DeepMind

Graph Neural Networks in Computational Biology

(a Personal Perspective)

Petar Veličković

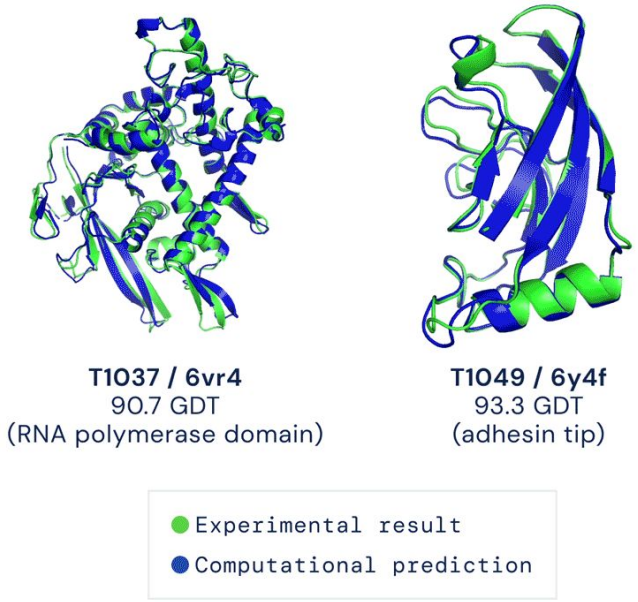
Computational Biology Society Seminar
Imperial College London
19 April 2021

DeepMind

In this talk:
Graph neural networks for biological data

What this talk is *not*!

What this talk is *not*! 🙈



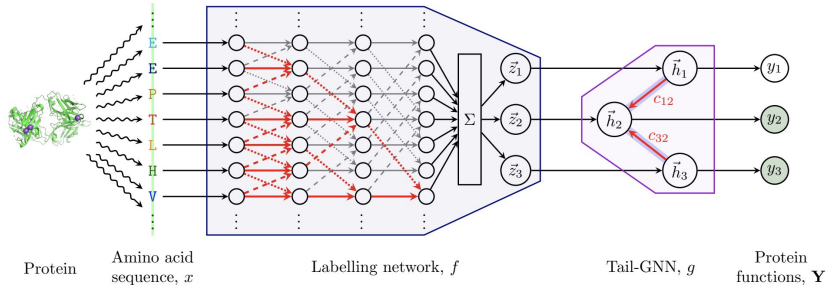
AlphaFold: a solution to a 50-year-old grand challenge in biology

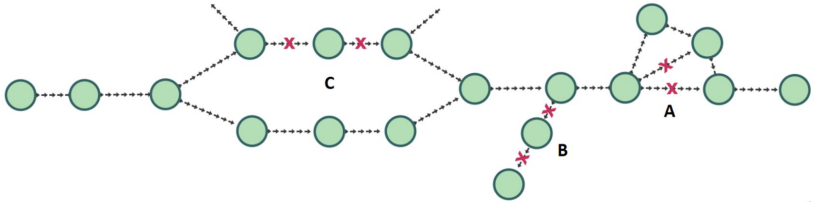
For more on AlphaFold, see:

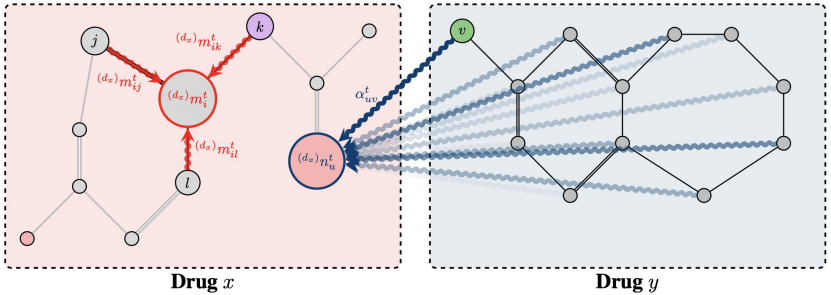
https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology

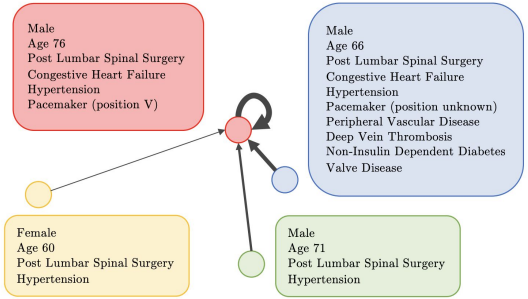
What this talk *is*

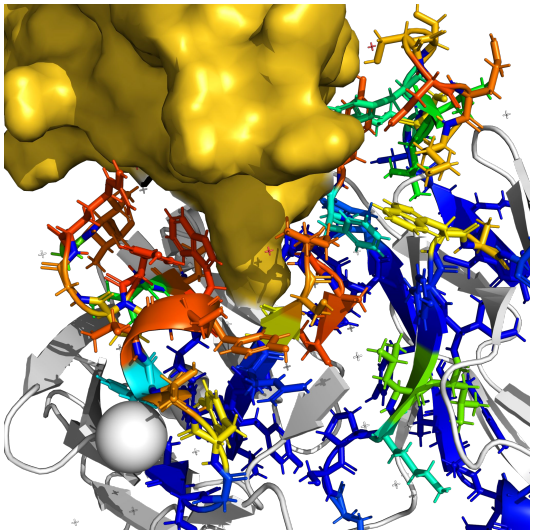
- Hopefully an exciting field from *many* angles :)
  - Molecular interactions
  - Protein function prediction
  - Genome assembly
  - Computational neuroscience
  - Electronic Health Records

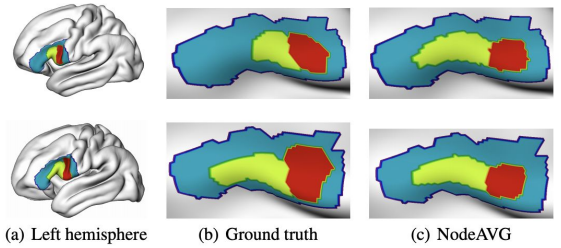












What this talk *is*

- Hopefully an exciting field from *many* angles :)
  - Molecular interactions
  - Protein function prediction
  - Genome assembly
  - Computational neuroscience
  - Electronic Health Records

- More broadly...
  - Personal perspective on this rich, interdisciplinary field
  - For ML audience: **you can do it**!
    - + a **blueprint** for approaching the area
  - For Bio audience: hopefully a useful **computational tool**

  (for both: interdisciplinary collaboration can work wonders!)

Let's start at the beginning

- Born in Belgrade (Serbia) in the 1990s
  - Family members worked for local representatives of "big pharma" (Merck)

Let's start at the beginning

- Born in Belgrade (Serbia) in the 1990s
  - Family members worked for local representatives of "big pharma" (Merck)

- Gradually increasing interest towards **computer science** – especially classical algorithms

Let's start at the beginning

- Born in Belgrade (Serbia) in the 1990s
  - Family members worked for local representatives of "big pharma" (Merck)

- Gradually increasing interest towards **computer science** – especially classical algorithms

- Developed strong interest in biology in high school (primarily thanks to **Branka Dobrković**)

I'm a biologist, and Petar already has high experience in computer science. To me this combination seems like an ideal link for very attractive scientific disciplines in the world – bioinformatics and similar. It seems to me like this connection of natural sciences with computer science would be the perfect choice for him.

Let's start at the beginning

- Born in Belgrade (Serbia) in the 1990s
  - Family members worked for local representatives of "big pharma" (Merck)
- Gradually increasing interest towards **computer science** – especially classical algorithms
- Developed strong interest in biology in high school (primarily thanks to **Branka Dobrković**)
- Computer Science at Cambridge (2012--15)
  - Lost nearly all contact with biology
- Reached out to Prof Pietro Liò for my final-year project
  - Realised that bioinformatics is **brimming** with classical algorithms
  - Pietro suggested a project in *machine learning*, however...
  - The rest is history (i.e. this talk)

Before GNNs...

- I started my PhD in 2016, with a paper classifying **breast cancer**

- Officially I was a “Research Assistant in Computational Biology”
  - But **no formal training** in biology!
  - Luckily, the field is remarkably accessible and full of interesting problems to solve
  - It was **very** helpful to talk to domain experts and understand the “burning questions”

- Fruitful collaborations lead to **Parapred** (Bioinformatics) and **ChronoMID** (PLOS ONE)
  - Carefully crafted machine learning solutions to problems posed by domain experts

Parapred: antibody paratope prediction using convolutional and recurrent neural networks FREE

Edgar Liberis ✉, Petar Veličković, Pietro Sormanni ✉, Michele Vendruscolo, Pietro Liò

Bioinformatics, Volume 34, Issue 17, 01 September 2018, Pages 2944–2950,
https://doi.org/10.1093/bioinformatics/bty305

Published: 16 April 2018 **Article history** ▾

ChronoMID—Cross-modal neural networks for 3-D temporal medical imaging data

Alexander G. Rakowski, Petar Veličković, Enrico Dall’Ara ✉, Pietro Liò

Published: February 21, 2020 • https://doi.org/10.1371/journal.pone.0228962

Before GNNs...

