

# Towards a Categorical Foundation of Deep Learning: A Survey

Una rassegna di approcci categorici al *deep learning*

---

Francesco Riccardo Crescenzi

12 settembre 2024

Alma mater studiorum - Università di Bologna  
CdL in Matematica

**We are in an AI summer,  
but is winter coming?**

Mancano fondamenta teoriche:

- **design ad hoc** (Gavranović, 2024)

Mancano fondamenta teoriche:

- **design ad hoc** (Gavranović, 2024)
- **complessità fine a se stessa** (Rahimi, 2017)

Mancano fondamenta teoriche:

- **design ad hoc** (Gavranović, 2024)
- **complessità fine a se stessa** (Rahimi, 2017)
- **fragilità** (Gavranović, 2024)

La ricerca viene rallentata da:

- **research debt** (Olah e Carter, 2017)

La ricerca viene rallentata da:

- **research debt** (Olah e Carter, 2017)
- **mancata replicabilità** (Raff, 2019)

# Teoria delle categorie:

una lingua franca della matematica



La teoria delle categorie studia strutture e relazioni, e può essere vista come un'estensione del celebre *Erlangen Programme*.

# Teoria delle categorie:

una lingua franca delle scienze

La teoria delle categorie può essere applicata con successo anche in fisica, informatica, chimica... ovunque ci sia **composizionalità** (Fong e Spivak, 2018).

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)
- **integral transforms** (Dudzik e Veličković, 2022, Dudzik et al., 2024)

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)
- **integral transforms** (Dudzik e Veličković, 2022, Dudzik et al., 2024)
- **functor learning** (Gavranović, 2019, Sheshmani e You, 2021, Chytas et al., 2024)

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)
- **integral transforms** (Dudzik e Veličković, 2022, Dudzik et al., 2024)
- **functor learning** (Gavranović, 2019, Sheshmani e You, 2021, Chytas et al., 2024)
- **compositional distributional model of meaning** (Clark e Pulman, 2007, Coecke et al., 2010, Lewis, 2019)



- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)
- **integral transforms** (Dudzik e Veličković, 2022, Dudzik et al., 2024)
- **functor learning** (Gavranović, 2019, Sheshmani e You, 2021, Chytas et al., 2024)
- **compositional distributional model of meaning** (Clark e Pulman, 2007, Coecke et al., 2010, Lewis, 2019)
- **neural circuit diagrams** (Abbott, 2023)

- **ottiche parametriche** (Gavranović, 2024, Cruttwell et al., 2022)
- **(co)algebre categoriche** (Gavranović et al., 2024)
- **integral transforms** (Dudzik e Veličković, 2022, Dudzik et al., 2024)
- **functor learning** (Gavranović, 2019, Sheshmani e You, 2021, Chytas et al., 2024)
- **compositional distributional model of meaning** (Clark e Pulman, 2007, Coecke et al., 2010, Lewis, 2019)
- **neural circuit diagrams** (Abbott, 2023)
- **string diagrams with universal approximators** (Khatri et al., 2024)

# Lenti parametriche

per modellare il gradient-based learning

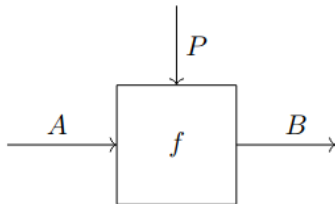
## DEFINIZIONE: Il costrutto Para

Sia  $(\mathcal{C}, I, \otimes)$  una categoria monoidale strettamente simmetrica. Allora,  $\mathbf{Para}_{\otimes}(\mathcal{C})$  è la 2-categoria definita come segue.

- Le 0-celle sono oggetti di  $\mathcal{C}$ .
- Le 1-cells sono coppie  $(P, f) : A \rightarrow B$ , dove  $P : \mathcal{C}$  e  $f : P \otimes A \rightarrow B$ .
- The 2-celle sono  $r : (P, f) \Rightarrow (Q, g)$ , dove  $r : P \rightarrow Q$  è un morfismo in  $\mathcal{C}$  che rispetta certe condizioni di naturalità.

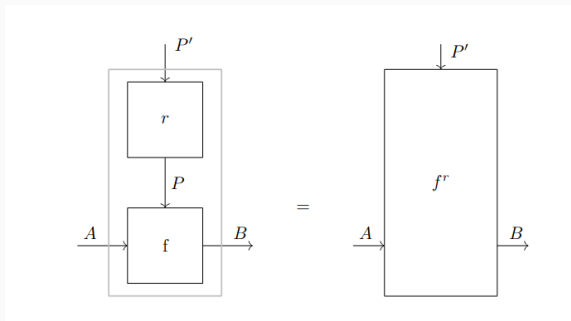
Vedasi Gavranović, 2024.

