Universal Imitation Games:
The (Co)End of Generative AI

Sridhar Mahadevan
Adobe Research and University of Massachusetts

Nothing is more practical than a good theory.

~ Ludwig Boltzmann

AZ QUOTES

Birds and Frogs

Freeman Dyson

Some mathematicians are birds, others are frogs. Birds fly high in the air and survey broad vistas of mathematics out to the far horizon. They delight in concepts that unify our thinking and bring together diverse problems from different parts of the landscape. Frogs live in the mud below and see only the flowers that grow nearby. They delight in the details of particular objects, and they solve problems one at a time. I happen to be a frog, but many of my best friends are birds. The main theme of my talk tonight is this. Mathematics needs both birds and frogs. Mathematics is rich and beautiful because birds give it broad visions and frogs give it intricate details. Mathematics is both great art and important science, because it combines generality of concepts with depth of structures. It is stupid to claim that birds are better than frogs because they see farther, or that frogs are better than birds because they see deeper. The world of mathematics is both broad and deep, and we need birds and frogs working together to explore it.

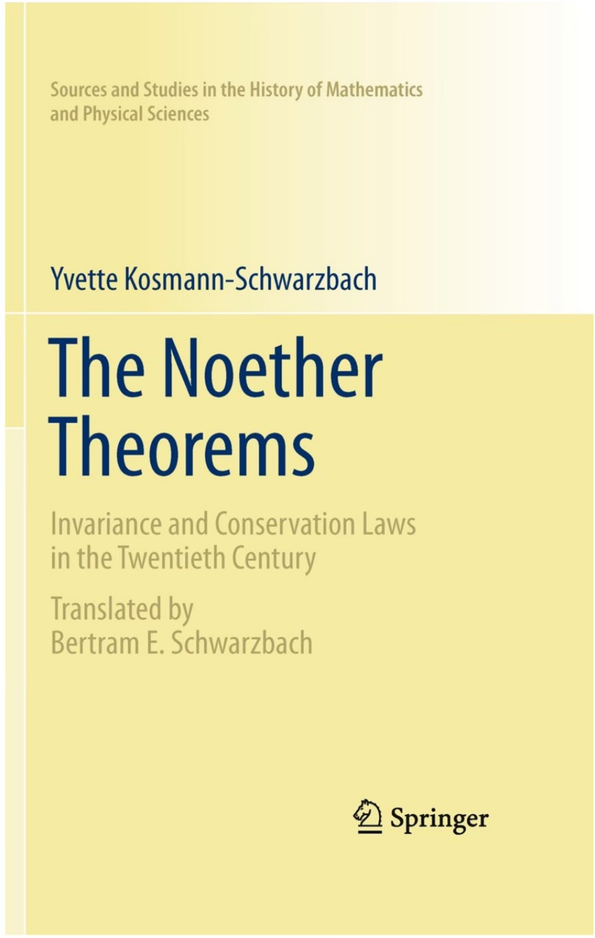
This talk is called the Einstein lecture, and I am grateful to the American Mathematical Society for inviting me to do honor to Albert Einstein. Einstein was not a mathematician, but a physicist who had mixed feelings about mathematics. On the one hand, he had enormous respect for the power of mathematics to describe the workings of nature, and he had an instinct for mathematical beauty which led him onto the right track to find nature's laws. On the other hand, he had no interest in pure mathematics, and he had no technical skill as a mathematician. In his later years he hired younger colleagues with the title of assistants to do mathematical calculations for him. His way of thinking was physical rather than mathematical. He was supreme among physicists as a bird who saw further than others. I will not talk about Einstein since I have nothing new to say.

Freeman Dyson is an emeritus professor in the School of Natural Sciences, Institute for Advanced Study, Princeton, NJ. His email address is dyson@ias.edu.

This article is a written version of his AMS Einstein Lecture, which was to have been given in October 2008 but which unfortunately had to be canceled.

Francis Bacon and René Descartes

At the beginning of the seventeenth century, two great philosophers, Francis Bacon in England and René Descartes in France, proclaimed the birth of modern science. Descartes was a bird, and Bacon was a frog. Each of them described his vision of the future. Their visions were very different. Bacon said, "All depends on keeping the eye steadily fixed on the facts of nature." Descartes said, "I think, therefore I am." According to Bacon, scientists should travel over the earth collecting facts, until the accumulated facts reveal how Nature works. The scientists will then induce from the facts the laws that Nature obeys. According to Descartes, scientists should stay at home and deduce the laws of Nature by pure thought. In order to deduce the laws correctly, the scientists will need only the rules of logic and knowledge of the existence of God. For four hundred years since Bacon and Descartes led the way, science has raced ahead by following both paths simultaneously. Neither Baconian empiricism nor Cartesian dogmatism has the power to elucidate Nature's secrets by itself, but both together have been amazingly successful. For four hundred years English scientists have tended to be Baconian and French scientists Cartesian. Faraday and Darwin and Rutherford were Baconians; Pascal and Laplace and Poincaré were Cartesians. Science was greatly enriched by the cross-fertilization of the two contrasting cultures. Both cultures were always at work in both countries. Newton was at heart a Cartesian, using



In 1915 and 1916 Emmy Noether was asked by Felix Klein and David Hilbert to assist them in understanding issues involved in any attempt to formulate a general theory of relativity, in particular the new ideas of Einstein. She was consulted particularly over the difficult issue of the form a law of conservation of energy could take in the new theory, and she succeeded brilliantly, finding two deep theorems.

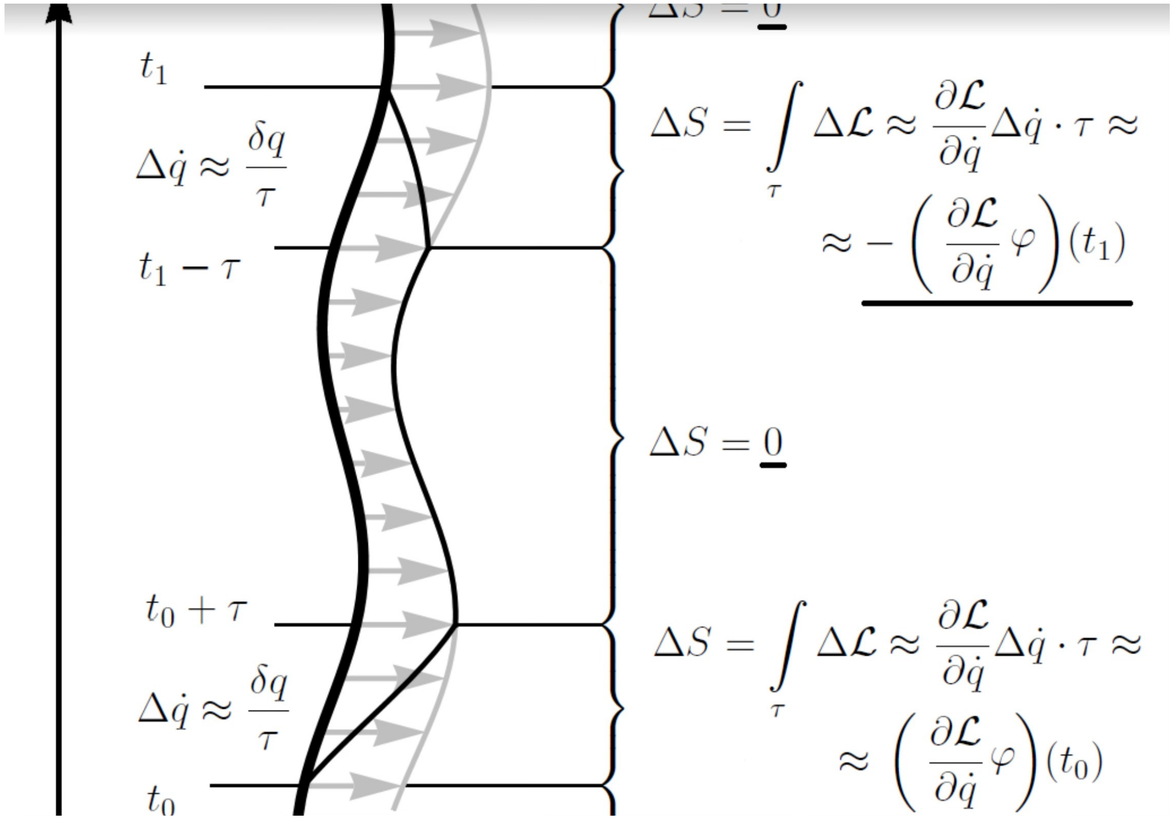
ScienceNews
INDEPENDENT JOURNALISM SINCE 1921