- I started my PhD in 2016, with a paper classifying **breast cancer**

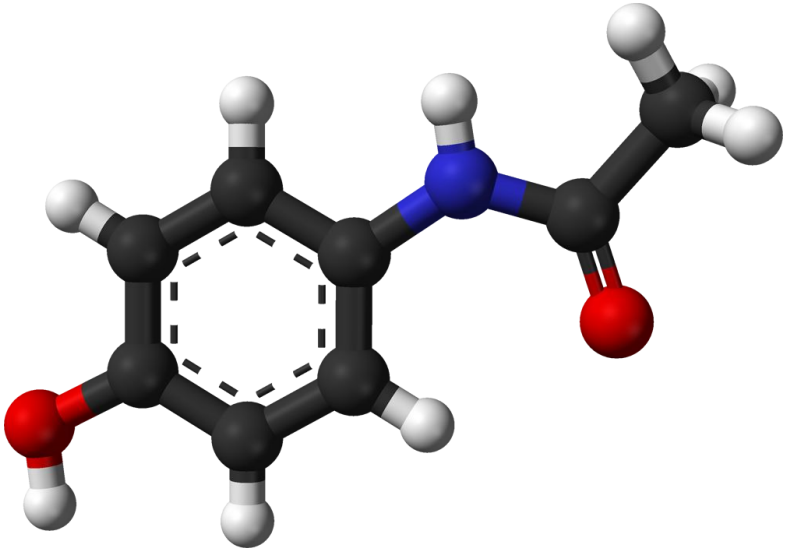
- Officially I was a “Research Assistant in Computational Biology”
  - But **no formal training** in biology!
  - Luckily, the field is remarkably accessible and full of interesting problems to solve
  - It was **very** helpful to talk to domain experts and understand the “burning questions”

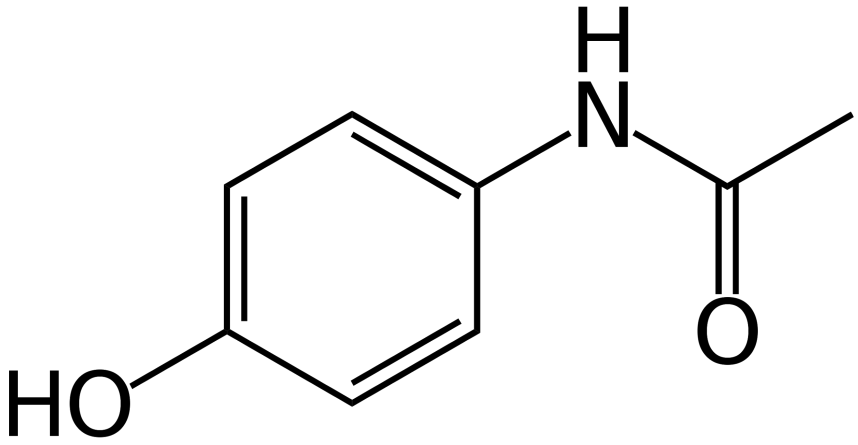
- Fruitful collaborations lead to **Parapred** (Bioinformatics) and **ChronoMID** (PLOS ONE)
  - Carefully crafted machine learning solutions to problems posed by domain experts

- “Game changing” moment in 2017, when I discovered **graph representation learning**
  - Why should you care?

Molecules are graphs!

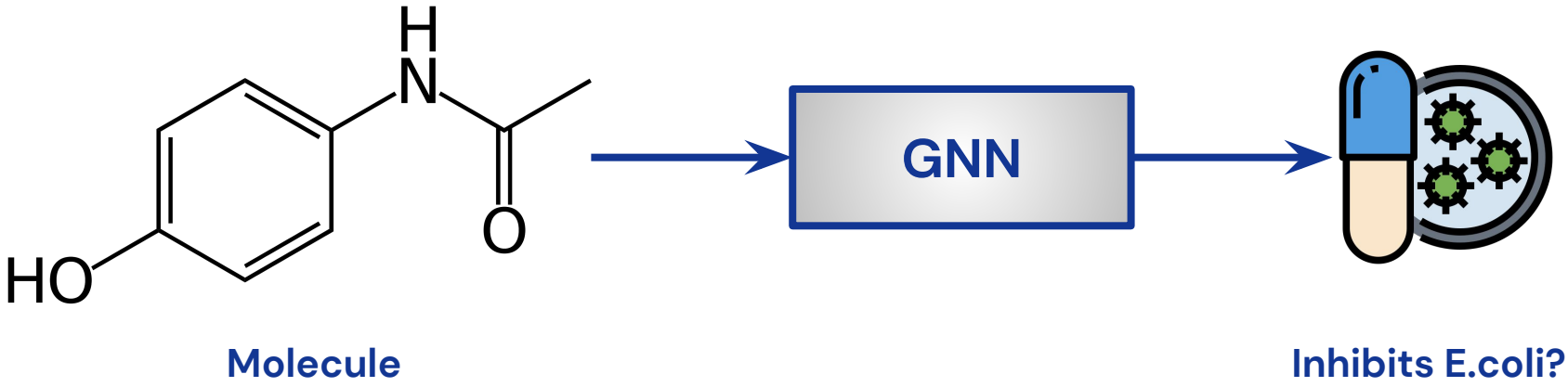
- A very natural way to represent molecules is as a **graph**
  - **Atoms** as nodes, **bonds** as edges
  - Features such as **atom type**, **charge**, **bond type**...





GNNs for molecule classification

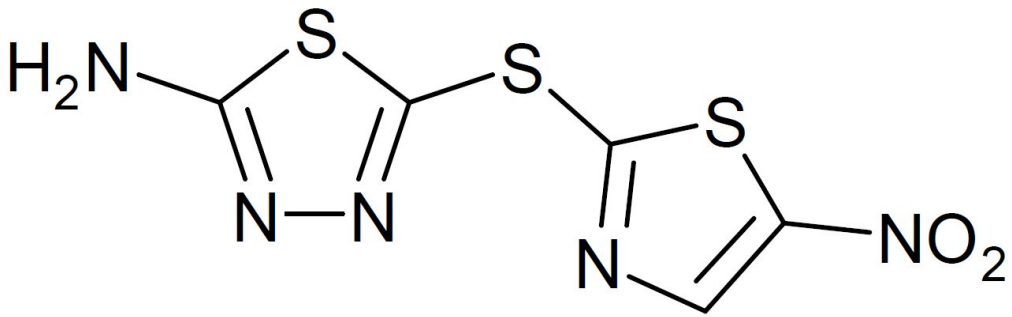
- Interesting task to predict is, for example, whether the molecule is a potent **drug**.
  - Can do *binary classification* on whether the drug will inhibit certain bacteria. (*E.coli*)
  - Train on a **curated dataset** for compounds where response is known.



Follow-up study

- Once trained, the model can be applied to *any* molecule.
  - Execute on a large dataset of known candidate molecules.
  - Select the *~top-100* candidates from your GNN model.
  - Have chemists thoroughly investigate those (after some additional filtering).

