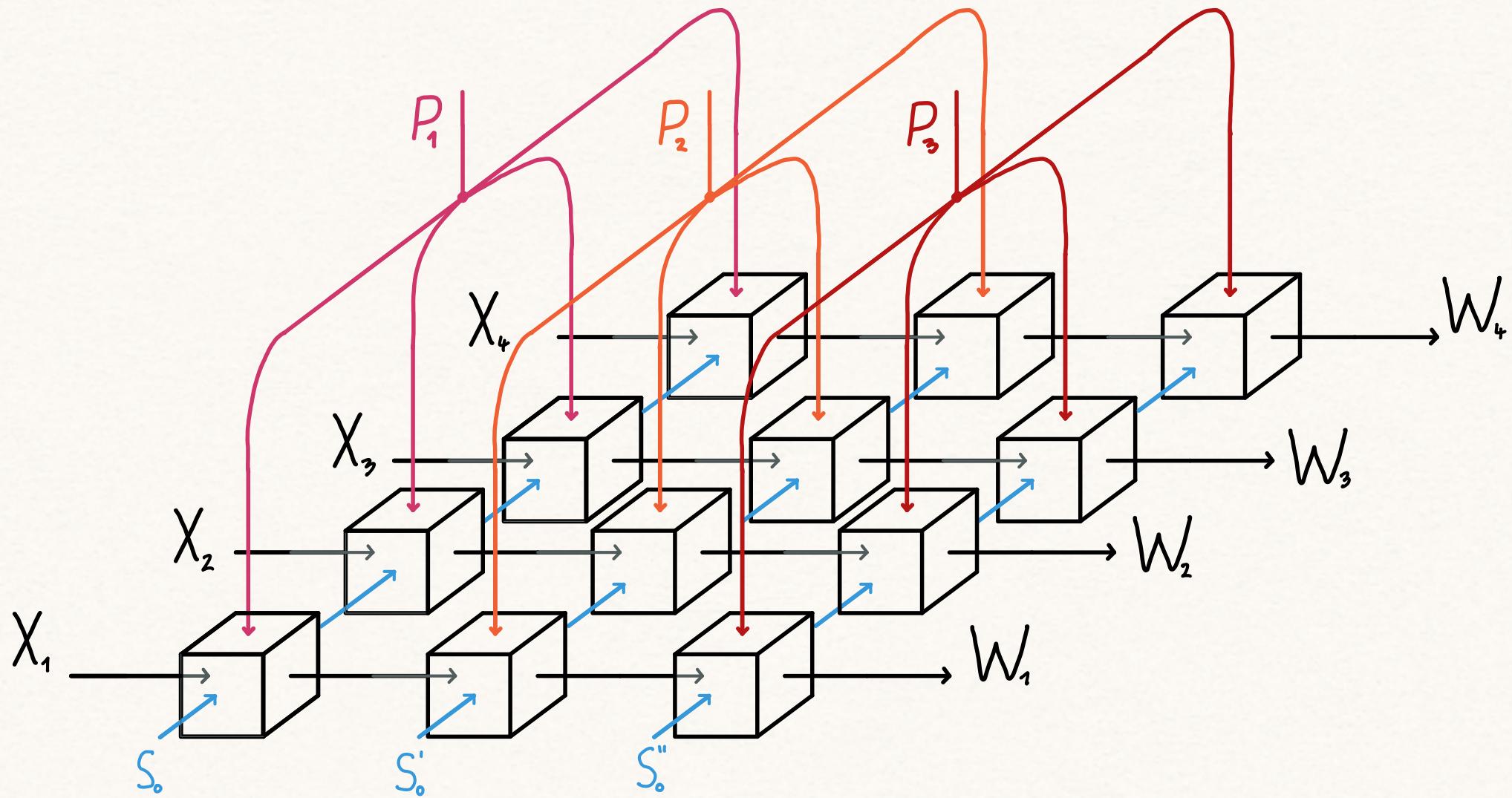
$$\text{Para}(e) \xrightarrow{\text{Para}(R)} \text{Para}(\text{Lens}(e))$$