ALL TOPICS LIFE HUMANS EARTH SPACE PHYSICS

FEATURE PHYSICS

In her short life, mathematician Emmy Noether changed the face of physics

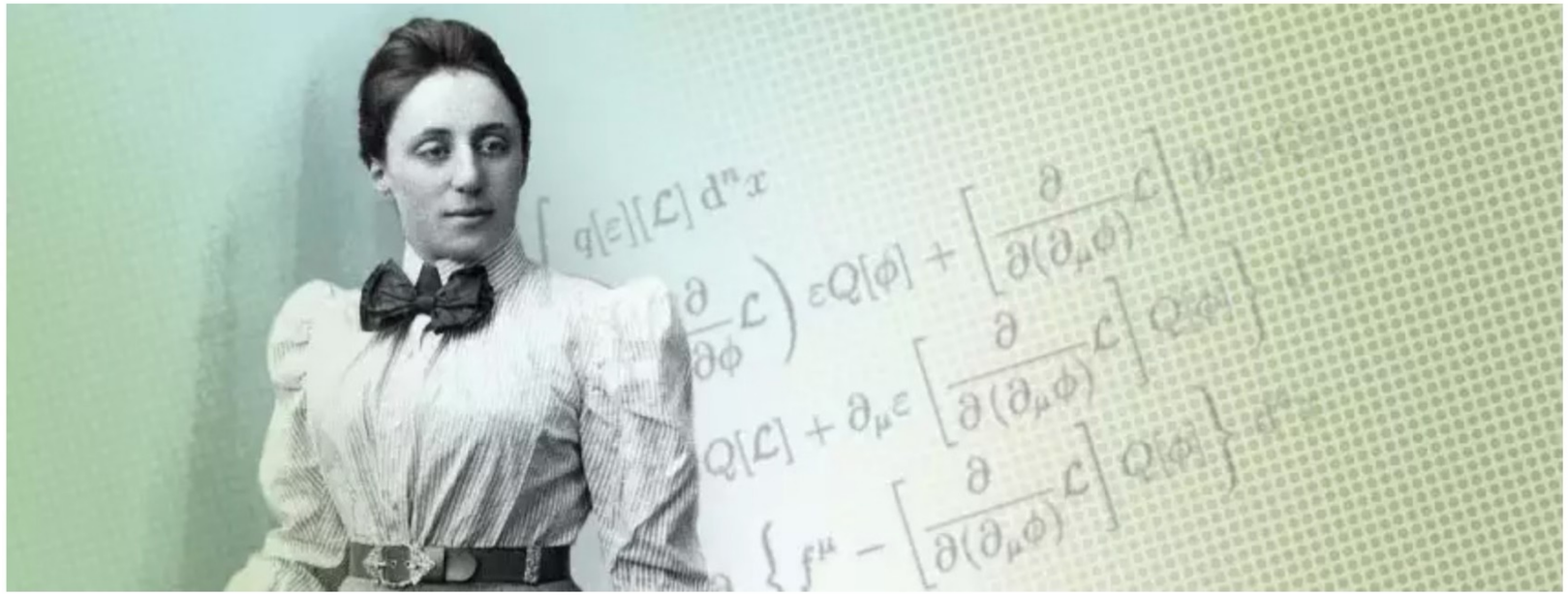
Noether linked two important concepts in physics: conservation laws and symmetries



If a system has a continuous symmetry property, then there are corresponding quantities whose values are conserved in time.

About Noémie Combe

Noémie Combe is a Minerva Fast Track Fellow / research group leader at the Max Planck Institute for Mathematics in the Sciences in Leipzig. Her research is at the intersection of algebraic geometry and algebraic topology. After obtaining her Bachelor's and Master's degree in pure mathematics from the University of Geneva, Switzerland, Combes was awarded a prestigious Labex grant to become a research assistant and PhD student at University of Aix-Marseille and Sorbonne University, where she completed her PhD on configuration spaces in 2018. After a research stay at the Sorbonne University in Paris, she held a postdoc position at the Max Planck Institute for Mathematics in Bonn.



"Without Emmy Noether, there would be a huge gap in mathematics and its understanding"

Mathematics

Noémie Combe, Max Planck Institute for the Mathematics of Sciences, about the brilliant mathematician Emmy Noether, regarded as the inventor of modern algebra

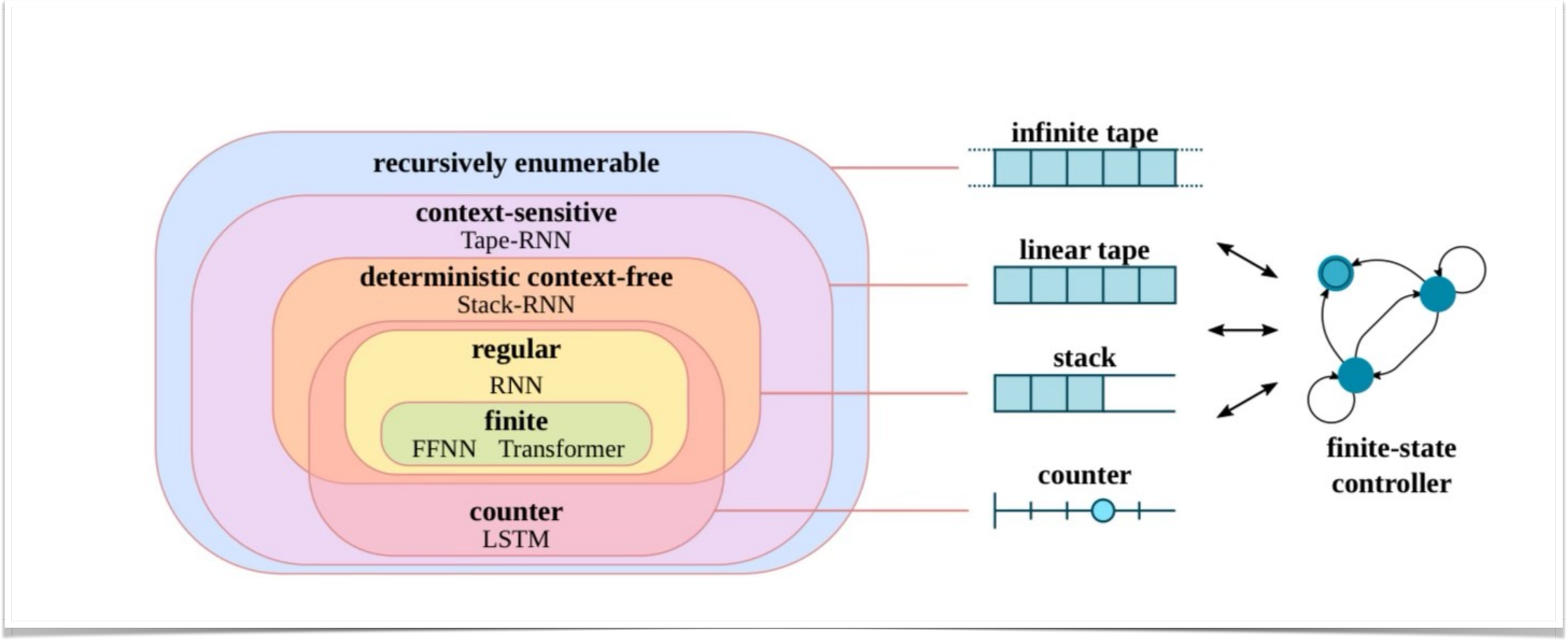


Figure source: Neural Networks and the Chomsky Hierarchy, ICLR 2023

Published as a conference paper at ICLR 2023

NEURAL NETWORKS AND THE CHOMSKY HIERARCHY

Grégoire Delétang[*1] **Anian Ruoss**[*1] **Jordi Grau-Moya**[1] **Tim Genewein**[1] **Li Kevin Wenliang**[1]

Elliot Catt[1] **Chris Cundy**[†2] **Marcus Hutter**[1] **Shane Legg**[1] **Joel Veness**[1] **Pedro A. Ortega**[†]

ABSTRACT

Reliable generalization lies at the heart of safe ML and AI. However, understanding when and how neural networks generalize remains one of the most important unsolved problems in the field. In this work, we conduct an extensive empirical study (20 910 models, 15 tasks) to investigate whether insights from the theory of computation can predict the limits of neural network generalization in practice. We demonstrate that grouping tasks according to the Chomsky hierarchy allows us to forecast whether certain architectures will be able to generalize to out-of-distribution inputs. This includes negative results where even extensive amounts of data and training time never lead to any non-trivial generalization, despite models having sufficient capacity to fit the training data perfectly. Our results show that, for our subset of tasks, RNNs and Transformers fail to generalize on non-regular tasks, LSTMs can solve regular and counter-language tasks, and only networks augmented with structured memory (such as a stack or memory tape) can successfully generalize on context-free and context-sensitive tasks.