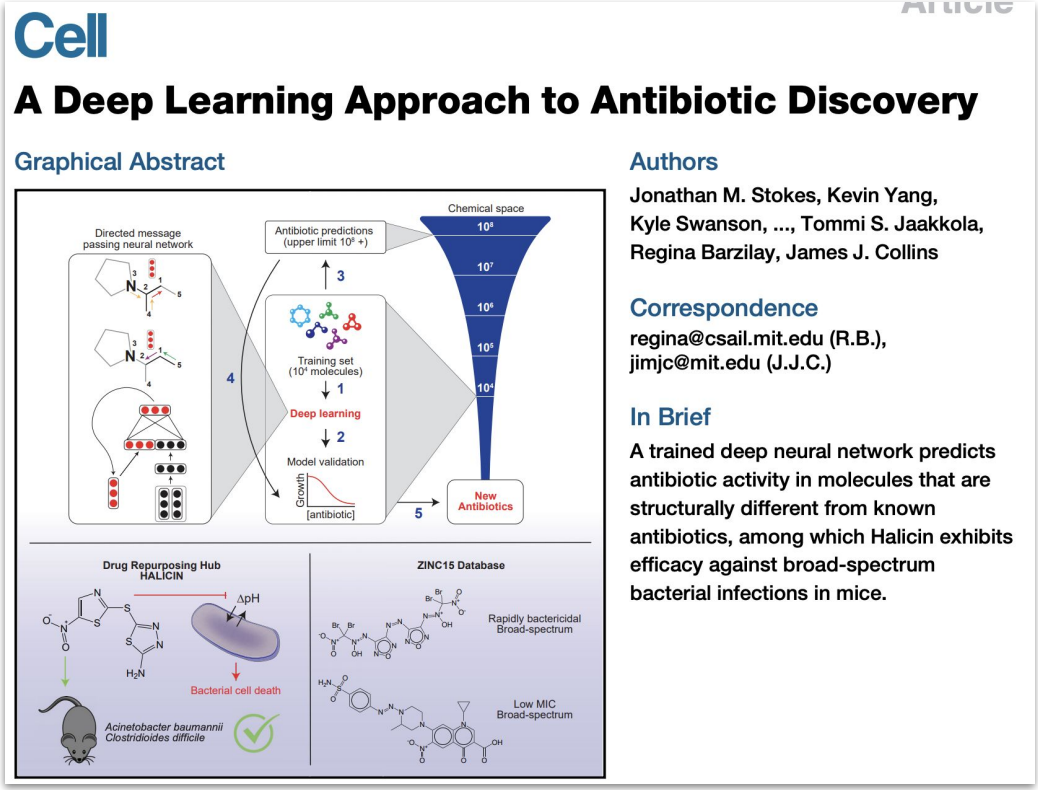
- Discover a previously overlooked compound that is a **highly potent** antibiotic!



Halicin

…Achieve wide acclaim!

Arguably the most popularised **success story** of graph neural networks to date!



Cell

A Deep Learning Approach to Antibiotic Discovery

Graphical Abstract

Authors

Jonathan M. Stokes, Kevin Yang, Kyle Swanson, ..., Tommi S. Jaakkola, Regina Barzilay, James J. Collins

Correspondence

regina@csail.mit.edu (R.B.), jimjc@mit.edu (J.J.C.)

In Brief

A trained deep neural network predicts antibiotic activity in molecules that are structurally different from known antibiotics, among which Halicin exhibits efficacy against broad-spectrum bacterial infections in mice.

(Stokes *et al.*, Cell'20)

…Achieve wide acclaim!

Arguably the most popularised **success story** of graph neural networks to date!

Cell

nature

Subscribe

NEWS · 20 FEBRUARY 2020

Powerful antibiotics discovered using AI

Machine learning spots molecules that work even against ‘untreatable’ strains of bacteria.

(Stokes *et al.*, Cell’20)

…Achieve wide acclaim!

Arguably the most popula


nature

NEWS · 20 FEBRUARY 2020

Powerful an

Machine learning spot
bacteria.

(Stokes *et al.*, Cell'20)


FINANCIAL TIMES

S COMPANIES TECH MARKETS GRAPHICS OPINION WORK & CAREERS LIFE & ARTS HOW TO SPEND IT

CORONAVIRUS **BUSINESS UPDATE**
Get 30 days' complimentary access to our Coronavirus Business Update newsletter

l intelligence

Robotics

'Death of the office' homeworking claims exaggerated

Anti-social robots harr increase social distanci

Artificial intelligence + Add to myFT

AI discovers antibiotics to treat drug-resistant diseases

Machine learning uncovers potent new drug able to kill 35 powerful bacteria

…Achieve wide acclaim!

Argua

BBC Sign in News Sport Reel Worklife Travel Future

NEWS

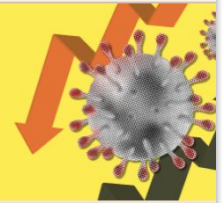
Home Video World UK Business Tech Science Stories Entertainment & Arts

BBC WORKLIFE

Our new guide for getting ahead

TIMES

S HOW TO SPEND IT



Anti-social robots harr
increase social distanci

NEWS

Po

Mach

bacte

g-resistant

Scientists discover powerful antibiotic using AI

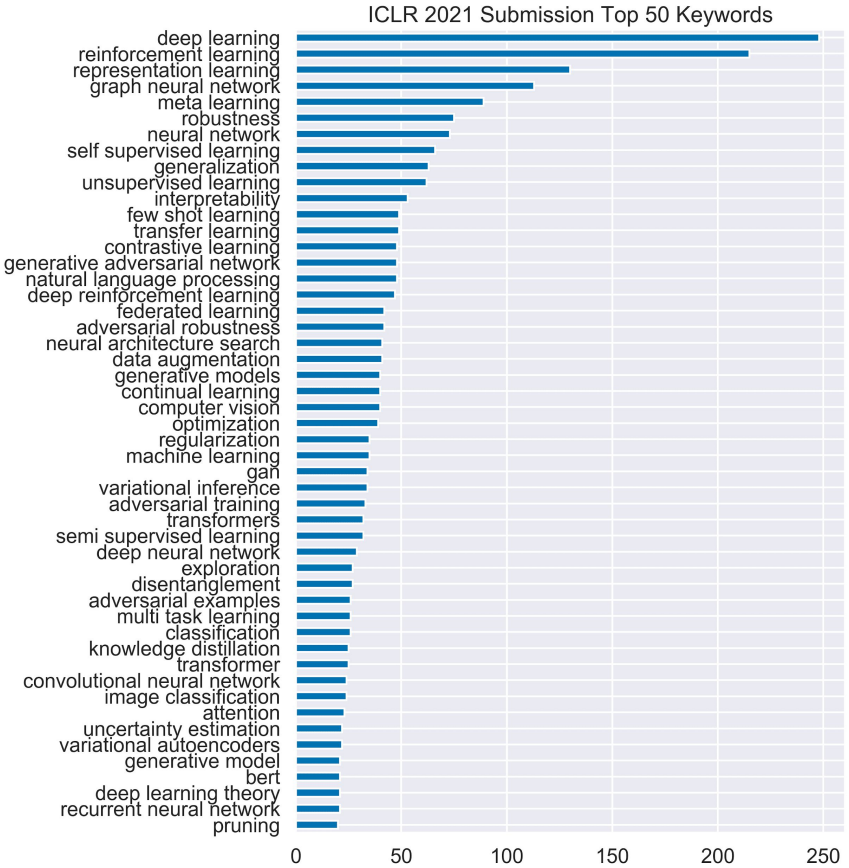
21 February 2020

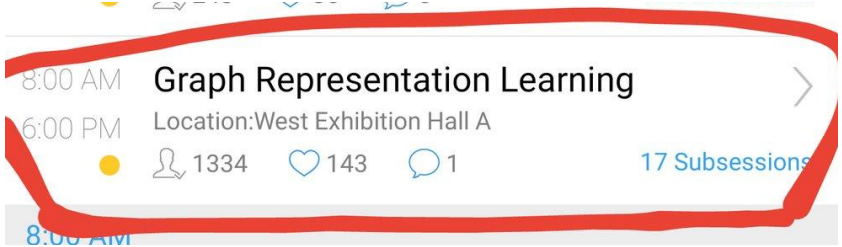
Share

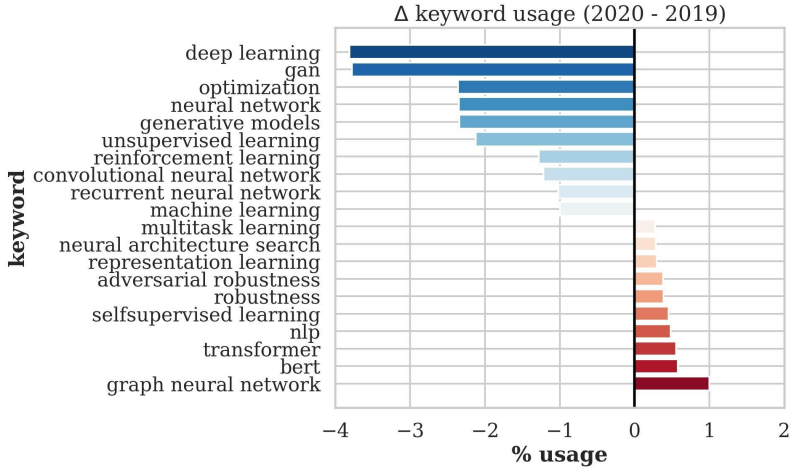
(Stokes *et al.*, Cell'20)