# Gradient-based learning con lenti parametriche



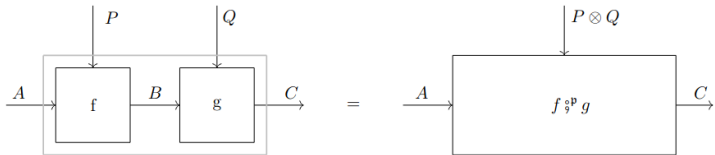
Gavranović, 2024

# Gradient-based learning con lenti parametriche



Gavranović, 2024

# Gradient-based learning con lenti parametriche



Gavranović, 2024

## DEFINIZIONE: Il costrutto **Lens**

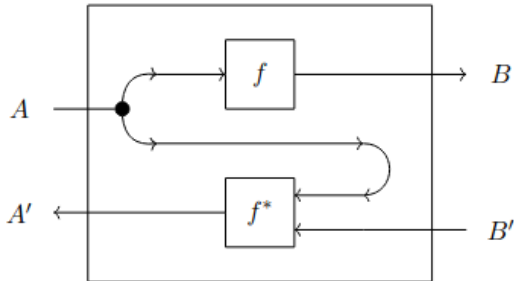
Sia  $(\mathcal{C}, 1, \times)$  una categoria Cartesiana. Allora, **Lens** $(\mathcal{C})$  è la categoria definita come segue.

- Un oggetto di **Lens** $(\mathcal{C})$  è una coppia  $\left( \begin{smallmatrix} A \\ A' \end{smallmatrix} \right)$  di oggetti di  $\mathcal{C}$ .
- Un morfismo  $\left( \begin{smallmatrix} A \\ A' \end{smallmatrix} \right) \rightarrow \left( \begin{smallmatrix} B \\ B' \end{smallmatrix} \right)$  (anche chiamato lente) è una coppia  $\left( \begin{smallmatrix} f \\ f' \end{smallmatrix} \right)$  di morfismi di  $\mathcal{C}$  tali che  $f : A \rightarrow B$  and  $f' : A \times B' \rightarrow A'$ . La mappa  $f$  è nota come *forward pass* della lente, mentre la mappa  $f'$  è nota come *backward pass*.

Vedasi Cruttwell et al., 2022.

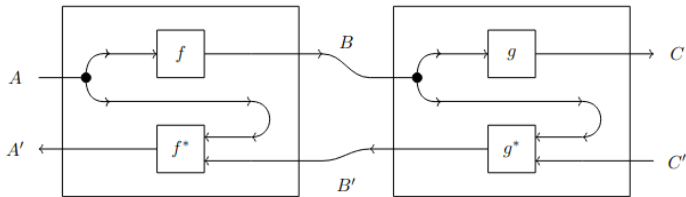


## Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

# Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

## DEFINIZIONE: Cartesian reverse differential category

Una *Cartesian reverse differential category* (CRDC)  $\mathcal{C}$  è una categoria Cartesiana con una struttura additiva dove è definito un operatore differenziale  $R$  che ha le proprietà di una *reverse derivative*. Si vedano Cockett et al., 2019 e Gavranović, 2024.

## ESEMPIO: Smooth

Consideriamo **Smooth**, ovvero la categoria degli spazi Euclidei e delle funzioni lisce. **Smooth** è una CRDC rispetto all'operatore

$$R[f] : (x, y) \mapsto \mathcal{J}_f(x)^T y.$$

## DEFINIZIONE: Lenti con backward pass additivo

Sia  $\mathcal{C}$  una CRDC. Allora, definiamo la sottocategoria  $\mathbf{Lens}_A(\mathcal{C})$  di  $\mathbf{Lens}_A(\mathcal{C})$ , i cui oggetti sono coppie  $\left(\begin{smallmatrix} A \\ A' \end{smallmatrix}\right)$  e i cui morfismi hanno la forma  $\left(\begin{smallmatrix} f \\ \mathbf{R}[f] \end{smallmatrix}\right)$ . Vedasi Cruttwell et al., 2022.

## TEOREMA: Struttura cartesiana di $\mathbf{Lens}_A(\mathcal{C})$

La struttura

$$I = \left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right), \quad \left(\begin{smallmatrix} A \\ A \end{smallmatrix}\right) \otimes \left(\begin{smallmatrix} B \\ B \end{smallmatrix}\right) = \left(\begin{smallmatrix} A \times B \\ A \times B \end{smallmatrix}\right)$$

definita su  $\mathbf{Lens}_A(\mathcal{C})$  è Cartesiana. Vedasi Cruttwell et al., 2022.