Theoretical Limitations of Self-Attention in Neural Sequence Models

Michael Hahn
Stanford University
mhahn2@stanford.edu

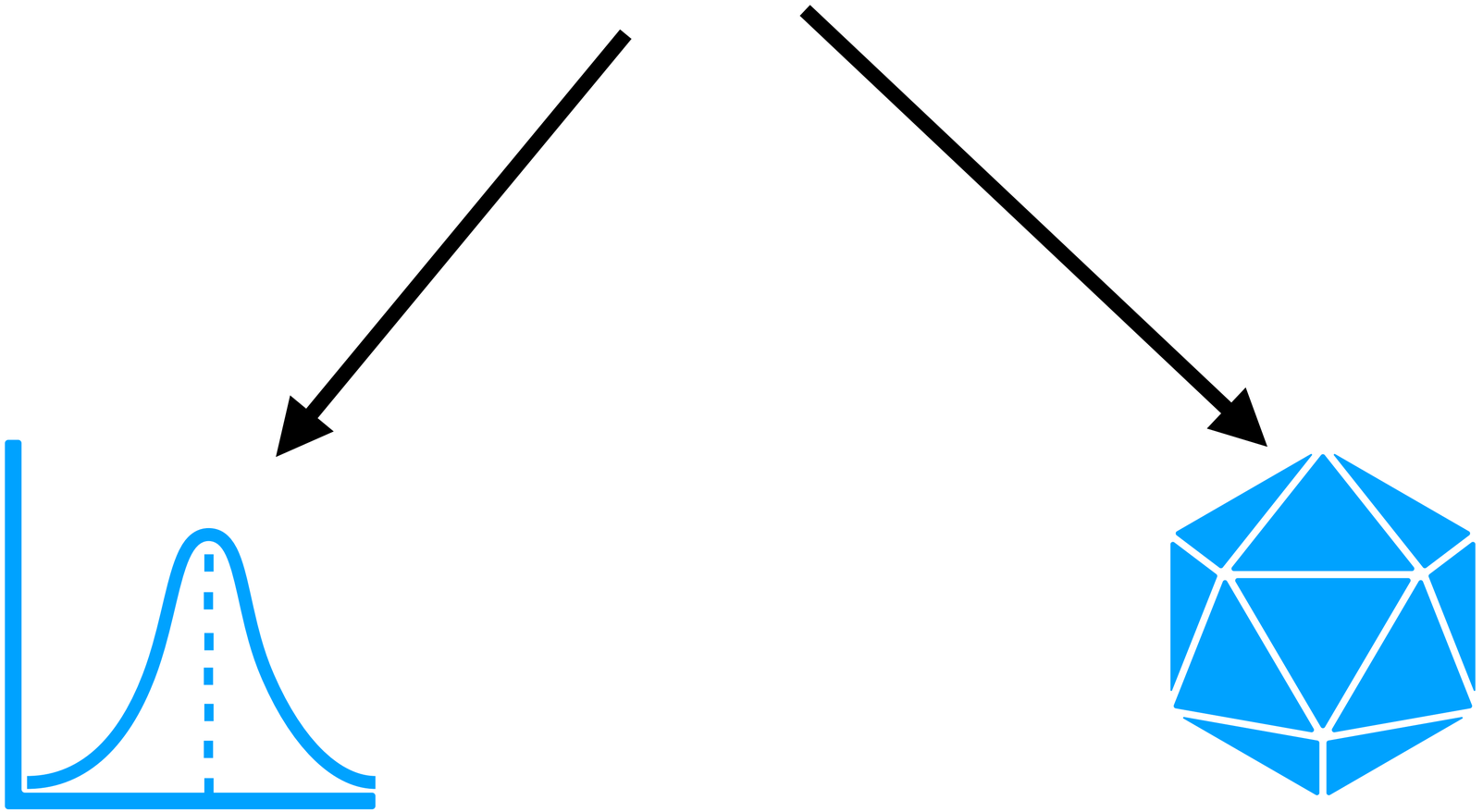
Abstract

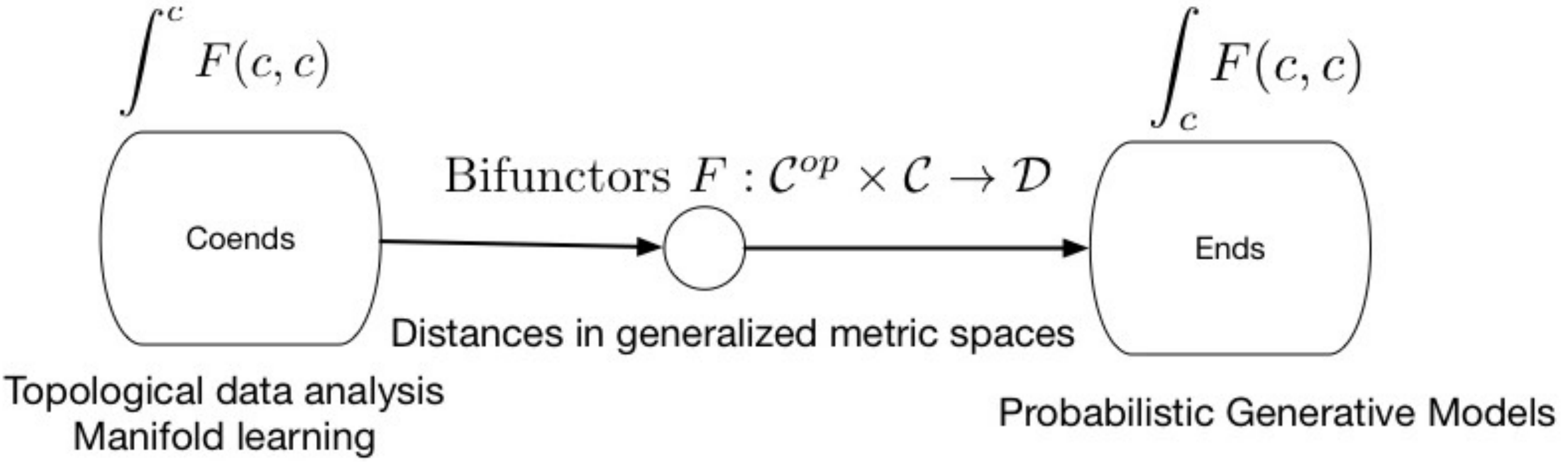
Transformers are emerging as the new workhorse of NLP, showing great success across tasks. Unlike LSTMs, transformers process input sequences entirely through self-attention. Previous work has suggested that the computational capabilities of self-attention to process hierarchical structures are limited. In this work, we mathematically investigate the computational power of self-attention to model formal languages. Across both soft and hard attention, we show strong theoretical limitations of the computational abilities of self-attention, finding that it cannot model periodic finite-state languages, nor hierarchical structure, unless the number of layers or heads increases with input length. These limitations seem surprising given the practical success of self-attention and the prominent role assigned to hierchical structure and recursion. Hierarchical structure is widely thought to be essential to modeling natural language, in particular its syntax (Everaert et al., 2015). Consequently, many researchers have studied the capability of recurrent neural network models to capture context-free languages (e.g., Kalinke and Lehmann (1998); Gers and Schmidhuber (2001); Grüning (2006); Weiss et al. (2018); Sennhauser and Berwick (2018); Korsky and Berwick (2019)) and linguistic phenomena involving hierarchical structure (e.g., Linzen et al. (2016); Gulordava et al. (2018)). Some experimental evidence suggests that transformers might not be as strong as LSTMs at modeling hierarchical structure (Tran et al., 2018), though analysis studies have shown that transformer-based models encode a good amount of syntactic knowledge (e.g., Clark et al. (2019); Lin et al. (2019); Tenney et al. (2019)).

Transformers cannot solve simple problems:

parity, integer modulo arithmetic, balancing arithmetic expressions

Dataset





Mahadevan, Universal
Imitation Games, 2024

Real-World Applications: Category Theory

Haskell is category theory programming

```haskell
quicksort :: Ord a => [a] -> [a]
quicksort []     = []
quicksort (p:xs) = (quicksort lesser) ++ [p] ++ (quicksort greater)
    where
        lesser  = filter (< p) xs
        greater = filter (>= p) xs
```

Haskell

Quicksort

C

```c
// To sort array a[] of size n: qsort(a,0,n-1)

void qsort(int a[], int lo, int hi)
{
  int h, l, p, t;

  if (lo < hi) {
    l = lo;
    h = hi;
    p = a[hi];

    do {
      while ((l < h) && (a[l] <= p))
          l = l+1;
      while ((h > l) && (a[h] >= p))
          h = h-1;
      if (l < h) {
          t = a[l];
          a[l] = a[h];
          a[h] = t;
      }
    } while (l < h);

    a[hi] = a[l];
    a[l] = p;

    qsort( a, lo, l-1 );
    qsort( a, l+1, hi );
  }
}
```

Why Cardano chose Haskell — and why you should care

Cardano Foundation · Follow

5 min read · Dec 16, 2020

912 4

*Written by **@ElliotHill** of the Cardano Foundation*

UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction

Uniform Manifold Approximation and Projection (UMAP) is a dimension reduction technique that can be used for visualisation similarly to t-SNE, but also for general non-linear dimension reduction. The algorithm is founded on three assumptions about the data

1. The data is uniformly distributed on Riemannian manifold;
2. The Riemannian metric is locally constant (or can be approximated as such);
3. The manifold is locally connected.

From these assumptions it is possible to model the manifold with a fuzzy topological structure. The embedding is found by searching for a low dimensional projection of the data that has the closest possible equivalent fuzzy topological structure.

The details for the underlying mathematics can be found in our paper on ArXiv:

McInnes, L, Healy, J, *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, ArXiv e-prints 1802.03426, 2018

You can find the software on github.

Installation

UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction

Leland McInnes
Tutte Institute for Mathematics and Computing
leland.mcinnes@gmail.com

John Healy
Tutte Institute for Mathematics and Computing
jchealy@gmail.com

James Melville
jlmelville@gmail.com

September 21, 2020

Abstract

UMAP (Uniform Manifold Approximation and Projection) is a novel manifold learning technique for dimension reduction. UMAP is constructed from a theoretical framework based in Riemannian geometry and algebraic topology. The result is a practical scalable algorithm that is applicable to real world data. The UMAP algorithm is competitive with t-SNE for visualization quality, and arguably preserves more of the global structure with superior run time performance. Furthermore, UMAP has no computational restrictions on embedding dimension, making it viable as a general purpose dimension reduction technique for machine learning.