Machine learning uncovers potent new drug able to kill 35 powerful bacteria

GNNs are a **very hot** research topic

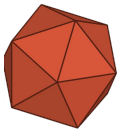






GNNs are currently experiencing their "ImageNet" moment

Rich ecosystem of libraries



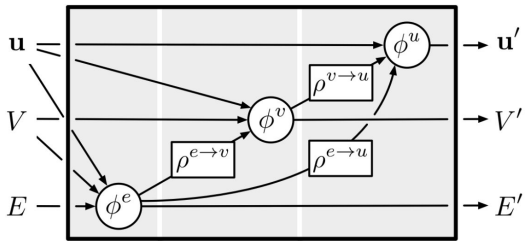
PyTorch geometric

github.com/rusty1s/pytorch_geometric

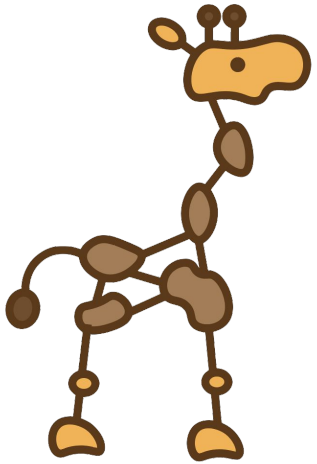
Spektral

graphneural.network

dgl.ai



github.com/deepmind/graph_nets



github.com/deepmind/jraph

Rich ecosystem of datasets

ogb.stanford.edu

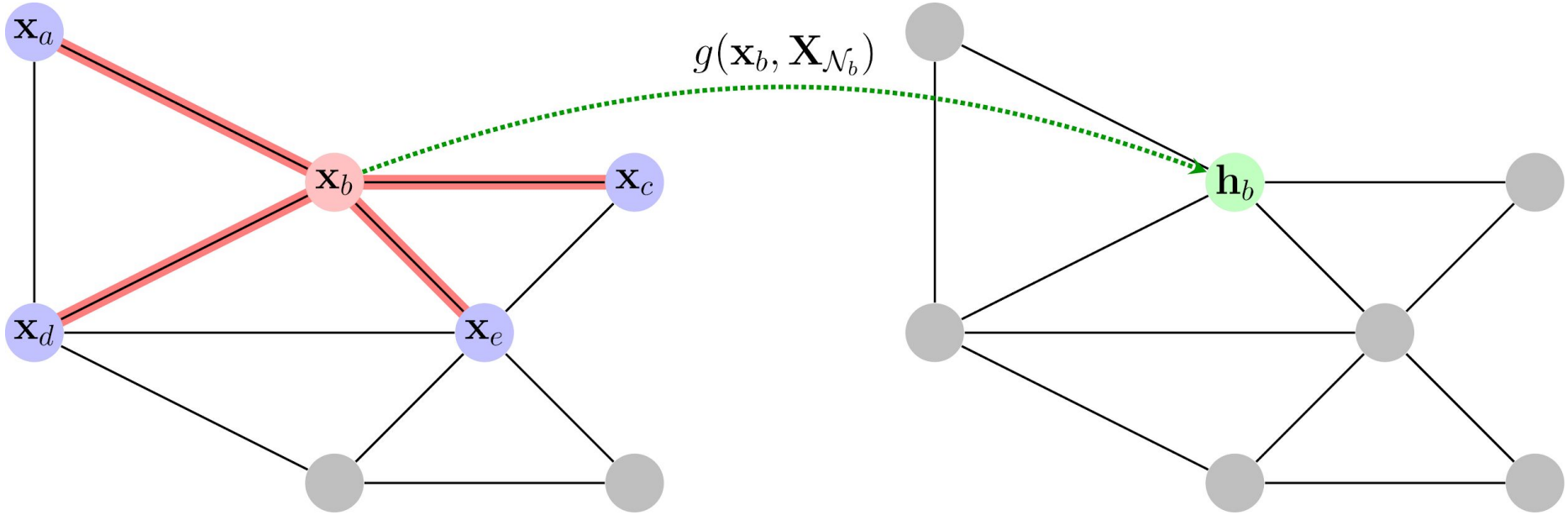
https://pytorch-geometric.readthedocs.io/en/latest/modules/datasets.html

graphlearning.io

Benchmarking Graph Neural Networks

github.com/graphdeeplearning/benchmarking-gnns

How to **process** the graph?



$$\mathbf{X}_{\mathcal{N}_b} = \{\{\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c, \mathbf{x}_d, \mathbf{x}_e\}\}$$

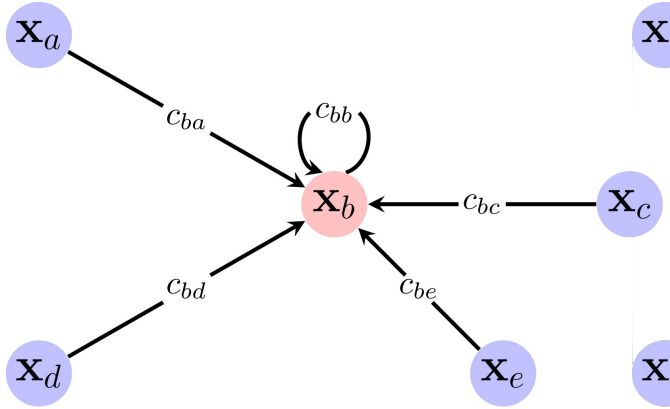
What's in a GNN layer?

- We construct useful functions over graphs, f, by shared application of a local permutation-invariant function $g(\mathbf{x}_i, \mathbf{X}_{Ni})$.
  - We often refer to f as "GNN layer", g as "diffusion", "propagation", "message passing"

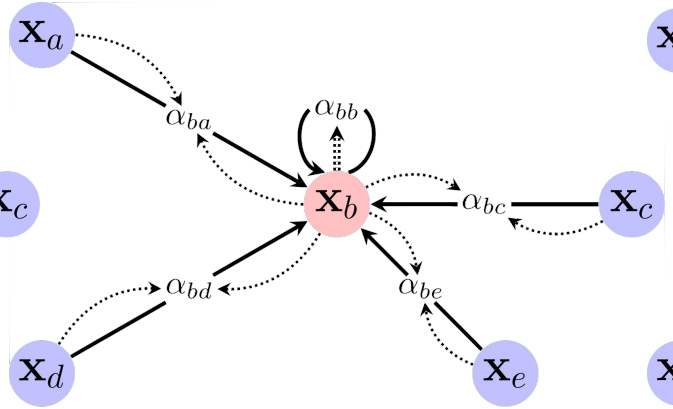
- We will take a quick look at ways in which we can actually concretely **define** g.
  - **Very intense** area of research!

- Fortunately, *almost all* proposed layers can be classified as one of three ***spatial*** "*flavours*".

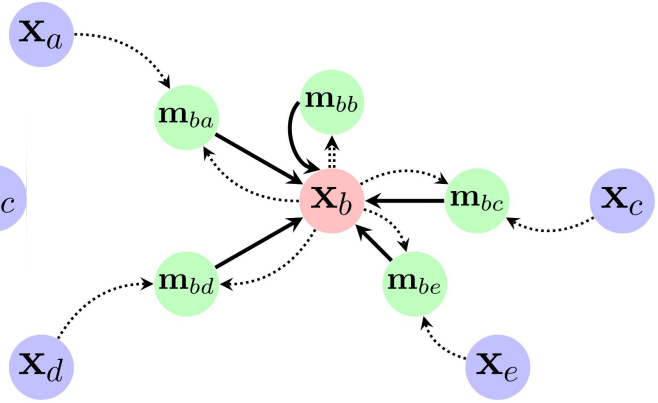
The three “flavours” of GNN layers



Convolutional



Attentional



Message-passing

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} c_{ij}\psi(\mathbf{x}_j)\right)$$

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} a(\mathbf{x}_i, \mathbf{x}_j)\psi(\mathbf{x}_j)\right)$$

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} \psi(\mathbf{x}_i, \mathbf{x}_j)\right)$$

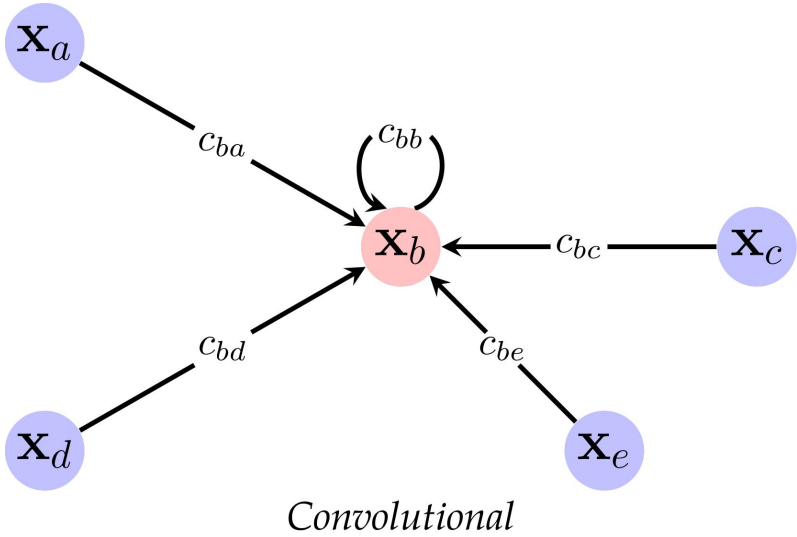
Convolutional GNN

- Features of neighbours aggregated with fixed weights, c_{ij}

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} c_{ij}\psi(\mathbf{x}_j)\right)$$