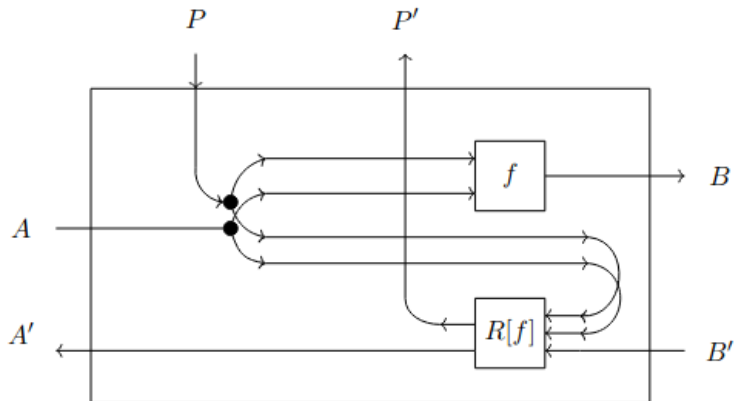
## **DEFINIZIONE: Lenti parametriche**

Sia  $\mathcal{C}$  una CRDC. Allora, definiamo la categoria delle lenti parametriche su  $\mathcal{C}$  come

$$\mathbf{Para}_{\otimes}(\mathbf{Lens}_A(\mathcal{C})).$$

Vedasi Gavranović, 2024.

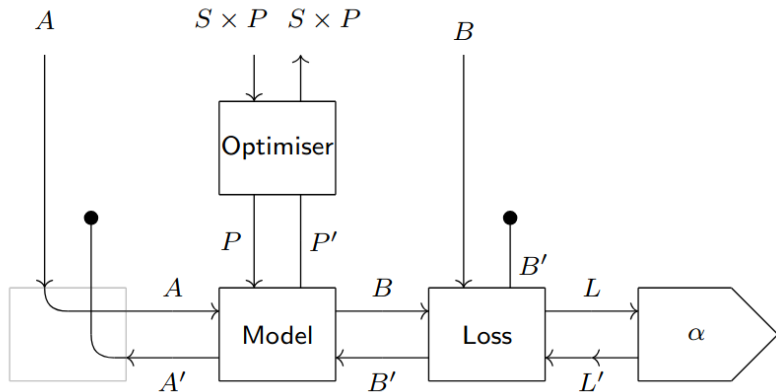
## Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

Le lenti parametriche in  $\mathbf{Para}_{\otimes}(\mathbf{Lens}_A(\mathcal{C}))$  supportano la *automatic differentiation* e possono essere utilizzate per implementare il *gradient-based learning* (Cruttwell et al., 2022, Gavranović, 2024).

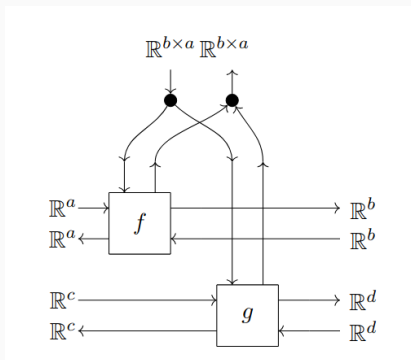
## Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

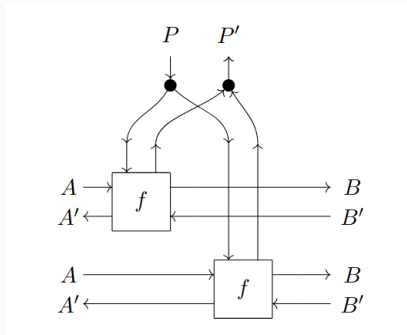


# Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

# Gradient-based learning con lenti parametriche



Cruttwell et al., 2022

# Categorical deep learning:

(co)algebre categoriche

come teoria delle architetture

### Geometric deep learning

Il *geometric deep learning* è una teoria delle architetture di reti neurali che imita l'*Erlangen Programme*, organizzando le architetture in base al concetto di equivarianza rispetto ad azioni di gruppi (Bronstein et al., 2021).

### DEFINIZIONE: Funzione equivariante

Sia  $G$  be un gruppo e siano  $(S, \cdot)$  e  $(T, *)$  azioni di  $G$ . Una funzione  $f : S \rightarrow T$  è equivariante rispetto a tali azioni se

$$f(g \cdot s) = g * f(s),$$

per ogni  $s \in S$  e per ogni  $g \in G$ .

### ESEMPIO

I convolutional layers delle reti neurali rappresentano mappe invarianti rispetto a traslazioni.

### Categorical deep learning

Il *categorical deep learning* è una teoria delle architetture di reti neurali che generalizza il GDL, organizzando le architetture in base al concetto di omomorfismo di (co)algebre categoriche (Gavranović et al., 2024).

### DEFINIZIONE: Algebra su un endofuntore

Sia  $F : \mathcal{C} \rightarrow \mathcal{C}$  un endofuntore. Un'algebra su  $F$  è una coppia  $(A, a)$  dove  $A$  è un oggetto di  $\mathcal{C}$  e  $a : F(A) \rightarrow A$  è un morfismo in  $\mathcal{C}$ .