1 Introduction

Dimension reduction plays an important role in data science, being a fundamental technique in both visualisation and as pre-processing for machine

Definition 1. *The category* $\mathbf{\Delta}$ *has as objects the finite order sets* $[n] = \{1, \dots, n\}$*, with morphims given by (non-strictly) order-preserving maps.*

Following standard category theoretic notation, $\mathbf{\Delta}^{\mathrm{op}}$ denotes the category with the same objects as $\mathbf{\Delta}$ and morphisms given by the morphisms of $\mathbf{\Delta}$ with the direction (domain and codomain) reversed.

Definition 2. *A* simplicial set *is a functor from* $\mathbf{\Delta}^{op}$ *to* **Sets**, *the category of sets; that is, a contravariant functor from* $\mathbf{\Delta}$ *to* **Sets**.

Given a simplicial set $X : \mathbf{\Delta}^{\mathrm{op}} \to \mathbf{Sets}$, it is common to denote the set $X([n])$ as X_n and refer to the elements of the set as the n-simplices of X. The simplest possible examples of simplicial sets are the *standard simplices* Δ^n, defined as the representable functors $\hom_{\mathbf{\Delta}}(\cdot, [n])$. It follows from the Yoneda lemma that there is a natural correspondence between n-simplices of X and morphisms $\Delta^n \to X$ in the category of simplicial sets, and it is often helpful to think in these terms. Thus for each $x \in X_n$ we have a corresponding morphism $x : \Delta^n \to X$. By the density theorem and employing a minor abuse of notation we then have

$$\operatorname*{colim}_{x \in X_n} \Delta^n \cong X$$

There is a standard covariant functor $|\cdot| : \mathbf{\Delta} \to \mathbf{Top}$ mapping from the category $\mathbf{\Delta}$ to the category of topological spaces that sends $[n]$ to the standard n-simplex $|\Delta^n| \subset \mathbb{R}^{n+1}$ defined as

$$|\Delta^n| \triangleq \left\{ (t_0, \dots, t_n) \in \mathbb{R}^{n+1} \mid \sum_{i=0}^{n} t_i = 1, t_i \geq 0 \right\}$$

with the standard subspace topology. If $X : \mathbf{\Delta}^{\mathrm{op}} \to \mathbf{Sets}$ is a simplicial set then we can construct the realization of X (denoted $|X|$) as the colimit

$$|X| = \operatorname*{colim}_{x \in X_n} |\Delta^n|$$

Definition 7. *Define the functor* $\mathsf{FinReal} : \mathbf{Fin\text{-}sFuzz} \to \mathbf{FinEPMet}$ *by setting*

$$\mathsf{FinReal}(\Delta^n_{<a}) \triangleq (\{x_1, x_2, \ldots, x_n\}, d_a),$$

where

$$d_a(x_i, x_j) = \begin{cases} -\log(a) & \text{if } i \neq j, \\ 0 & \text{otherwise} \end{cases}.$$

and then defining

$$\mathsf{FinReal}(X) \triangleq \operatorname*{colim}_{\Delta^n_{<a} \to X} \mathsf{FinReal}(\Delta^n_{<a}).$$

[McInnes et al., 2020]

Definition 8. *Define the functor* FinSing : **FinEPMet** $\to$ **Fin-sFuzz** *by*

$$\mathsf{FinSing}(Y) : ([n], [0, a)) \mapsto \hom_{\mathbf{FinEPMet}}(\mathsf{FinReal}(\Delta^n_{<a}), Y).$$

We then have the following theorem.

Theorem 1. *The functors* FinReal : **Fin-sFuzz** $\to$ **FinEPMet** *and* FinSing : **FinEPMet** $\to$ **Fin-sFuzz** *form an adjunction with* FinReal *the left adjoint and* FinSing *the right adjoint.*

[McInnes et al., 2020]

Algorithm 1 UMAP algorithm

function UMAP(X, n, d, min-dist, n-epochs)

 # Construct the relevant weighted graph

for all $x \in X$ **do**

fs-set$[x] \leftarrow$ LocalFuzzySimplicialSet(X, x, n)

top-rep $\leftarrow \bigcup_{x \in X}$ fs-set$[x]$ *# We recommend the probabilistic t-conorm*

 # Perform optimization of the graph layout

$Y \leftarrow$ SpectralEmbedding(top-rep, d)

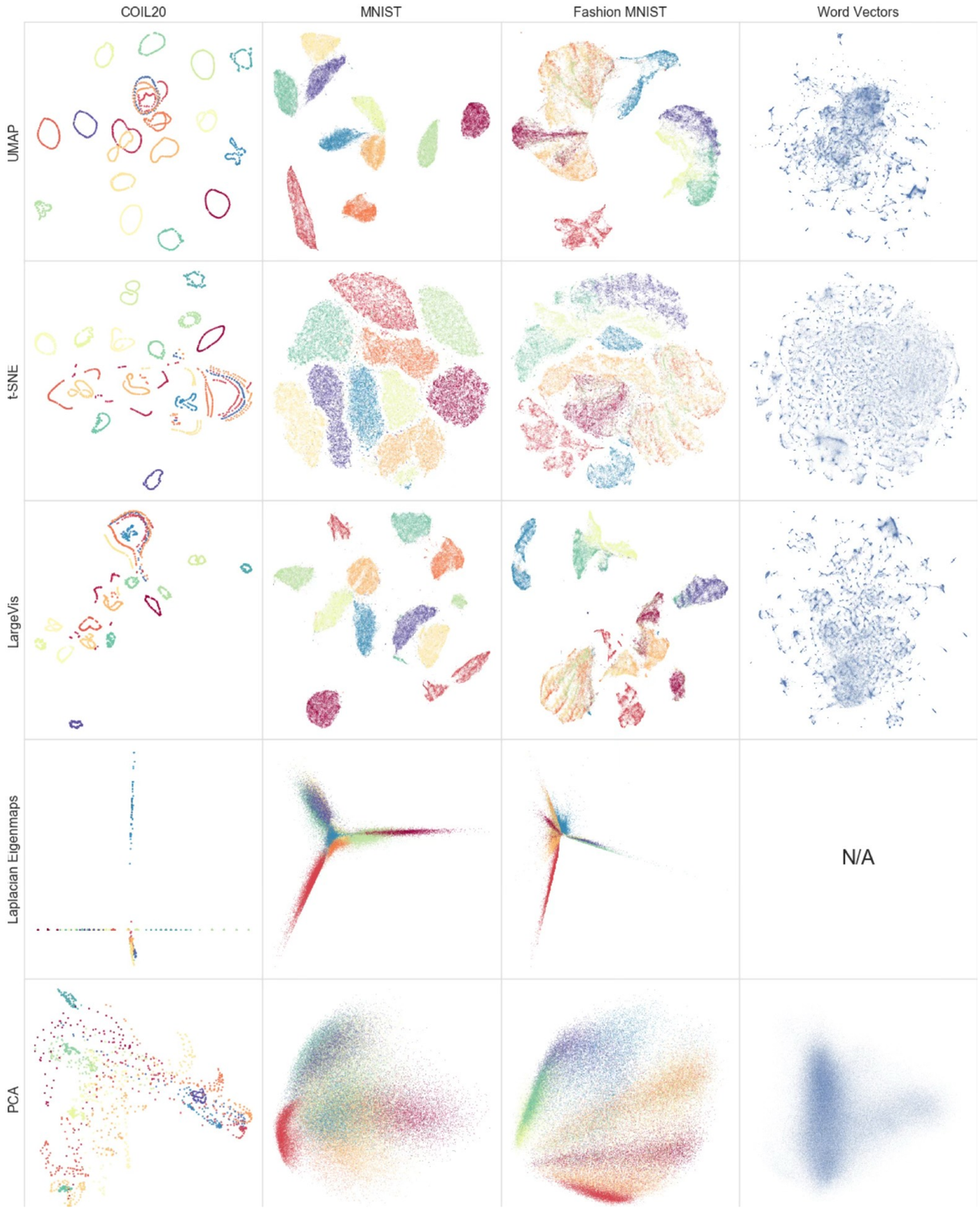
$Y \leftarrow$ OptimizeEmbedding(top-rep, Y, min-dist, n-epochs)

return Y

[McInnes et al., 2020]

UMAP

PCA


COIL20
MNIST
Fashion MNIST
Word Vectors
UMAP
t-SNE
LargeVis
Laplacian Eigenmaps
PCA
N/A

https://doi.org/10.1038/s41586-019-0969-x

The single-cell transcriptional landscape of mammalian organogenesis

Junyue Cao[1,2,10], Malte Spielmann[1,10], Xiaojie Qiu[1,2], Xingfan Huang[1,3], Daniel M. Ibrahim[4,5], Andrew J. Hill[1], Fan Zhang[6], Stefan Mundlos[4,5], Lena Christiansen[6], Frank J. Steemers[6], Cole Trapnell[1,7,8]* & Jay Shendure[1,7,8,9]*

Mammalian organogenesis is a remarkable process. Within a short timeframe, the cells of the three germ layers transform into an embryo that includes most of the major internal and external organs. Here we investigate the transcriptional dynamics of mouse organogenesis at single-cell resolution. Using single-cell combinatorial indexing, we profiled the transcriptomes of around 2 million cells derived from 61 embryos staged between 9.5 and 13.5 days of gestation, in a single experiment. The resulting 'mouse organogenesis cell atlas' (MOCA) provides a global view of developmental processes during this critical window. We use Monocle 3 to identify hundreds of cell types and 56 trajectories, many of which are detected only because of the depth of cellular coverage, and collectively define thousands of corresponding marker genes. We explore the dynamics of gene expression within cell types and trajectories over time, including focused analyses of the apical ectodermal ridge, limb mesenchyme and skeletal muscle.