- Usually, the weights depend directly on **A**.
  - ChebyNet (Defferrard *et al.*, NeurIPS'16)
  - GCN (Kipf & Welling, ICLR'17)
  - SGC (Wu *et al.*, ICML'19)

- Useful for **homophilous** graphs and **scaling up**
  - When edges encode *label similarity*



Convolutional

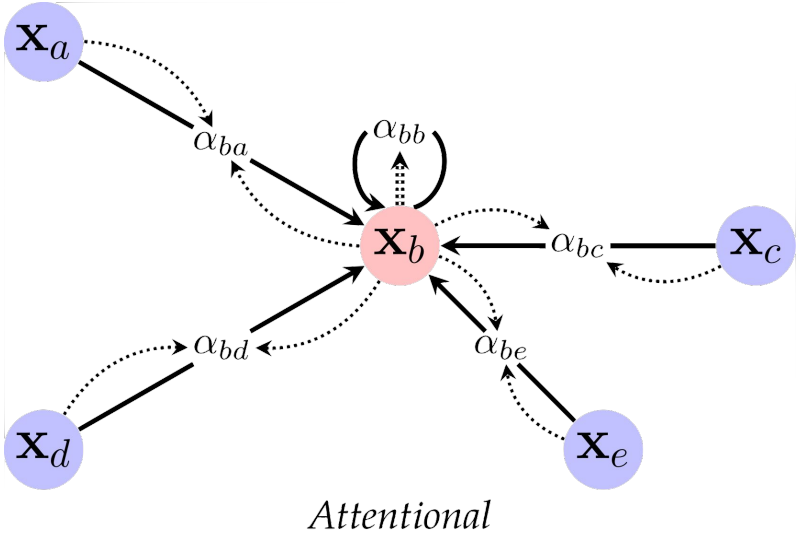
Attentional GNN

- Features of neighbours aggregated with **implicit** weights (via *attention*)

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} a(\mathbf{x}_i, \mathbf{x}_j)\psi(\mathbf{x}_j)\right)$$

- Attention weight computed as $a_{ij} = a(\mathbf{x}_i, \mathbf{x}_j)$
  - MoNet (Monti *et al.*, CVPR'17)
  - GAT (Veličković *et al.*, ICLR'18)
  - GaAN (Zhang *et al.*, UAI'18)

- Useful as "middle ground" w.r.t. **capacity** and **scale**
  - Edges need not encode homophily
  - But still compute *scalar* value in each edge



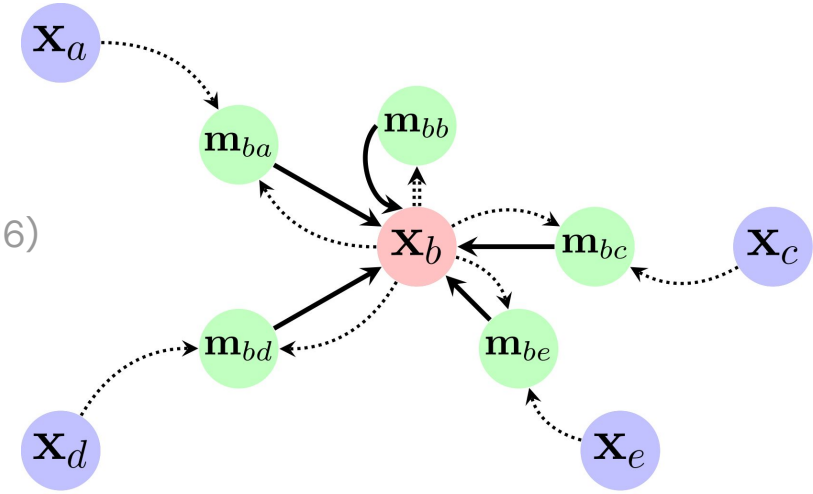
Attentional

Message-passing GNN

- Compute **arbitrary vectors** (*"messages"*) to be sent across edges

$$\mathbf{h}_i = \phi\left(\mathbf{x}_i, \bigoplus_{j \in \mathcal{N}_i} \psi(\mathbf{x}_i, \mathbf{x}_j)\right)$$

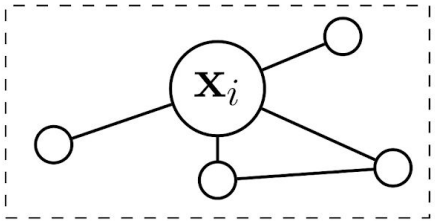
- Messages computed as $\mathbf{m}_{ij} = \psi(\mathbf{x}_i, \mathbf{x}_j)$
  - Interaction Networks (Battaglia *et al.*, NeurIPS'16)
  - MPNN (Gilmer *et al.*, ICML'17)
  - GraphNets (Battaglia *et al.*, 2018)



Message-passing

- Most **generic** GNN layer
  - May have *scalability* or *learnability* issues
  - Ideal for *computational chemistry*, *reasoning* and *simulation*

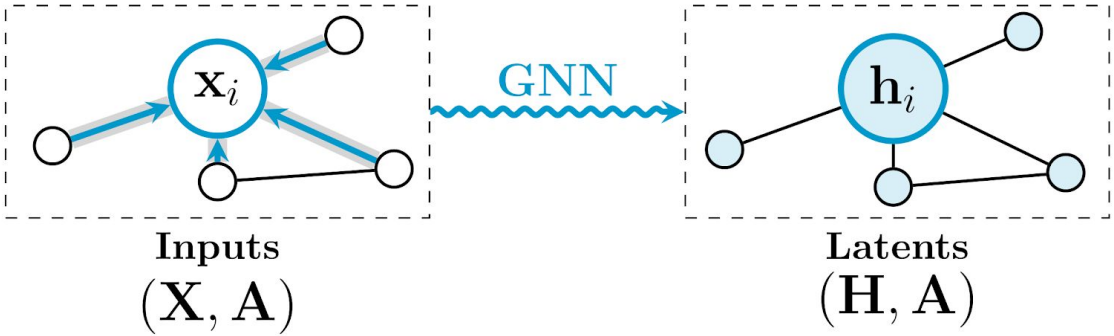
How to use GNNs?