- FRAMEWORK OF PARAMETRIC LENSES IS INCREDIBLY FLEXIBLE

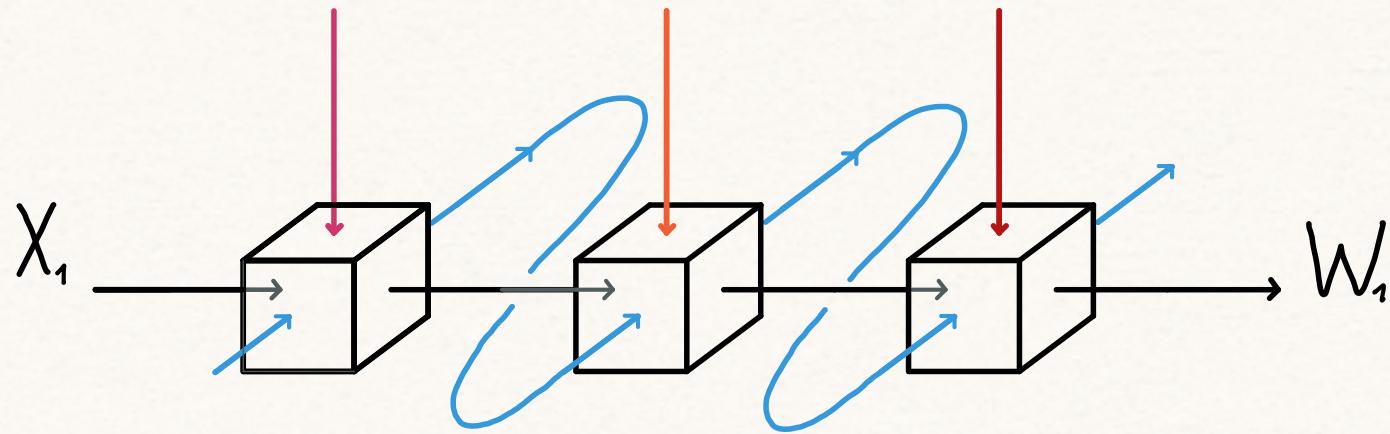


- IT DOESN'T MAKE ANY ASSUMPTIONS ABOUT THE ARCHITECTURE, THUS MODELLING...