Most studies of mammalian organogenesis rely on model organisms, and, in particular, the mouse. Mice develop quickly, with just 21 days between fertilization and birth. The implantation of the blastocyst on embryonic day (E) 4.0 is followed by gastrulation and the formation of germ layers on E6.5–E7.5[1,2]. At the early-somite stages, the embryo transits from gastrulation to early organogenesis, forming the neural plate and heart tube (E8.0–E8.5). In the ensuing days (E9.5–E13.5), the embryo expands from hundreds-of-thousands to over ten-million cells, and concurrently develops nearly all major organ systems. Unsurprisingly, these four days have been intensively studied. Indeed, most genes that underlie major developmental defects can be studied in this window[3,4].

The transcriptional profiling of single cells (scRNA-seq) represents a promising strategy for obtaining a global view of developmental processes[5–7]. For example, scRNA-seq recently revealed a large degree of heterogeneity in neurons and myocardiocytes during mouse development[8,9]. However, although two scRNA-seq atlases of the mouse were recently released[10,11], they are mostly restricted to adult organs and do not attempt to characterize the emergence and dynamics of cell types during development.

Single-cell RNA-seq of two million cells

Single-cell combinatorial indexing is a methodological framework involving split-pool barcoding of cells or nuclei[12–19]. We previously developed single-cell combinatorial-indexing RNA-sequencing analysis (sci-RNA-seq) and applied it to generate 50-fold shotgun coverage of the cellular content of L2-stage *Caenorhabditis elegans*[17]. A conceptually identical method was recently termed SPLiT-seq[20]. To increase the throughput, we explored more than 1,000 experimental conditions (Extended Data Fig. 1a, b, Methods). The major improvements of the resulting method, sci-RNA-seq3, include: (i) nuclei are extracted directly from fresh tissues without enzymatic treatment, then fixed and stored; (ii) for the third level of indexing[17], we switched from Tn5 tagmentation to hairpin ligation; (iii) individual enzymatic reactions were optimized; and (iv) fluorescence-activated cell sorting was replaced by dilution, and sonication and filtration steps were added to minimize aggregation. Even without automation, sci-RNA-seq3 library preparation can be completed through the intensive effort of a single researcher in one week at a cost of less than $0.01 per cell.

We collected 61 C57BL/6 mouse embryos at E9.5, E10.5, E11.5, E12.5 or E13.5, and snap-froze them in liquid nitrogen. Nuclei from each embryo were isolated and deposited in individual wells in four 96-well plates, such that the first index identified the originating embryo of a given cell. As a control, we spiked a mixture of human HEK-293T and mouse NIH/3T3 nuclei into two wells. The resulting sci-RNA-seq3 library was sequenced in a single Illumina NovaSeq run, yielding 11 billion reads (Fig. 1a, Extended Data Fig. 1c, d).

From one experiment, we recovered 2,058,652 cells from mouse embryos and 13,359 cells from HEK-293T or NIH/3T3 cells (UMI (unique molecular identifier) count $\geq$ 200). Transcriptomes from human or mouse control wells were overwhelmingly species-coherent (3% collisions), with performance similar to previous experiments[17] (Extended Data Fig. 1e–i). A limitation is that only around 7% of cells entering the experiment were ultimately profiled, with losses largely consequent on filtration steps intended to remove aggregates of nuclei.

We profiled a median of 35,272 cells per embryo (Fig. 1b, Extended Data Fig. 1j). Despite shallow sequencing (about 5,000 raw reads per cell; 46% duplicate rate), we recovered a median of 671 UMIs (519 genes) per cell (Extended Data Fig. 1k). The 3.7-fold-deeper sequencing of a subset of wells nearly doubled the complexity (to a median of 1,142 UMIs per cell; 87% duplicate rate). Given that we are profiling RNA in nuclei, 59% of UMIs per cell strand specifically mapped to introns and 25% mapped to exons. The profiles may therefore primarily reflect nascent transcription, temporally offset, but also predictive[21] of the cellular transcriptome. Later-stage embryos exhibited somewhat reduced UMI counts, possibly reflecting decreasing nuclear mRNA content (Extended Data Fig. 1l). We used Scrublet[22] to detect 4.3% likely doublet cells, corresponding to a doublet estimate of 10.3% including both within-cluster and between-cluster doublets (Extended Data Fig. 1m, n).

[1]Department of Genome Sciences, University of Washington, Seattle, WA, USA. [2]Molecular and Cellular Biology Program, University of Washington, Seattle, WA, USA. [3]Department of Computer Science, University of Washington, Seattle, WA, USA. [4]Max Planck Institute for Molecular Genetics, RG Development & Disease, Berlin, Germany. [5]Institute for Medical and Human Genetics, Charité Universitätsmedizin Berlin, Berlin, Germany. [6]Illumina, San Diego, CA, USA. [7]Brotman Baty Institute for Precision Medicine, Seattle, WA, USA. [8]Allen Discovery Center for Cell Lineage Tracing, Seattle, WA, USA. [9]Howard Hughes Medical Institute, Seattle, WA, USA. [10]These authors contributed equally: Junyue Cao, Malte Spielmann. *e-mail: coletrap@uw.edu; shendure@uw.edu

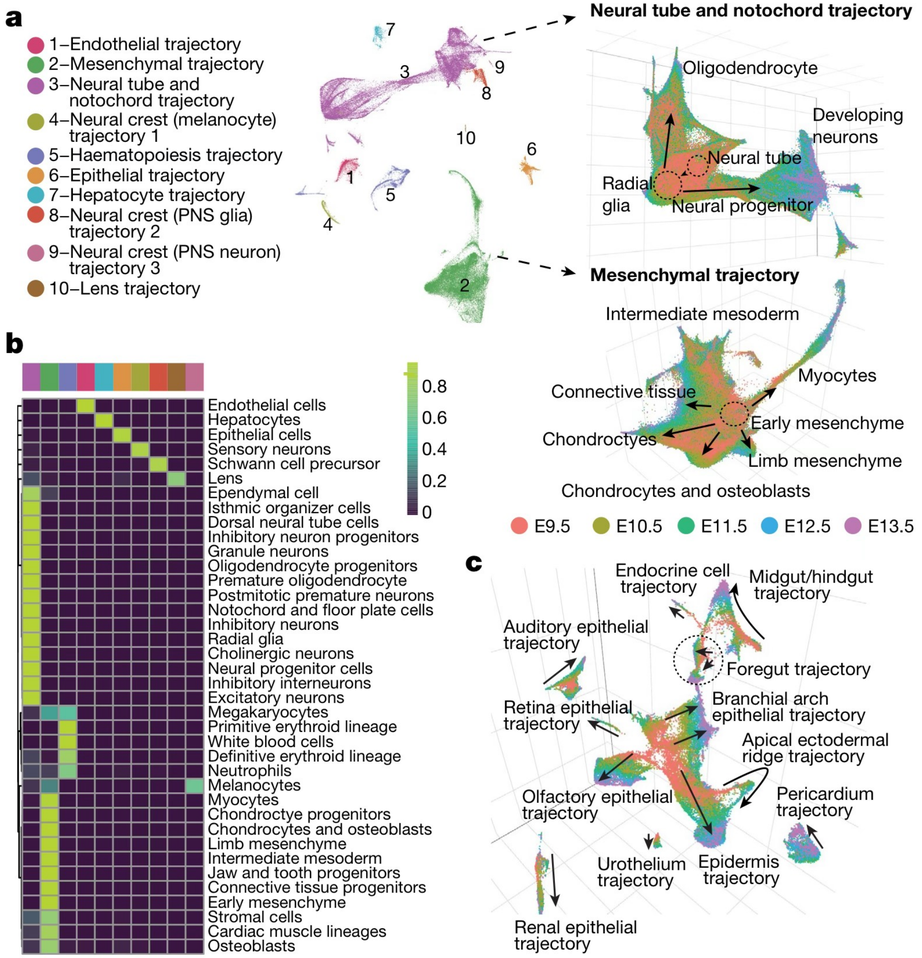
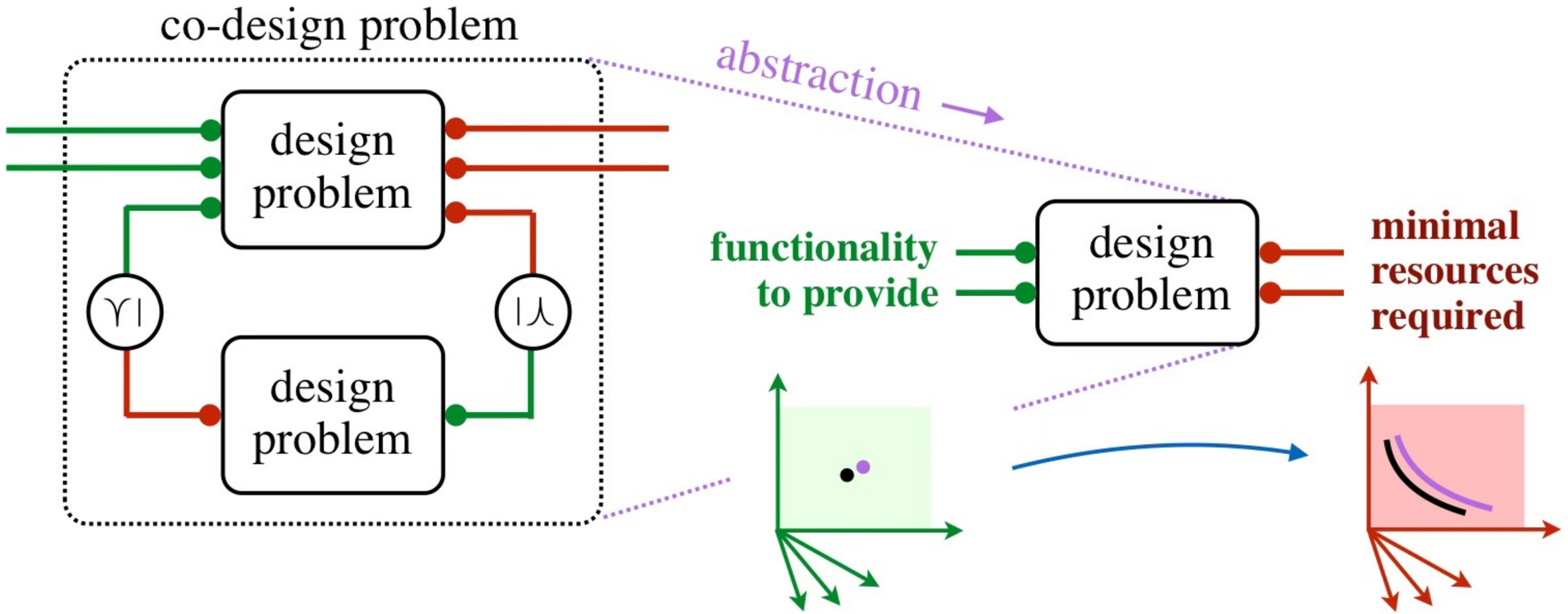