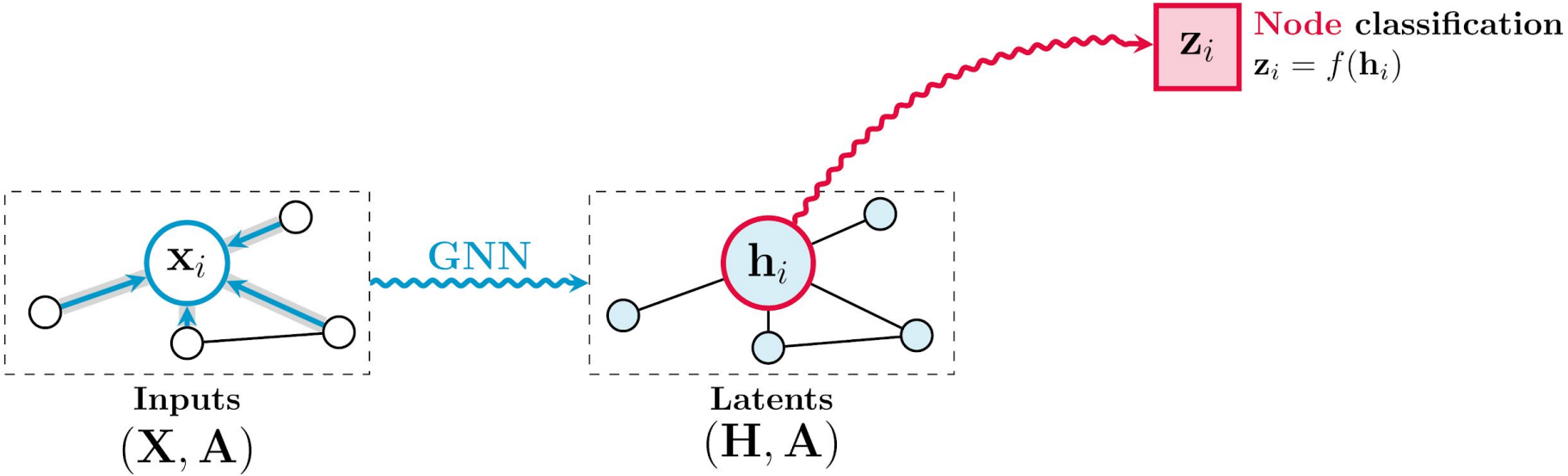
Inputs

$(\mathbf{X}, \mathbf{A})$

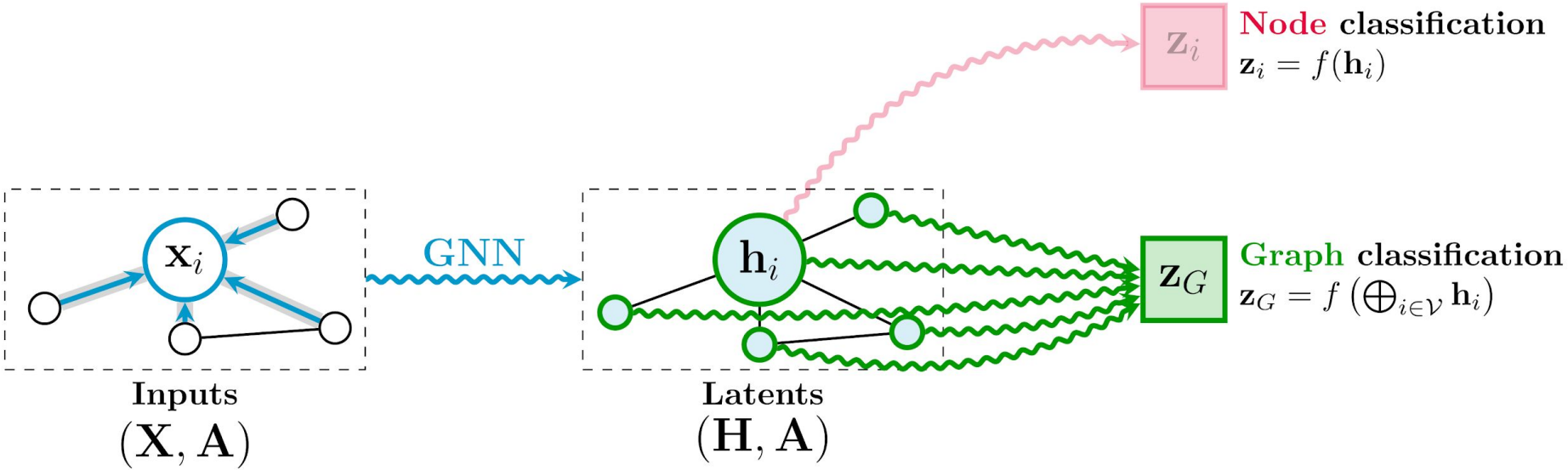
How to use GNNs?



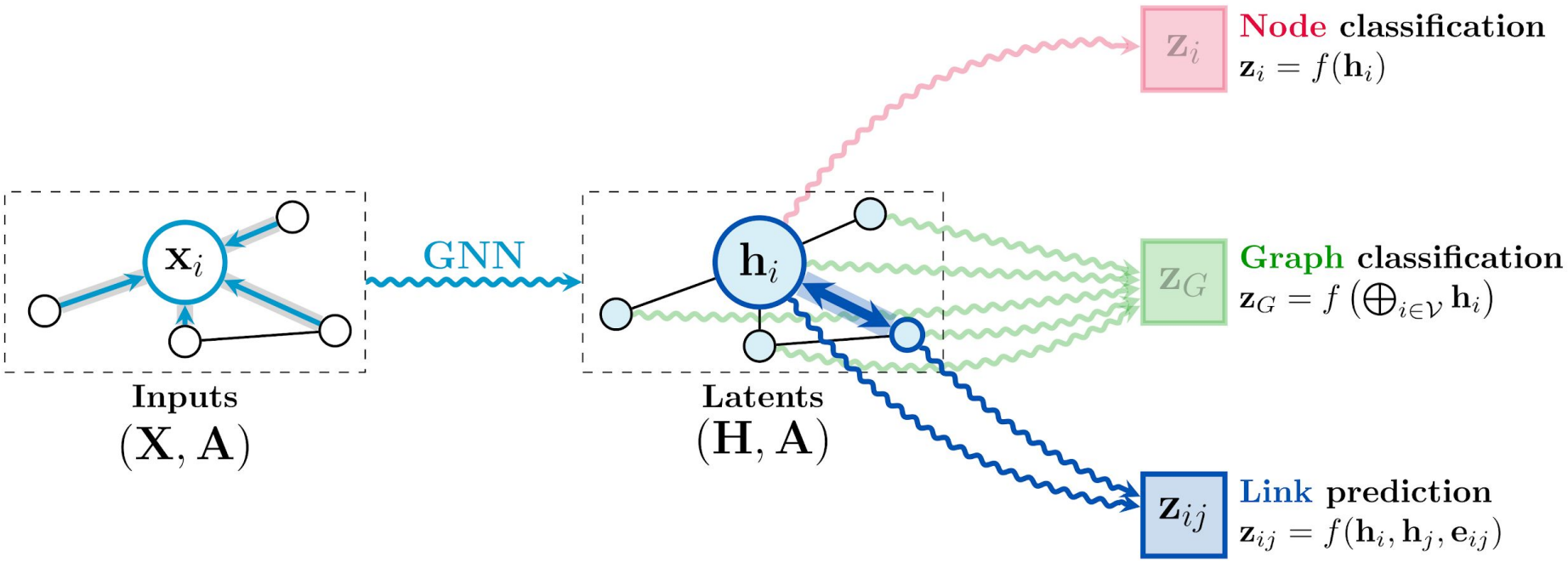
How to use GNNs?



How to use GNNs?

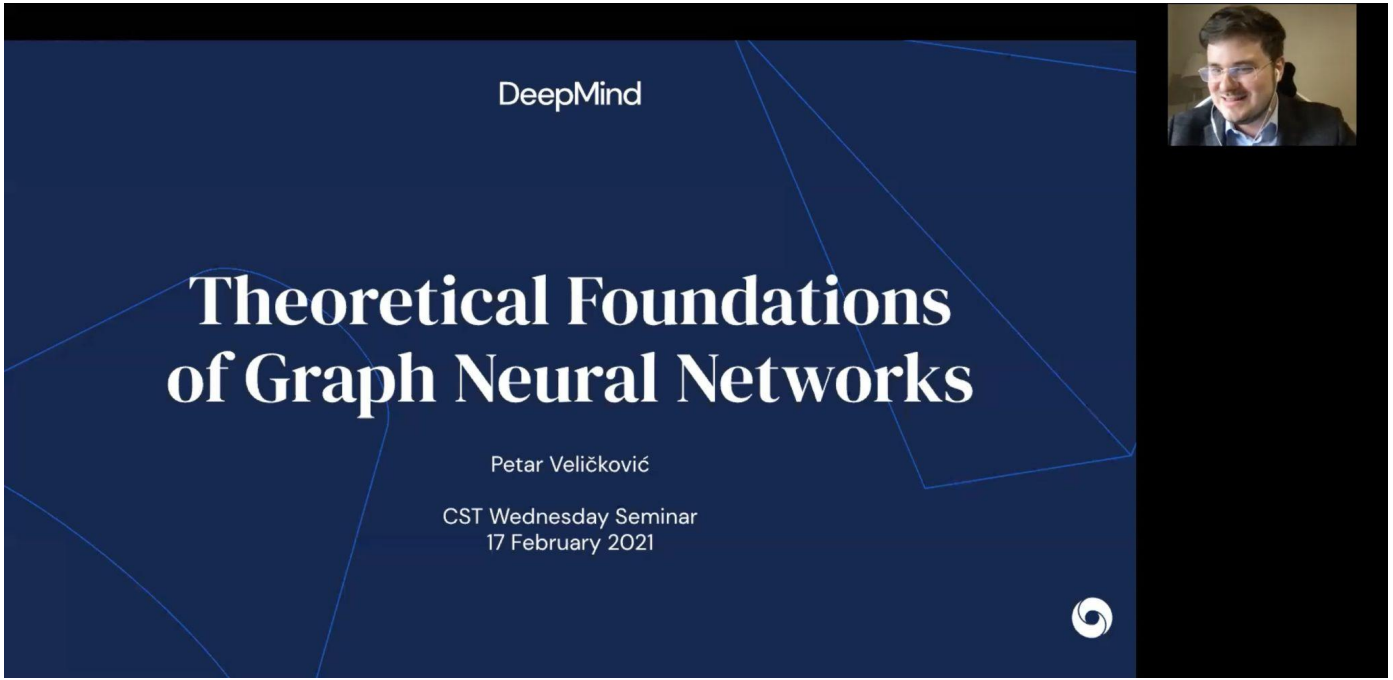


How to use GNNs?



If you'd like to know more

For (substantially!) more context, I recently gave a talk on **theoretical GNN foundations**:
https://www.youtube.com/watch?v=uF53xsT7mjc



...Back to the past 🕸

- In 2017, as part of my Mila internship we proposed Graph Attention Networks (GATs)
  - One of the first prominent examples of attentional GNN
  - They remain a popular model to this day

- It was only loosely clear that models like these could benefit my biological projects
  - We set out to find out exactly how...