### DEFINIZIONE: Omomorfismo di algebre

Siano  $(A, a)$  e  $(B, b)$  algebre sullo stesso endofuntore  $F : \mathcal{C} \rightarrow \mathcal{C}$ . Un omomorfismo di algebre  $(A, a) \rightarrow (B, b)$  è un morfismo  $f : A \rightarrow B$  in  $\mathcal{C}$  tale che  $F(f) \circ b = a \circ f$ .

$$\begin{array}{ccc} T(A) & \xrightarrow{T(f)} & T(B) \\ \downarrow a & & \downarrow b \\ A & \xrightarrow{f} & B \end{array}$$



Il CDL generalizza il GDL poiché le azioni di un gruppo  $G$  si possono definire come algebre su una monade, e le mappe invarianti si recuperano come omomorfismi tra queste algebre.

### DEFINIZIONE: Monade delle azioni di $G$

Consideriamo l'endofuntore  $G \times - : \mathbf{Set} \rightarrow \mathbf{Set}$  che mappa  $A \mapsto G \times A$  e  $f \mapsto G \times f$ . La monade delle azioni di  $G$  è definita dotando l'endofuntore delle trasformazioni naturali di  $\mu_A : (g, h, a) \mapsto (gh, a)$  e  $\eta_A : a \mapsto (e, a)$ . (Vedasi Gavranović et al., 2024.)

Il CDL costruisce collega algoritmi e strutture dell'informatica classica con le reti neurali.

### ESEMPIO: Liste

Sia  $A$  un insieme. Consideriamo l'endofuntore  $1 + A \times -$  su **Set**. Sia  $\text{List}(A)$  l'insieme delle liste di elementi di  $A$ . Allora, se  $\text{Nil} : 1 \rightarrow \text{List}(A)$  mappa l'unico oggetto di  $1$  alla lista vuota e  $\text{Cons} : A \times \text{List}(A) \rightarrow \text{List}(A)$  aggiunge un elemento a una lista,  $(\text{List}(A), [\text{Nil}, \text{Cons}])$ , è un algebra su  $1 + A \times -$ . (Esempio da Gavranović et al., 2024.)

## ESEMPIO: List folds

Consideriamo due algebre  $(\text{List}(A), [\text{Nil}, \text{Cons}])$  e  $(Z, [r_0, r_1])$  su  $1 + A \times -$ . Un omomorfismo  $f : \text{List}(A) \rightarrow Z$  tra queste due algebre deve soddisfare

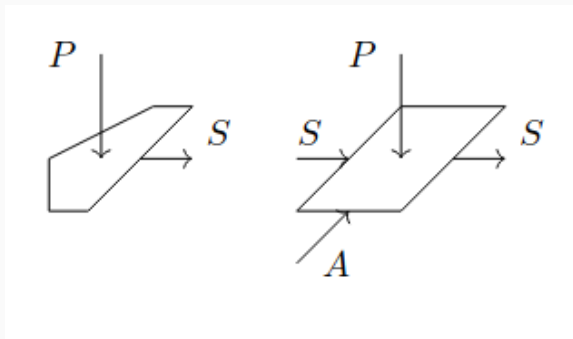
$$\begin{aligned} f(\text{Nil}) &= r_0, \\ f(\text{Cons}(a, l)) &= r_1(a, f(l)). \end{aligned}$$

Hence,  $f$  è necessariamente un *fold* che riduce liste di elementi di  $A$  a singoli elementi di  $Z$ . (Esempio da Gavranović et al., 2024.)

### ESEMPIO: Una cella di un folding RNN

Consideriamo l'endofuntore  $1 + A \times -$  : e la struttura cartesiana  $(1, \times)$  su **Set**. Su questo funtore, può essere costruito un 2-funtore  $\mathbf{Para}(1 + A \times -) : \mathbf{Para}_\bullet(\mathbf{Set}) \rightarrow \mathbf{Para}_\bullet(\mathbf{Set})$ .

Consideriamo un algebra  $(S, (P, \text{Cell}))$  su tale funtore. Tramite l'isomorfismo  $P \times (1 + A \times -) \cong P + P \times A \times X$ , deduciamo che  $\text{Cell} = [\text{Cell}_0, \text{Cell}_1]$ , dove  $\text{Cell}_0 : P \rightarrow S$  e  $\text{Cell}_1 : P \times A \times S \rightarrow S$ . Le funzioni  $\text{Cell}_0$  e  $\text{Cell}_1$  si possono interpretare come celle di un folding recurrent neural network. (Esempio da Gavranović et al., 2024.)



Gavranović et al., 2024

### ESEMPIO: Unrolling di un folding RNN

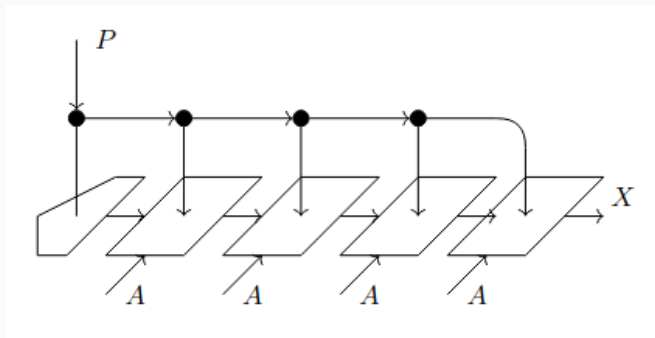
Consideriamo le due algebre  $(\text{List}(A), [\text{Nil}, \text{Cons}])$  e  $(S, (P, \text{Cell}))$  sull'endofuntore  $\mathbf{Para}(1 + A \times -)$ . Ora consideriamo un omomorfismo di algebre