Fig. 4 | Characterization of ten major developmental trajectories present during mouse organogenesis. a, UMAP 3D visualization of our overall dataset. Left, views from one direction; bottom: zoomed view of neural tube–notochord (top) and mesenchymal (bottom) trajectories, coloured by development stage. **b**, Heat map showing the proportion of cells from each of the 38 major cell types (rows) assigned to each of the 10 major trajectories (columns, colour key in **a**, left). **c**, UMAP 3D visualization of epithelial subtrajectories coloured by development stage (colour key in **a**, right).

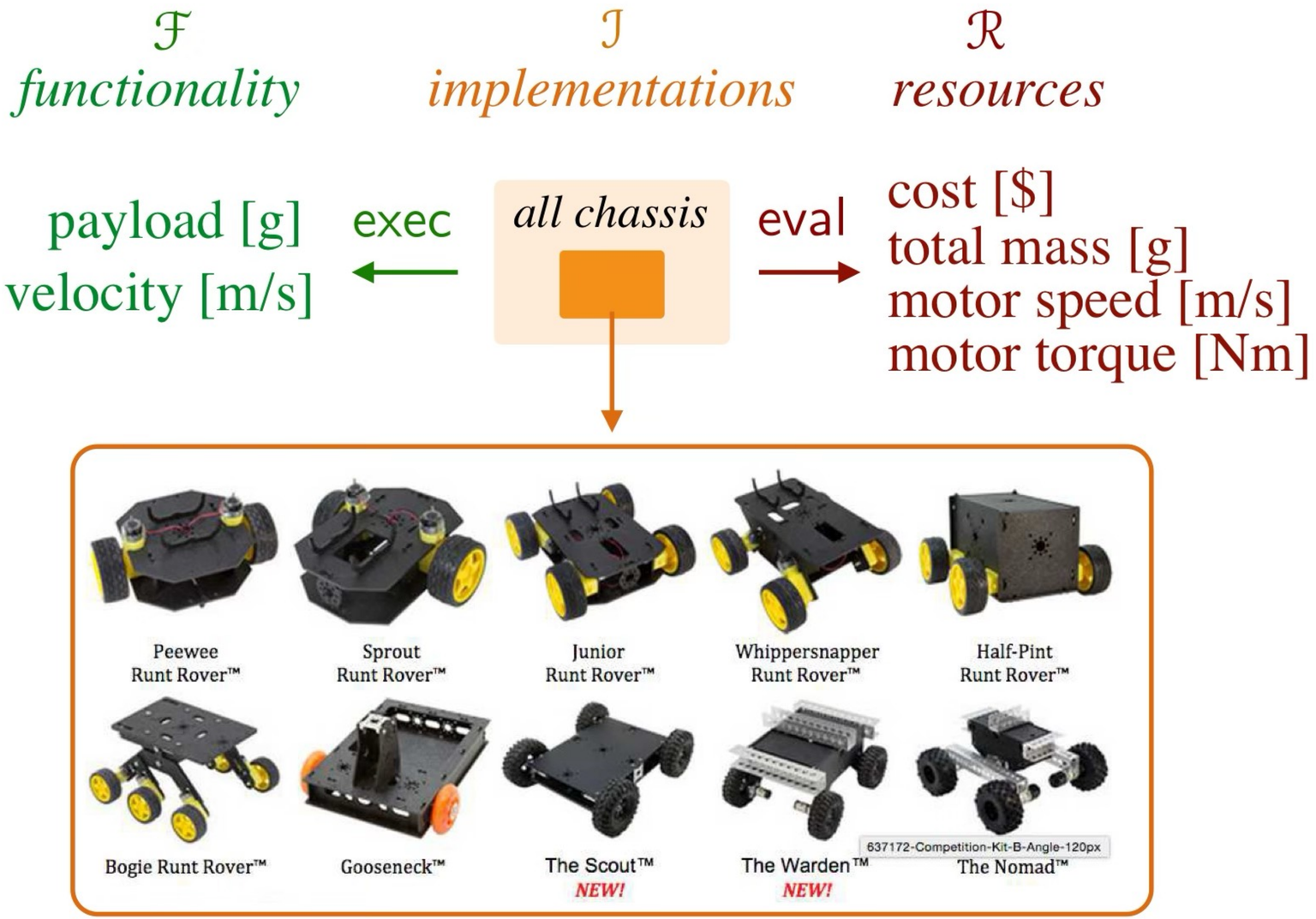
simplicial set representations. UMAP then optimizes the lower-dimension embedding, minimizing the cross-entropy between the low-dimensional representation and the high-dimensional one.

The computational efficiency of UMAP markedly accelerated the analysis of the mouse embryo data. We found that UMAP finished processing the two-million-cell dataset in around 3 CPU hours whereas *t*-SNE took more than 64 CPU hours. A few implementation details lead to the effectiveness of UMAP. Two major steps are involved in both the UMAP and *t*-SNE algorithms: first, the preprocessing step before UMAP is similar to Monocle 2. In brief, genes expressed in fewer than 10 cells (or fewer than 5 cells in datasets with fewer than 1,000 cells) were filtered out. The digital gene-count matrix was first normalized by cell-specific size factor

Collaborative design in robotics



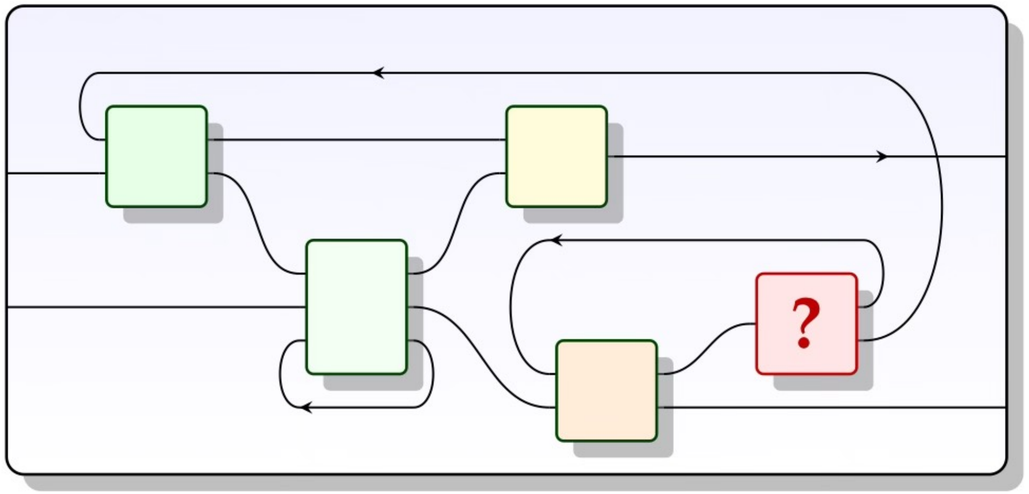
Andrea Censi. Mathematical Theory of co-design, 2016



Andrea Censi. Mathematical Theory of co-design, 2016

Seven Sketches in Compositionality:

An Invitation to Applied Category Theory



Brendan Fong David I. Spivak

Definition 4.2. Let $\mathcal{X} = (X, \leq_X)$ and $\mathcal{Y} = (Y, \leq_Y)$ be preorders. A *feasibility relation* for $\mathcal{X}$ given $\mathcal{Y}$ is a monotone map

$$\Phi \colon \mathcal{X}^{\mathrm{op}} \times \mathcal{Y} \to \mathbf{Bool}. \tag{4.3}$$

We denote this by $\Phi \colon \mathcal{X} \nrightarrow \mathcal{Y}$.