DeepMind

CNNs* for Mesh-based Parcellation of the Cerebral Cortex

Guillem Cucurull, Konrad Wagstyl, Arantxa Casanova, **Petar Veličković**,
Estrid Jakobsen, Michal Drozdzal, Adriana Romero, Alan Evans and Yoshua Bengio

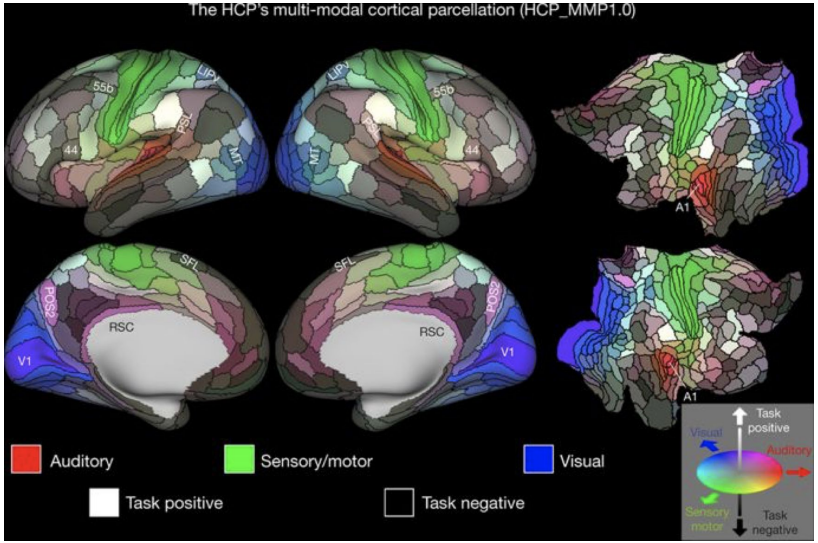
Cortex parcellation

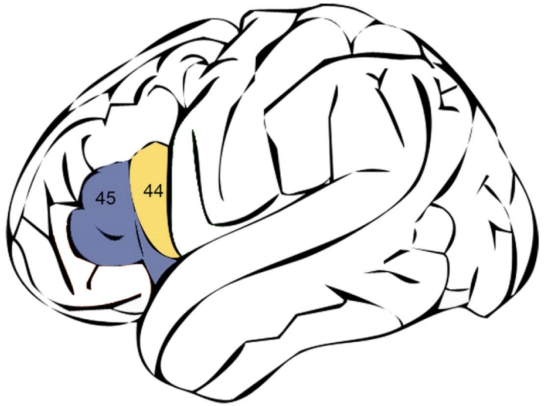
- Different areas of the cerebral cortex are involved in different cognitive processes
  - Visual processing
  - Language comprehension

- Mapping these areas helps us understand how the cortex is organised

- Our graph attention network paper was, in fact, built for this very purpose :)

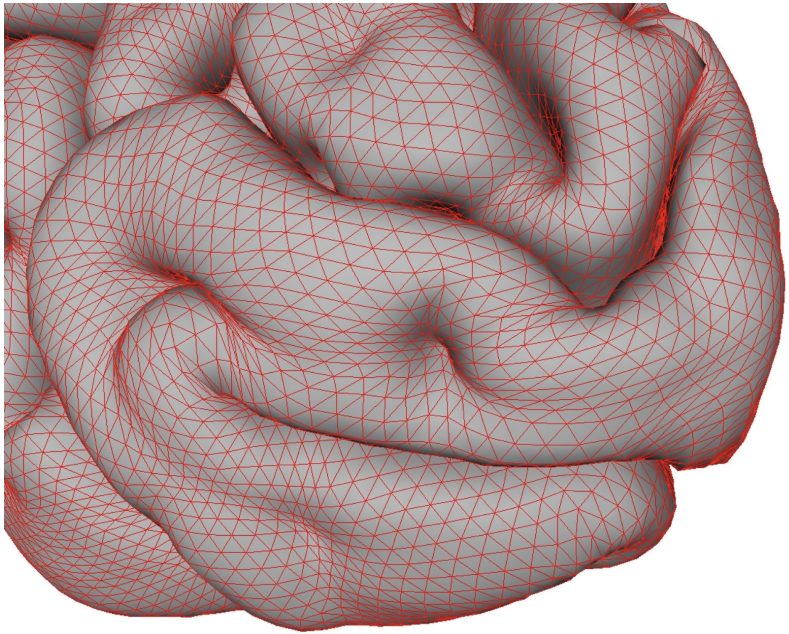
- We focus on regions 44 and 45 of **Broca's area**:



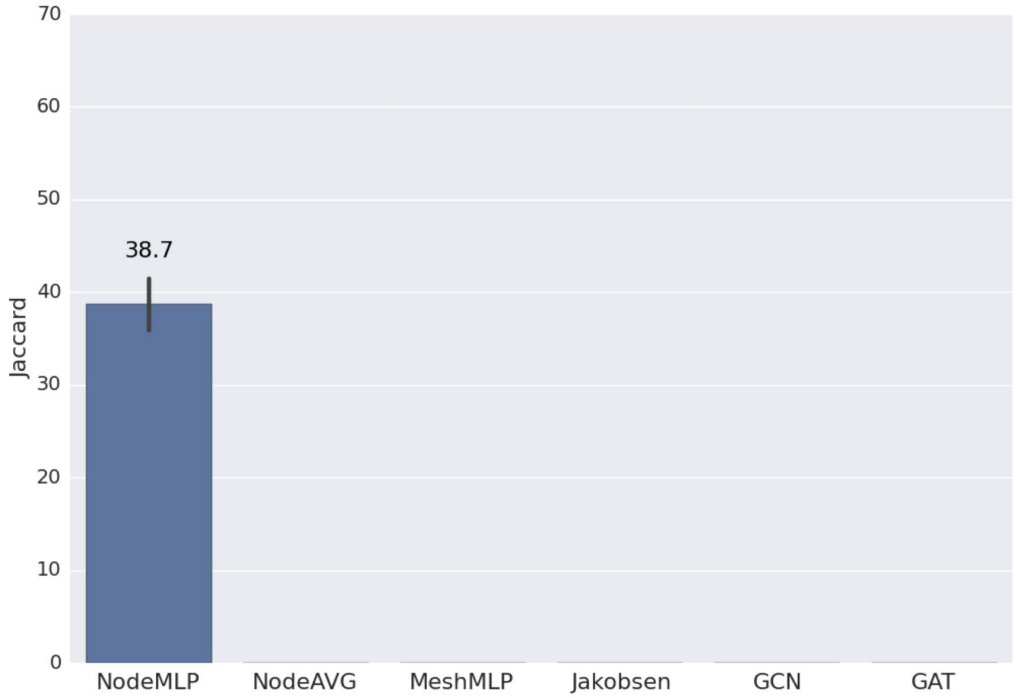


What is a cortical mesh?

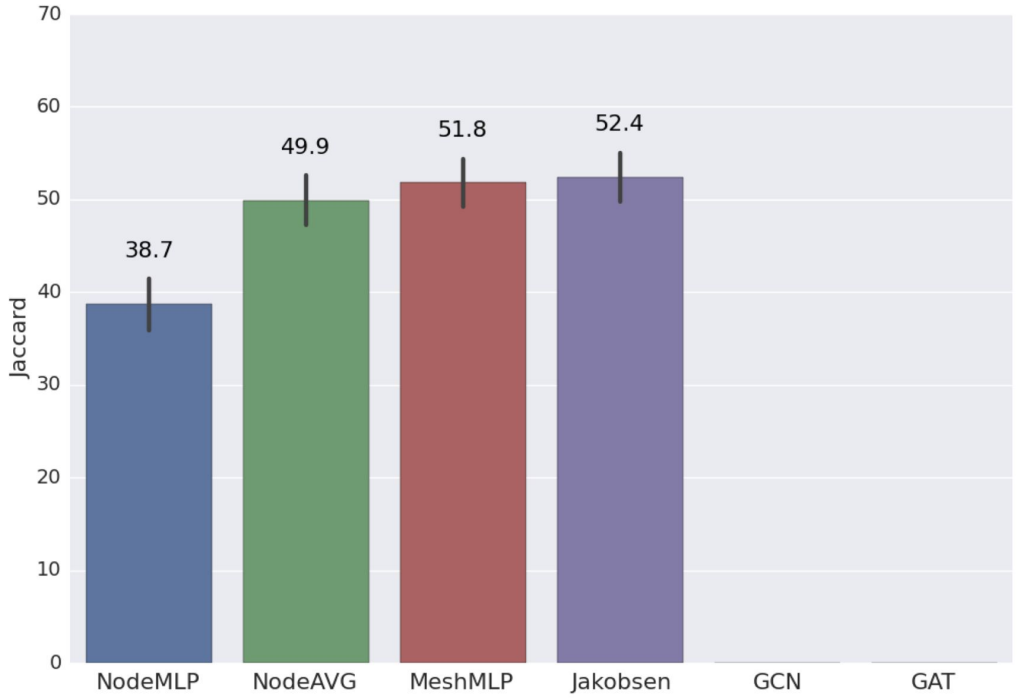
- Common coordinate system
- Can represent multiple modalities and features
- Can be used to coregister cortical surfaces between different individuals
- We can run a GNN over the nodes in the mesh!
    - Classify nodes as “44”, “45”, or “background”.