$(P, f, \Delta_P) : (\text{List}(A), [\text{Nil}, \text{Cons}]) \rightarrow (S, (P, \text{Cell}))$ . Si può dimostrare che una funzione  $f$  così definita è l'unrolling di un folding recurrent neural network. L'algebra  $(\text{List}(A), [\text{Nil}, \text{Cons}])$  fornisce gli input della rete neurale, mentre l'algebra  $(S, (P, \text{Cell}))$  fornisce le celle. (Esempio da Gavranović et al., 2024.)

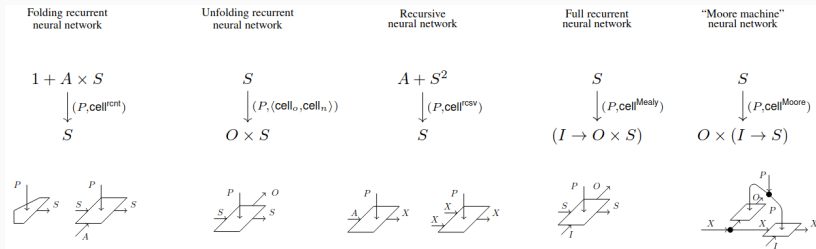
$$\begin{array}{ccc} T(A) & \xrightarrow{T(f)} & T(B) \\ a \downarrow & \swarrow \kappa & \downarrow b \\ A & \xrightarrow{f} & B \end{array}$$

Gavranović et al., 2024





Gavranović et al., 2024



Gavranović et al., 2024

# Functor learning:

modelli come funtori che sfruttano la struttura dei dati

## Categorical representation learning

Il *categorical representation learning* consiste nell'immergere funtorialmente una categoria  $\mathcal{C}$  di dati in una categoria  $\mathcal{R}$  di vettori latenti (Sheshmani e You, 2021).

Grazie alla funtorialità dell'embedding, il *categorical representation learning* consente di preservare la struttura dei dati nello spazio latente.

## **DEFINIZIONE: Struttura categorica dello spazio latente**

tegoria i cui oggetti sono i vettori di  $\mathbb{R}^n$  and tale che, per ogni  $u, v$  in  $\mathcal{R}$ , i morfismi  $u \rightarrow v$  sono le matrici  $M \in \mathbb{R}^{n \times n}$  tali che  $v = Mu$ . La composizione è l'ordinario prodotto riga per colonna e l'identità relativa a un generico  $v \neq 0$  è  $\text{id}_v = \frac{vv^T}{|v|^2}$ . L'identità relativa al vettore nullo è la matrice nulla. (Vedasi Sheshmani e You, 2021.)

Data una categoria  $\mathcal{C}$  di dati, il funtore di *embedding*  $\mathcal{C} \rightarrow \mathcal{R}$  è realizzato da due **neural embedding layers**: il primo produce rappresentazioni degli elementi di  $\mathcal{C}$  come vettori, mentre il secondo produce rappresentazioni dei morfismi di  $\mathcal{C}$  come matrici. (Vedasi Sheshmani e You, 2021.)

## DEFINIZIONE: Negative sampling loss

La *objective function* utilizzata per addestrare i due *layers* è la *negative sampling loss*

$$\mathcal{L} = -\mathbb{E}_{(a,b) \sim p(a,b)} (\log P(a \rightarrow b) + \mathbb{E}_{b' \sim p(b')} \log(1 - P(a \rightarrow b'))) ,$$

dove la probabilità che sussista una relazione  $a \rightarrow b$  è misurata come

$$P(a \rightarrow b) = \text{sigmoid} \left( F \left( \bigoplus_f v_a^T M_f v_b \right) \right) .$$

(Vedasi Sheshmani e You, 2021.)

## **ESEMPIO: Traduzione non supervisionata**

Se  $\mathcal{C}$  e  $\mathcal{D}$  sono database di formule chimiche in inglese e cinese, rispettivamente, possiamo usare il CRL per immergere le due categorie in  $\mathcal{R}$  funtorialmente. Poi si può imparare un funtore  $\mathcal{F} : \mathcal{R} \rightarrow \mathcal{R}$  che preservi la struttura categorica. Tale funtore opererà la traduzione. (Esempio da Sheshmani e You, 2021.)