Given $x \in X$ and $y \in Y$, if $\Phi(x, y) = \texttt{true}$ we say *x can be obtained given y*.

4.2.2 $\mathcal{V}$-profunctors

We are now ready to recast Eq. (4.3) in abstract terms. Recall the notions of enriched product (Definition 2.74), enriched functor (Definition 2.69), and quantale (Definition 2.79).

Definition 4.8. Let $\mathcal{V} = (V, \leq, I, \otimes)$ be a (unital commutative) quantale,[1] and let $\mathcal{X}$ and $\mathcal{Y}$ be $\mathcal{V}$-categories. A *$\mathcal{V}$-profunctor* from $\mathcal{X}$ to $\mathcal{Y}$, denoted $\Phi \colon \mathcal{X} \nrightarrow \mathcal{Y}$, is a $\mathcal{V}$-functor

$$\Phi \colon \mathcal{X}^{\mathrm{op}} \times \mathcal{Y} \to \mathcal{V}.$$

DisCoPy

The Python toolkit for computing with string diagrams

DisCoPy is a Python toolkit for computing with string diagrams.

- Documentation: https://docs.discopy.org
- Repository: https://github.com/discopy/discopy

Why?

Applied category theory is information plumbing. It's boring… but *plumbers save lives than doctors*.

As string diagrams become as ubiquitous as matrices, they need their own funda package: *DisCoPy*.

How?

DisCoPy began as an implementation of:

- DisCoCat (distributional compositional categorical) models,
- and QNLP (quantum natural language processing).

This application has now been packaged into its own library, lambeq.

Who?

- Giovanni de Felice (CEO)
- Alexis Toumi (COO)
- Richie Yeung (CFO)
- Boldizsár Poór (CTO)
- Bob Coecke (Honorary President)

Diagrammatic Differentiation for Quantum Machine Learning

Alexis Toumi[⋆†], Richie Yeung[†], Giovanni de Felice[⋆†]

⋆ Department of Computer Science, University of Oxford † Cambridge Quantum Computing Ltd.

We introduce diagrammatic differentiation for tensor calculus by generalising the dual number construction from rigs to monoidal categories. Applying this to ZX diagrams, we show how to calculate diagrammatically the gradient of a linear map with respect to a phase parameter. For diagrams of parametrised quantum circuits, we get the well-known parameter-shift rule at the basis of many variational quantum algorithms. We then extend our method to the automatic differentation of hybrid classical-quantum circuits, using diagrams with bubbles to encode arbitrary non-linear operators. Moreover, diagrammatic differentiation comes with an open-source implementation in DisCoPy, the Python library for monoidal categories. Diagrammatic gradients of classical-quantum circuits can then be simplified using the PyZX library and executed on quantum hardware via the tket compiler. This opens the door to many practical applications harnessing both the structure of string diagrams and the computational power of quantum machine learning.

Introduction

String diagrams are a graphical language introduced by Penrose [33] to manipulate tensor expressions: wires represent vector spaces, nodes represent multi-linear maps between them. In [34], these diagrams are used to describe the geometry of space-time and an extra piece of notation is introduced: the covariant derivative is represented as a bubble around the tensor to be differentiated. Joyal and Street [25, 26] characterised string diagrams as the arrows of free monoidal categories, however their geometry of tensor

Meaning arises out of words in a sentence using ‘quantum entanglement

Category Theory for Quantum Natural Language Processing

Alexis TOUMI

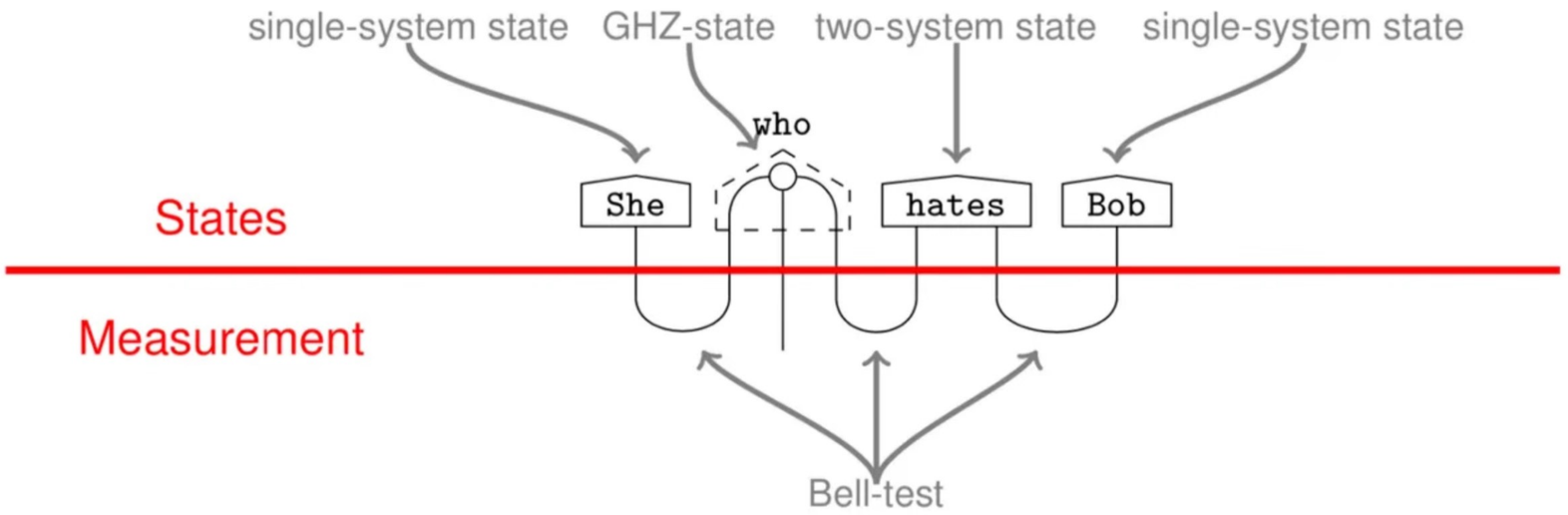
Wolfson College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

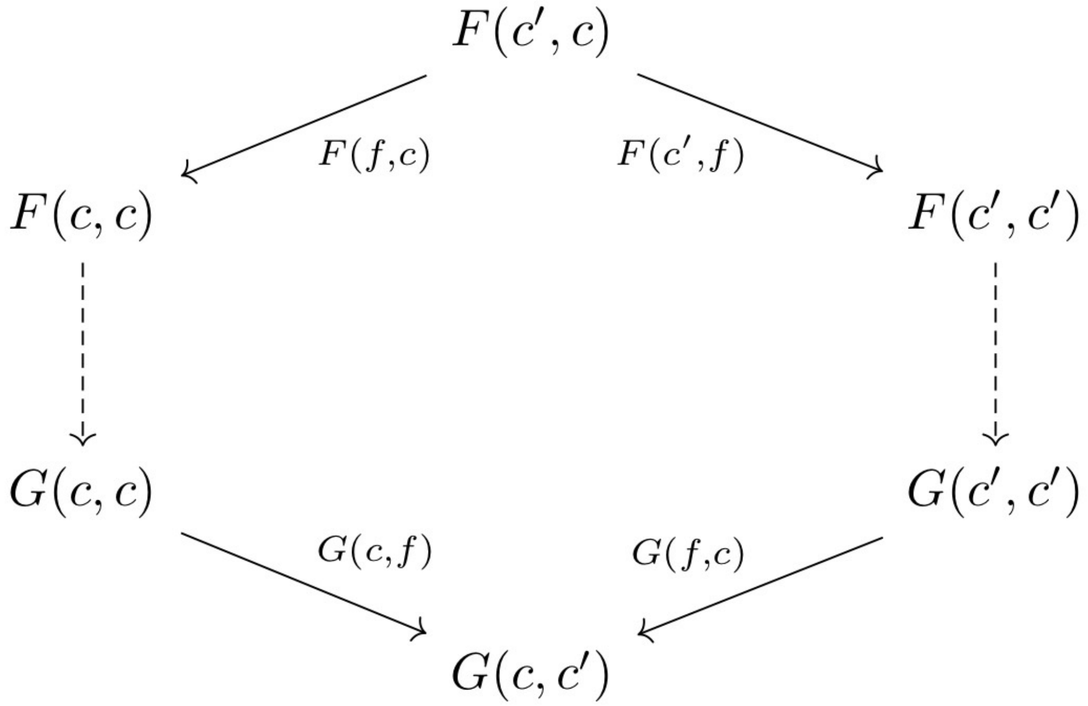
Trinity 2022



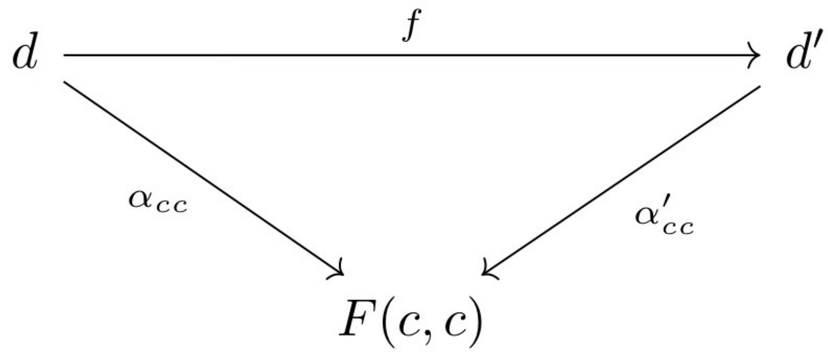
https://medium.com/cambridge-quantum-computing/quantum-natural-language-processing-748d6f27b31d