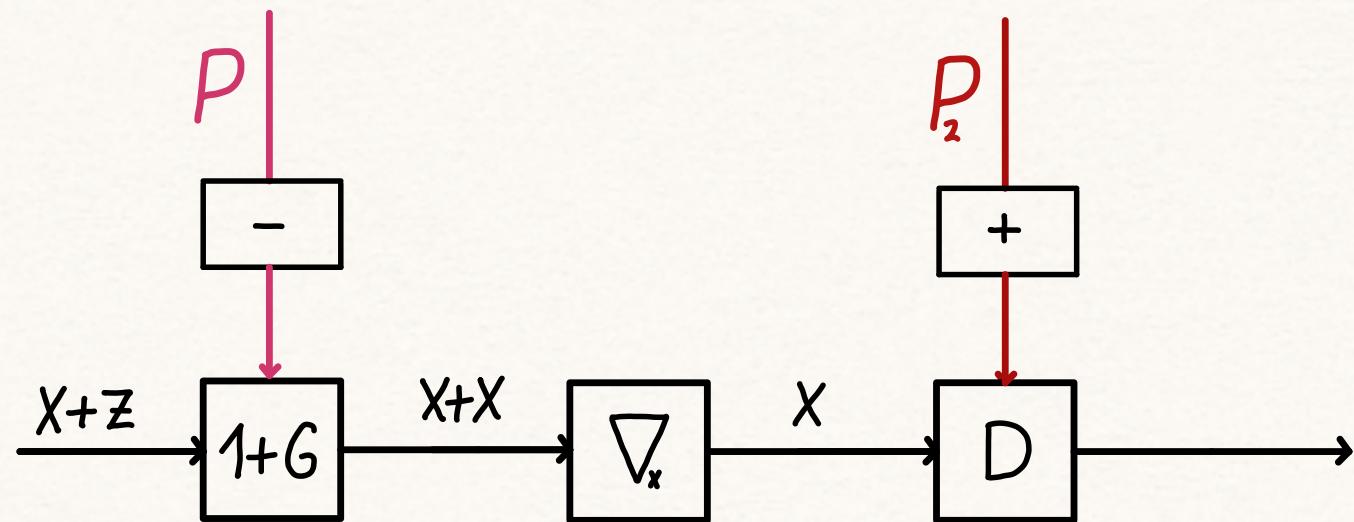
RECURRENT NEURAL NETWORKS



AUTOREGRESSIVE NEURAL NETWORKS



GENERATIVE ADVERSARIAL NETWORKS



- CONVOLUTIONAL NEURAL NETWORKS
- RECURSIVE NEURAL NETWORKS
- GRAPH NEURAL NETWORKS

• • •

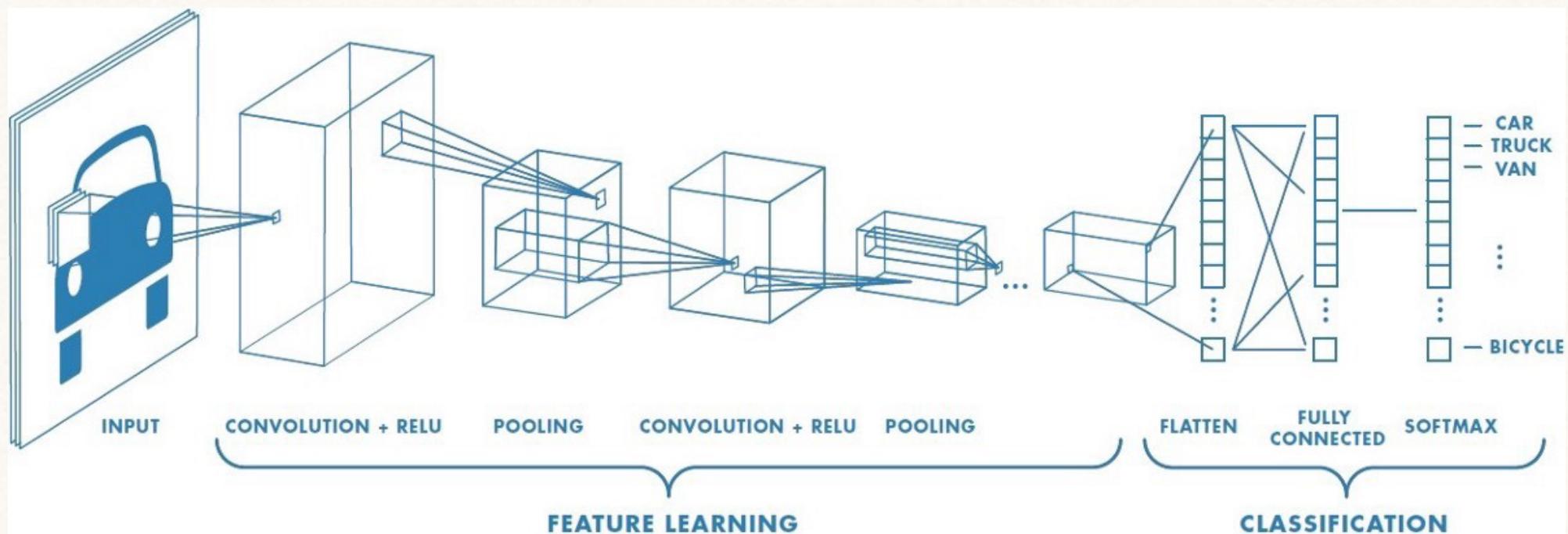
ALL OF THESE ARE JUST PARAMETRIC LENSES!

HOW CAN WE MAKE THIS STRUCTURE
VISIBLE IN CATEGORY THEORY?