## ESEMPIO: Traduzione non supervisionata

Se  $\mathcal{C}$  e  $\mathcal{D}$  sono database di formule chimiche in inglese e cinese, rispettivamente, possiamo usare il categorical representation learning per immergere le due categorie in  $\mathcal{R}$  funtorialmente. Poi si può imparare un funtore  $\mathcal{F} : \mathcal{R} \rightarrow \mathcal{R}$  che preservi la struttura categorica. Tale funtore effettuerà la traduzione.  $\mathcal{F}$  può essere implementato come una matrice  $V_{\mathcal{F}}$  che mappa

$$\begin{aligned}v &\mapsto V_{\mathcal{F}}v, \\M_f &\mapsto V_{\mathcal{F}}M_f.\end{aligned}$$

(Esempio da Sheshmani e You, 2021.)

## ESEMPIO: Traduzione non supervisionata

La matrice può essere imparata minimizzando la *loss function*

$$\mathcal{L}_{\text{struc}} = \sum_f \|V_{\mathcal{F}} M_f - M_{\mathcal{F}(f)} V_{\mathcal{F}}\|^2,$$

a cui si può aggiungere anche la *loss*

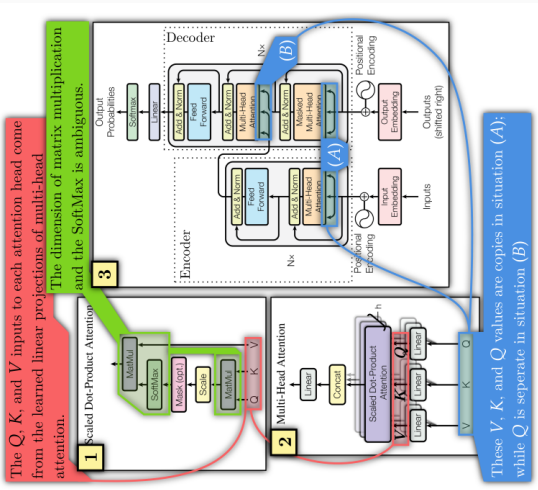
$$\mathcal{L}_{\text{align}} = \sum_{a \in A} \|V_{\mathcal{F}} v_a - v_{\mathcal{F}(a)}\|,$$

per dare parziale supervisione. (Esempio da Sheshmani e You, 2021.)

# Neural circuit diagrams:

rappresentazioni dettagliate di architetture neurali

# Neural circuit diagrams



Abbott, 2024

I diagrammi di *deep learning models* generalmente utilizzati nella letteratura scientifica sono inadeguati in quanto non riportano molti dettagli importanti ai fini dell'implementazione (Abbott, 2024; Khatri et al., 2024).

I *monoidal string diagrams* usati in teoria delle categorie applicata sono inadeguati poiché non riescono a rappresentare funtori e trasformazioni naturali (Abbott e Zardini, 2024).

I *functor string diagrams* prendono ispirazione dai *monoidal string diagrams* e li adattano per rappresentare funtori e trasformazioni naturali (Abbott e Zardini, 2024).

## **PRINCIPIO di decomposizione verticale**

Uno *string diagram* può essere diviso in colonne verticali. Una singola colonna deve contenere solo oggetti o solo morfismi. Colonne con oggetti e colonne con morfismi devono alternarsi.

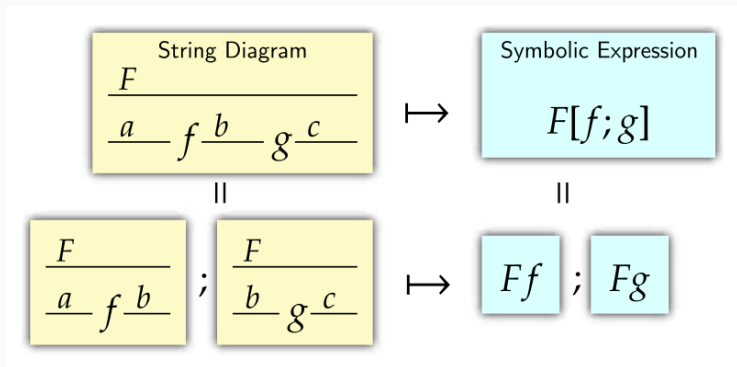
## **PRINCIPIO dell'espressione equivalente**

Deve essere possibile sostituire ogni diagramma che segue una nuova notazione grafica con uno diagramma equivalente che segue la vecchia notazione.

Vedasi Abbott e Zardini, 2024.



# Neural circuit diagrams



Abbott e Zardini, 2024

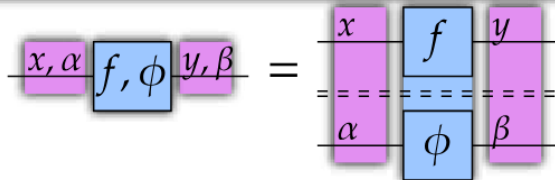
# Neural circuit diagrams

$$\begin{array}{c} \text{Functor Diagram} \\ \hline \frac{F}{a} \boxed{\eta_a} \frac{G}{a} \frac{G}{f} \frac{G}{b} \end{array} = \begin{array}{c} \text{Functor Diagram} \\ \hline \frac{F}{a} \eta \frac{G}{a} \frac{G}{f} \frac{G}{b} \end{array} = \begin{array}{c} \text{Functor Diagram} \\ \hline \frac{F}{a} \frac{F}{f} \eta \frac{G}{b} \frac{G}{b} \end{array}$$