Ends and Coends

Definition 26. Given a pair of bifunctors $F, G : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{D}$, a **dinatural transformation** is defined as follows:



Definition 28. Given a fixed bifunctor $F : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{D}$, we define the **category of wedges** $\mathcal{W}(F)$ where each object is a wedge $\Delta_d \Rightarrow F$ and given a pair of wedges $\Delta_d \Rightarrow F$ and $\Delta'_d \Rightarrow F$, we choose an arrow $f : d \to d'$ that makes the following diagram commute:



Analogously, we can define a **category of cowedges** where each object is defined as a cowedge $F \Rightarrow \Delta_d$.

Definition 29. Given a bifunctor $F : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{D}$, the **end** of F consists of a terminal wedge $\omega : \underline{\mathbf{end}}(F) \Rightarrow F$. The object $\underline{\mathbf{end}}(F) \in D$ is itself called the end. Dually, the **coend** of F is the initial object in the category of cowedges $F \Rightarrow \underline{\mathbf{coend}}(F)$, where the object $\underline{\mathbf{coend}}(F) \in \mathcal{D}$ is itself called the coend of F.

Definition 65. The **geometric realization** $|X|$ of a simplicial set X is defined as the topological space

$$|X| = \bigsqcup_{n \geqslant 0} X_n \times \Delta^n / \sim$$

where the n-simplex X_n is assumed to have a *discrete* topology (i.e., all subsets of X_n are open sets), and Δ^n denotes the *topological* n-simplex

$$\Delta^n = \{(p_0, \ldots, p_n) \in \mathbb{R}^{n+1} \mid 0 \leqslant p_i \leqslant 1, \sum_i p_i = 1$$

The spaces $\Delta^n, n \geqslant 0$ can be viewed as *cosimplicial* topological spaces with the following degeneracy and face maps:

$$\delta_i(t_0, \ldots, t_n) = (t_0, \ldots, t_{i-1}, 0, t_i, \ldots, t_n) \text{ for } 0 \leqslant i \leqslant n$$

$$\sigma_j(t_0, \ldots, t_n) = (t_0, \ldots, t_j + t_{j+1}, \ldots, t_n) \text{ for } 0 \leqslant i \leqslant n$$

Note that $\delta_i : \mathbb{R}^n \to \mathbb{R}^{n+1}$, whereas $\sigma_j : \mathbb{R}^n \to \mathbb{R}^{n-1}$.

The equivalence relation $\sim$ above that defines the quotient space is given as:

$$(d_i(x), (t_0, \ldots, t_n)) \sim (x, \delta_i(t_0, \ldots, t_n))$$

$$(s_j(x), (t_0, \ldots, t_n)) \sim (x, \sigma_j(t_0, \ldots, t_n))$$

Topological Embeddings as Coends

We now bring in the perspective that topological embeddings can be interpreted as coends as well. Consider the functor

$$F : \Delta^o \times \Delta \to \text{Top}$$

where

$$F([n], [m]) = X_n \times \Delta^m$$

where F acts *contravariantly* as a functor from Δ to $\mathbf{Sets}$ mapping $[n] \mapsto X_n$, and *covariantly* mapping $[m] \mapsto \Delta^m$ as a functor from Δ to the category Top of topological spaces.

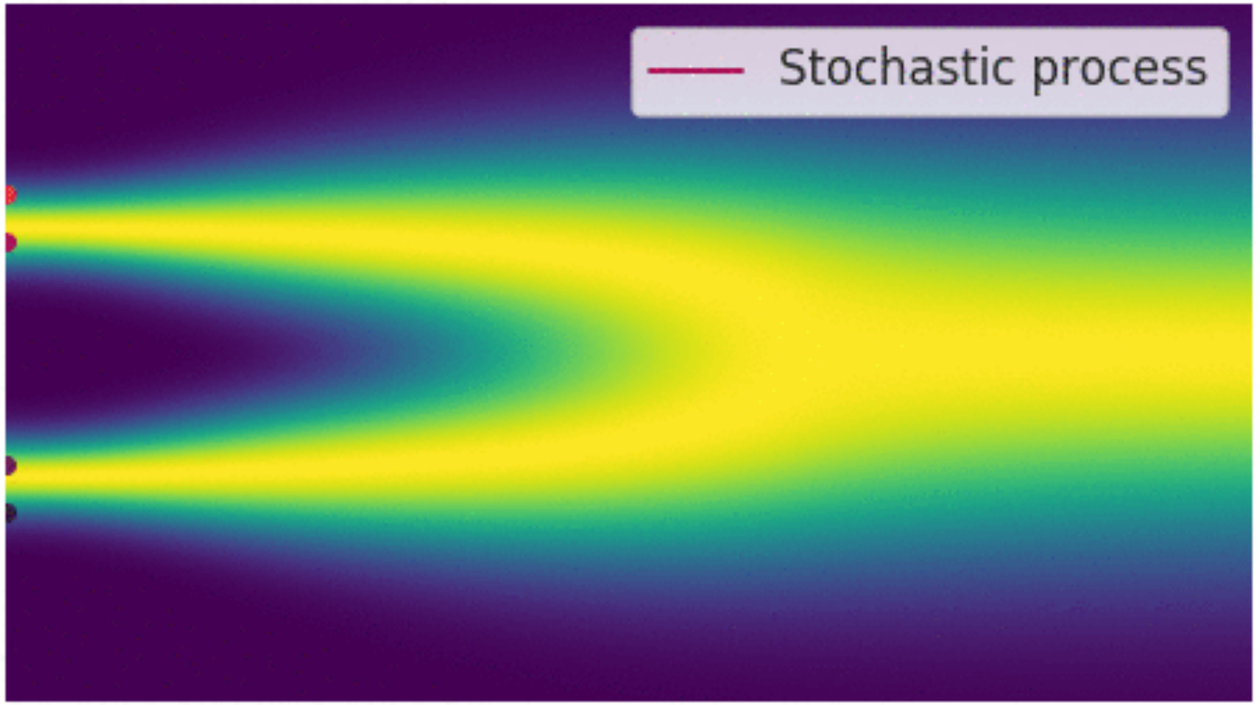
The “Geometric” Transformer Model

$$\int^{n} (\mathrm{Transformer}_{\bullet} n) \cdot \Delta n$$

Intuition: Construct a simplicial set of of Transformers by composing sequences of length n

Embed them in a Kan complex

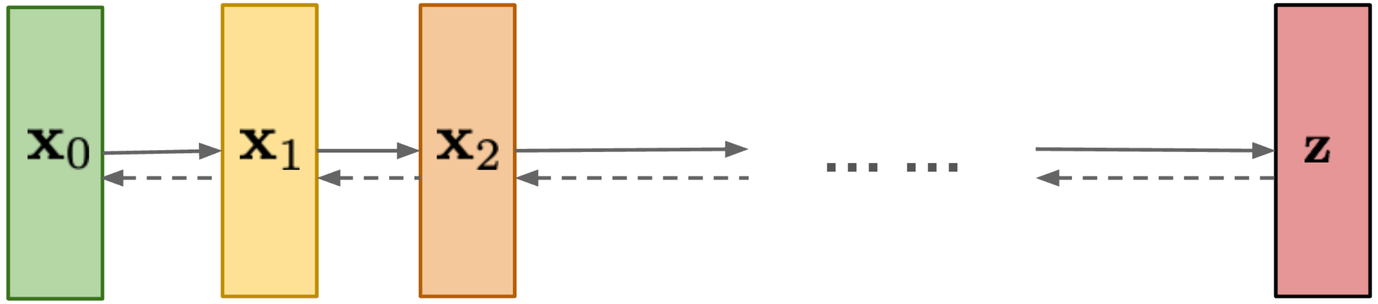
Diffusion Process and Kan Complexe



https://yang-song.net/blog/2021/score/

Generative AI and Kan Complexes

Diffusion models:
Gradually add Gaussian noise and then reverse



Every morphism

invertible!

Figure Source: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Summary

- In these three lectures, we constructed a (higher-order) category theory of generative AI, named GAIA
- Our goal was primary theoretical: we want to illustrate how category theory can give deep insight into hard practical problems
- Implementing GAIA is a problem for future work!
- Read my book drafts (continually updated) on my UMass web page

We need birds and frogs working together — Freeman Dyson