- AN ENORMOUS ENDEAVOR!
- STARTING WITH GCNNs

CONVOLUTIONAL NEURAL NETWORKS

- COMMONLY APPLIED TO PROCESSING IMAGE DATA



1 <small>$\times 1$</small>	1 <small>$\times 0$</small>	1 <small>$\times 1$</small>	0	0
0 <small>$\times 0$</small>	1 <small>$\times 1$</small>	1 <small>$\times 0$</small>	1	0
0 <small>$\times 1$</small>	0 <small>$\times 0$</small>	1 <small>$\times 1$</small>	1	1
0	0	1	1	0
0	1	1	0	0

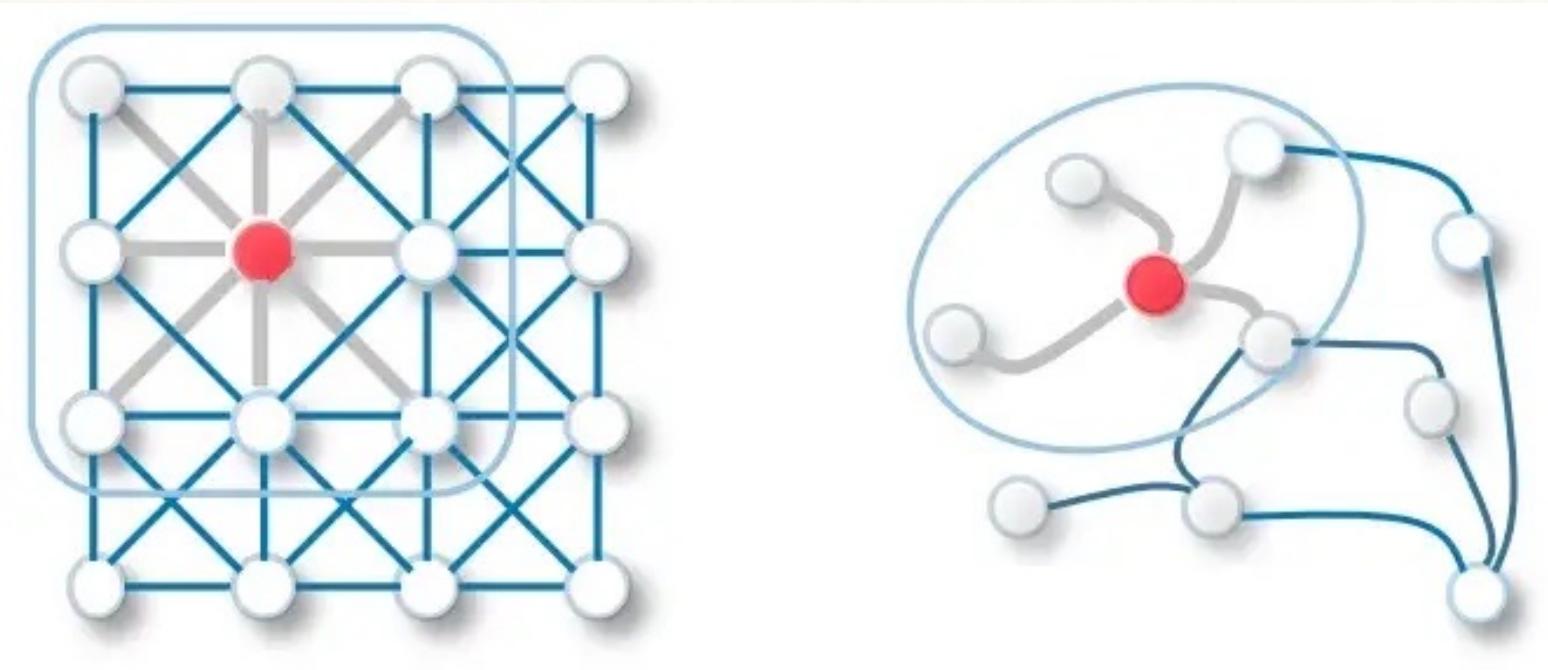
Image

4		

Convolved
Feature

GRAPH CONVOLUTIONAL NEURAL NETWORKS IN ONE SLIDE:

- THEY GENERALISE CONVOLUTIONAL NN's ...



... TO WORK FOR AN ARBITRARY GRAPH.

•IDEA: THINK ABOUT EACH NODE AS RECEIVING MESSAGES FROM ITS NEIGHBOURS!