Abbott e Zardini, 2024

# Neural circuit diagrams

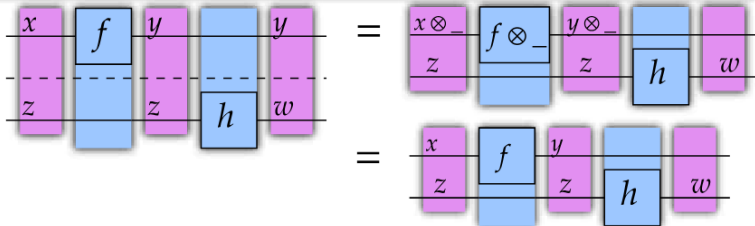
We develop double-dashed lines as an equivalent expression to represent product categories  $\mathcal{C} \times \mathcal{D}$ , where objects and morphisms are tuples.



Abbott e Zardini, 2024

# Neural circuit diagrams

We can reexpress monoidal products using monoidal string diagrams wherein we develop equivalent expressions to view  $x \otimes \_$  and  $f \otimes \_$  as functors and natural transformations.



Abbott e Zardini, 2024

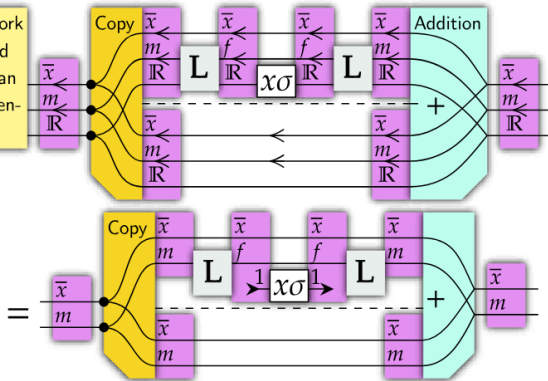
I *neural circuit diagrams* sono *functor string diagrams* specializzati nel rappresentare architetture di reti neurali (Abbott e Zardini, 2024).

# Neural circuit diagrams

A section of a neural network is an expression in **Set**, and can be re-expressed using an equivalent expression for tensor objects.

*Equivalent Expression for Tensor Objects*

$$\begin{array}{|c|} \hline \bar{x} \\ \hline m \\ \hline \mathbb{R} \end{array} = \begin{array}{|c|} \hline \bar{x} \\ \hline m \\ \hline \mathbb{R} \end{array}$$

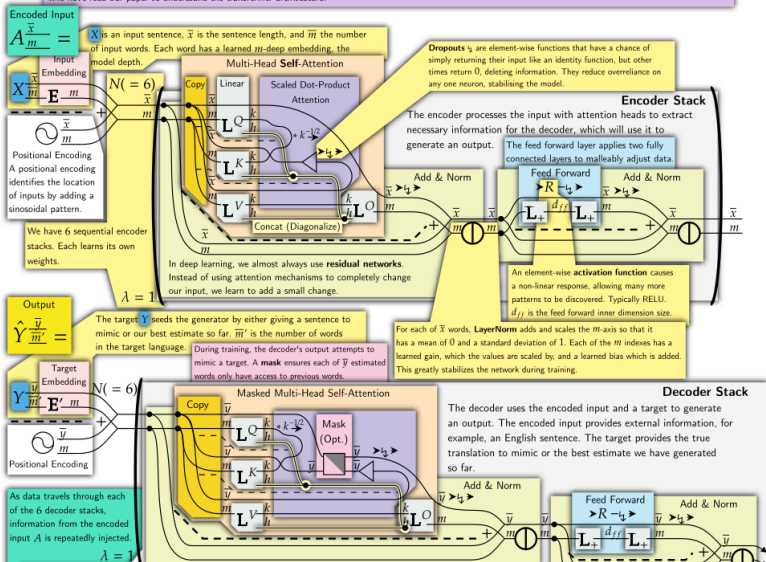


Abbott e Zardini, 2024

## Neural circuit diagrams

### Neural Circuit Diagram for Transformers

Neural circuit diagrams are a visual and explicit framework for representing deep learning models. Transformer architectures have changed the world, and we provide a novel and comprehensive diagram for the original architecture from *Attention is All You Need*. We describe all necessary components, enabling technically proficient novices who have read our paper to understand the transformer architecture.