1	1	1	0	0
0	1 _{x1}	1 _{x0}	1 _{x1}	0
0	0 _{x0}	1 _{x1}	1 _{x0}	1
0	0 _{x1}	1 _{x0}	1 _{x1}	0
0	1	1	0	0

Image

4	3	4
2	4	

Convolved Feature

'PROCESS MANY KINDS OF DATAPoints AT ONCE'

BASIC GCNN LAYER

$$f: \mathbb{R}^{n \times k} \longrightarrow \mathbb{R}^{n \times k'}$$
$$f(X) = \sigma(AXW)$$

$$n \begin{bmatrix} A \\ \vdots \\ A \end{bmatrix}^n \begin{bmatrix} X \\ \vdots \\ X \end{bmatrix}^k \begin{bmatrix} W \\ \vdots \\ W \end{bmatrix}^{k'}$$

n - NUMBER OF NODES IN A GRAPH

k - NUMBER OF INCOMING FEATURES

k' - NUMBER OF OUTGOING FEATURES

EACH LAYER HAS ITS OWN WEIGHT MATRIX

EACH LAYER SHARES THE ADJACENCY MATRIX

WE NEED SOMETHING DIFFERENT

THE COREADER COMONAD

LET \mathcal{C} BE A CARTESIAN CATEGORY.

FIX $A \in \mathcal{C}$.

$$\mathcal{C} \xrightarrow{A \times -} \mathcal{C}$$

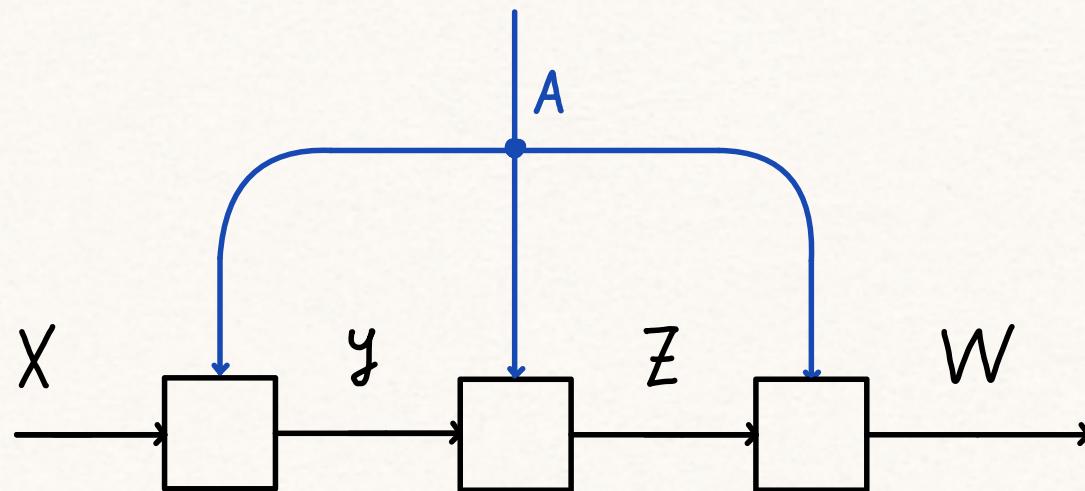
$$S_x: A \times X \rightarrow A \times A \times X$$

$$\varepsilon_x: A \times X \rightarrow X$$

$\text{Cokl}(A \times -)$

- CATEGORY WITH THE SAME OBJECTS AS \mathcal{C}
- $\text{Cokl}(A \times -)(X, Y) := \mathcal{C}(A \times X, Y)$

COMPOSITION SHARES A



Cokl(Ax-)

GLOBAL

CATEGORY

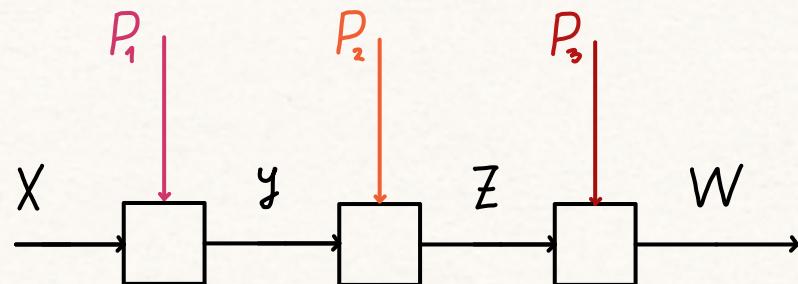
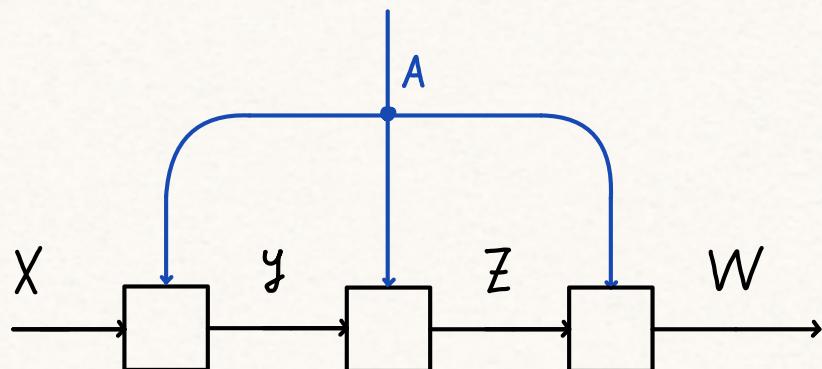
COMONAD

Para (e)

LOCAL

BICATEGORY

GRADED COMONAD



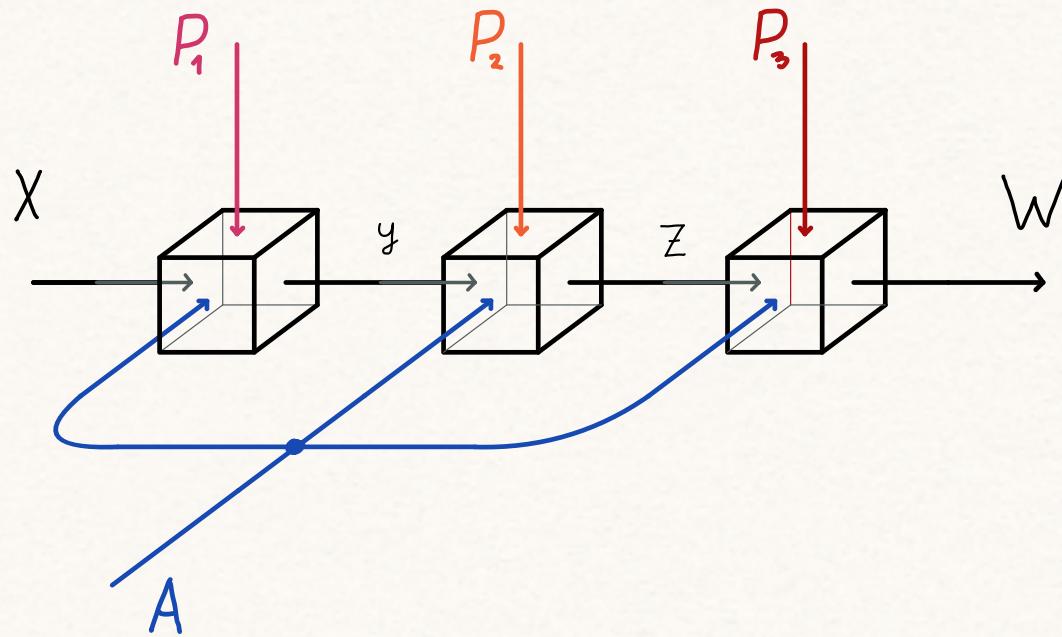
„Para IS THE LOCAL VERSION OF Cokl(- x=)“

CAN WE USE $\text{CoKL}(A_{x^-})$ AS THE BASE CATEGORY FOR LEARNING?