**Prospettive future**



C'è una competizione in corso tra varie discipline, che puntano a spiegare le reti neurali utilizzando ciascuna i propri strumenti:

- **fisica matematica** (e.g. Roberts et al., 2022),

C'è una competizione in corso tra varie discipline, che puntano a spiegare le reti neurali utilizzando ciascuna i propri strumenti:

- **fisica matematica** (e.g. Roberts et al., 2022),
- **topologia** (e.g. Hensel et al., 2021),

C'è una competizione in corso tra varie discipline, che puntano a spiegare le reti neurali utilizzando ciascuna i propri strumenti:

- **fisica matematica** (e.g. Roberts et al., 2022),
- **topologia** (e.g. Hensel et al., 2021),
- **probabilità** (e.g. Patel et al., 2015),

C'è una competizione in corso tra varie discipline, che puntano a spiegare le reti neurali utilizzando ciascuna i propri strumenti:

- **fisica matematica** (e.g. Roberts et al., 2022),
- **topologia** (e.g. Hensel et al., 2021),
- **probabilità** (e.g. Patel et al., 2015),
- e così via...

La teoria delle categorie, oltre a offrire strumenti propri, potrebbe creare un ponte tra queste discipline e potrebbe unificare i loro approcci in una teoria generale del deep learning.

# Riferimenti bibliografici

---



Abbott, V. (2023). **Robust Diagrams for Deep Learning Architectures: Applications and Theory** [Tesi di dottorato, Honours Thesis, The Australian National University, Canberra].



Abbott, V. (2024). **Neural Circuit Diagrams: Robust Diagrams for the Communication, Implementation, and Analysis of Deep Learning Architectures.** *arXiv preprint arXiv:2402.05424*.



Abbott, V., & Zardini, G. (2024). **Functor String Diagrams: A Novel Approach to Flexible Diagrams for Applied Category Theory.** *arXiv preprint arXiv:2404.00249*.



Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P.

(2021). **Geometric deep learning: Grids, groups, graphs, geodesics, and gauges.** *arXiv preprint arXiv:2104.13478*.



Chytas, S. P., Lokhande, V. S., Li, P., & Singh, V. (2024). **Pooling Image Datasets With Multiple Covariate Shift and Imbalance.** *arXiv preprint arXiv:2403.02598*.



Clark, S., & Pulman, S. (2007). **Combining symbolic and distributional models of meaning.** *American Association for Artificial Intelligence*.



Cockett, R., Cruttwell, G., Gallagher, J., Lemay, J.-S. P., MacAdam, B., Plotkin, G., & Pronk, D. (2019). **Reverse derivative categories.** *arXiv preprint arXiv:1910.07065*.



Coecke, B., Sadrzadeh, M., & Clark, S. (2010). **Mathematical foundations for a compositional distributional model of meaning.** *arXiv preprint arXiv:1003.4394*.



Cruttwell, G. S., Gavranović, B., Ghani, N., Wilson, P., & Zanasi, F. (2022). **Categorical foundations of gradient-based learning.** *European Symposium on Programming*.



Dudzik, A. J., & Veličković, P. (2022). **Graph neural networks are dynamic programmers.** *Advances in neural information processing systems*.



Dudzik, A. J., von Glehn, T., Pascanu, R., & Veličković, P. (2024). **Asynchronous algorithmic alignment with cocycles.** *Learning on Graphs Conference*.



Fong, B., & Spivak, D. I. (2018). **Seven sketches in compositionality: An invitation to applied category theory.** *arXiv preprint arXiv:1803.05316*.



Gavranović, B. (2019). **Compositional deep learning.** *arXiv preprint arXiv:1907.08292*.





Gavranović, B. (2024). **Fundamental Components of Deep Learning: A category-theoretic approach.** *arXiv preprint arXiv:2403.13001.*



Gavranović, B., Lessard, P., Dudzik, A. J., von Glehn, T., Araújo, J. G. M., & Veličković, P. (2024). **Position: Categorical Deep Learning is an Algebraic Theory of All Architectures.** *Forty-first International Conference on Machine Learning.*



Hensel, F., Moor, M., & Rieck, B. (2021). **A survey of topological machine learning methods.** *Frontiers in Artificial Intelligence.*



Khatri, N., Laakkonen, T., Liu, J., & Wang-Maścianica, V. (2024). **On the Anatomy of Attention.** *arXiv preprint arXiv:2407.02423.*



Lewis, M. (2019). **Compositionality for recursive neural networks.** *arXiv preprint arXiv:1901.10723.*



Olah, C., & Carter, S. (2017). **Research Debt** [<https://distill.pub/2017/research-debt>]. *Distill.*



Patel, A. B., Nguyen, T., & Baraniuk, R. G. (2015). **A probabilistic theory of deep learning.** *arXiv preprint arXiv:1504.00641.*



Raff, E. (2019). **A step toward quantifying independently reproducible machine learning research.** *Advances in Neural Information Processing Systems.*



Rahimi, A. (2017). **Machine Learning has become alchemy** [<https://www.youtube.com/watch?v=x7psGHgatGM>].



Roberts, D. A., Yaida, S., & Hanin, B. (2022). **The principles of deep learning theory.** Cambridge University Press  
Cambridge, MA, USA.



Sheshmani, A., & You, Y.-Z. (2021). **Categorical representation learning: morphism is all you need.** *Machine Learning: Science and Technology*.