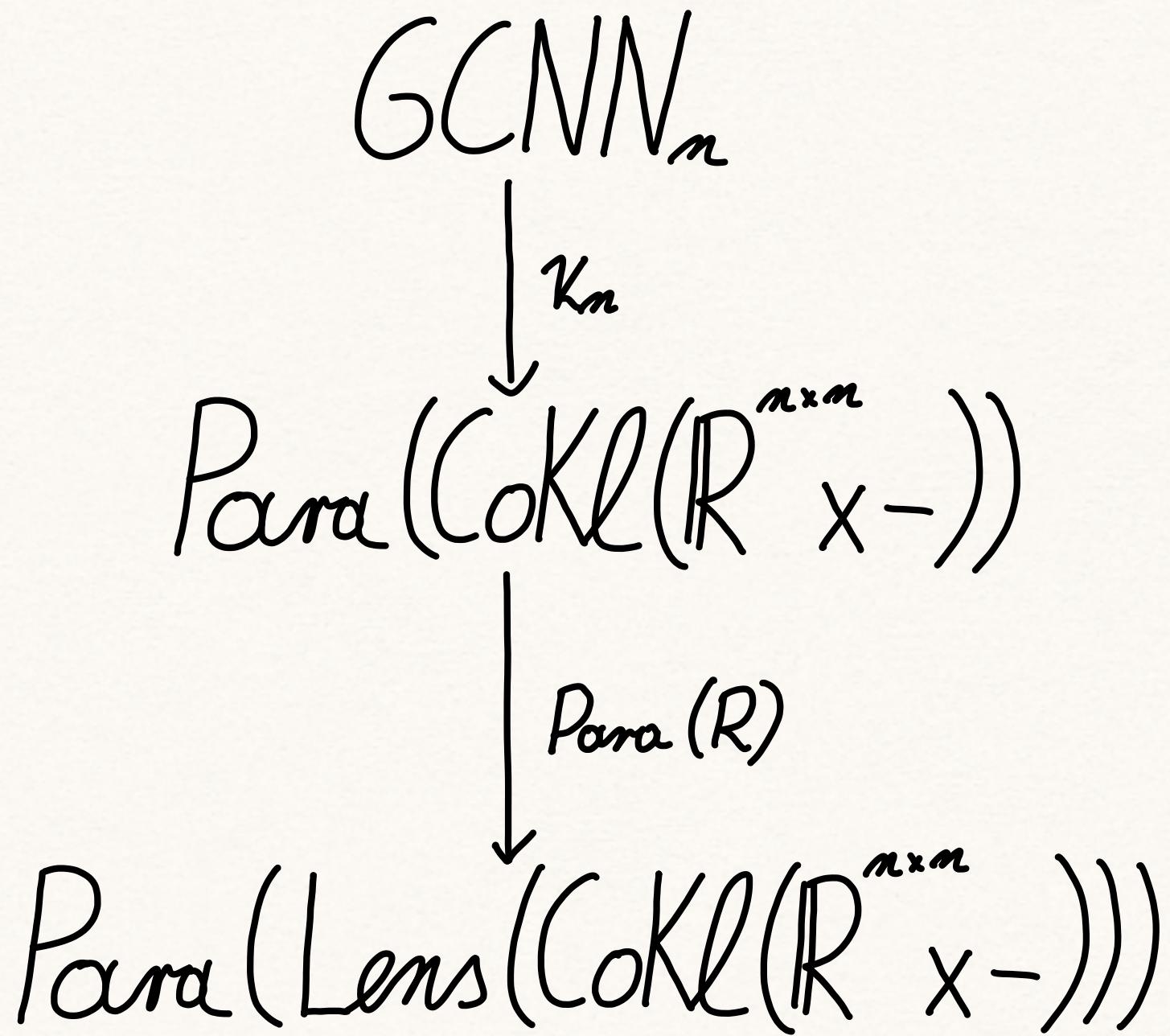
- $\text{Para}(\text{CoKL}(R^{n \times n}))$?
- $\text{CoKL}(R^{n \times n}_{x^-}) \rightarrow \text{Lens}(\text{CoKL}(R^{n \times n}_{x^-}))$?

YES!

- $\text{CoKL}(A_{x-})$ IS A ℓ -ACTEGORY
- $\text{CoKL}(A_{x-}) \rightarrow \text{Lens}_A(\text{CoKL}(A_{x-}))$ EXISTS WHEN
 $\ell \rightarrow \text{Lens}_A(\ell)$ DOES



$$GCNN_n \xrightarrow{\kappa_n} \text{Para}(\text{CoKl}(R^{n \times n} \times -))$$



FUTURE WORK

- REFINE THIS TO COVER GENERAL GNNs?
- GENERAL THEORY OF ARCHITECTURES?
- USE (CO)ALGEBRAS TO DESCRIBE THEIR OFTEN STRUCTURALLY RECURSIVE NATURE?