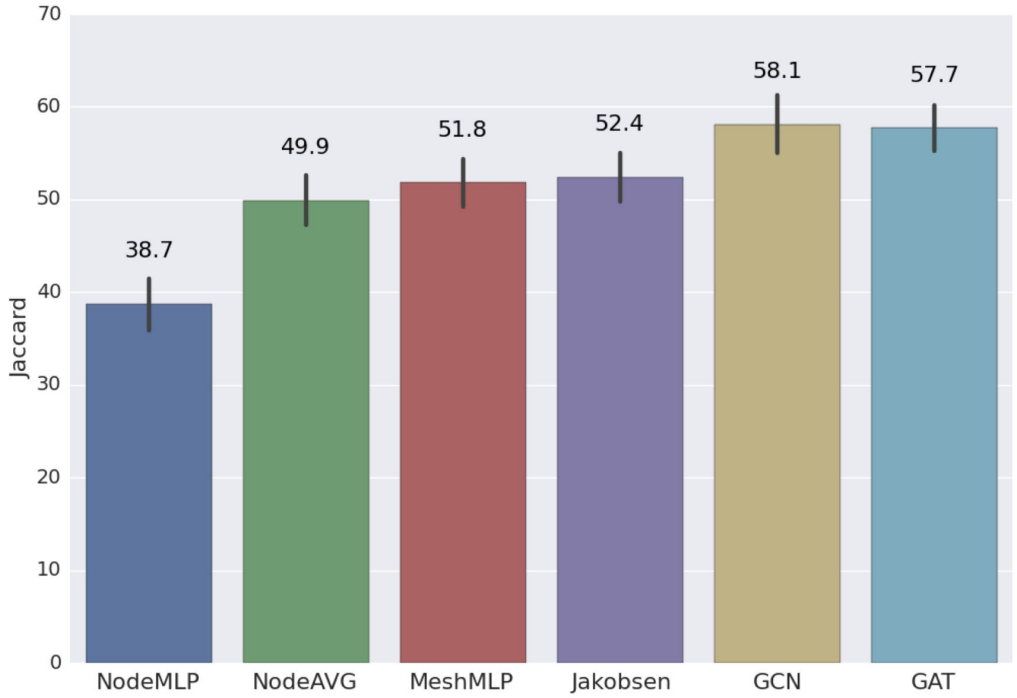
Quantitative results



Quantitative results

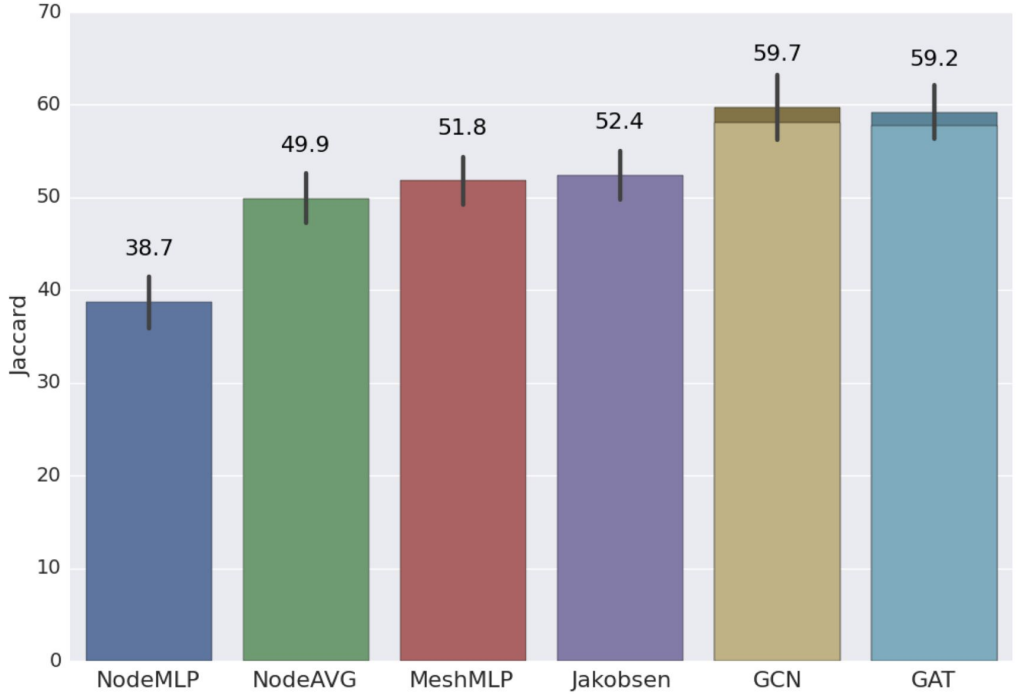


Quantitative results

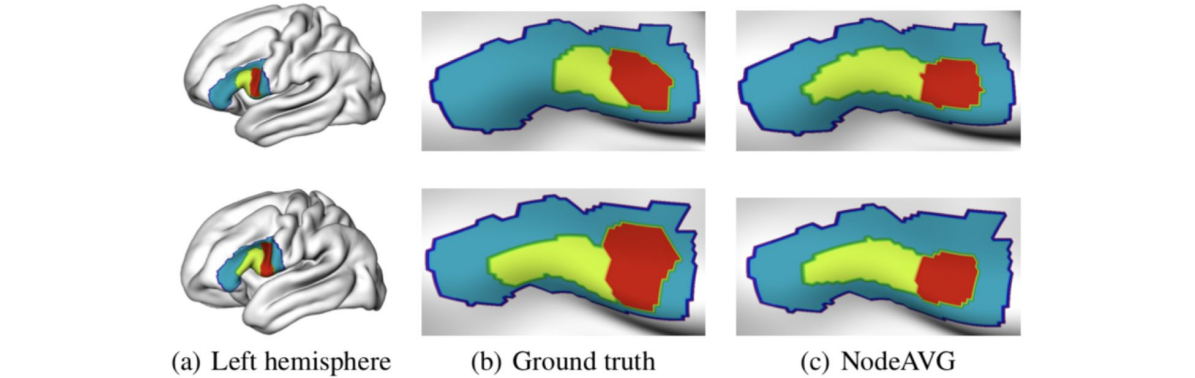


Quantitative results

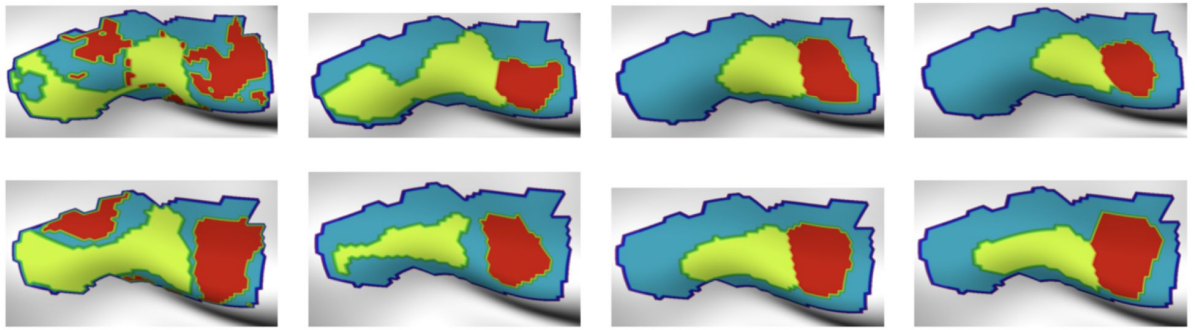
+ Node **positional** features!



Qualitative results

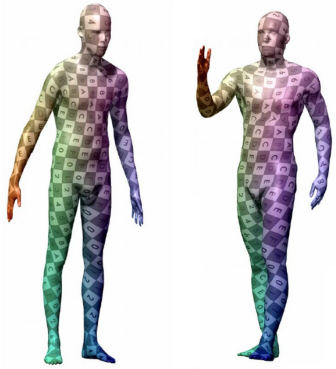


(a) Left hemisphere (b) Ground truth (c) NodeAVG



(d) NodeMLP (e) Jakobsen et al. [21] (f) GCN (g) GAT

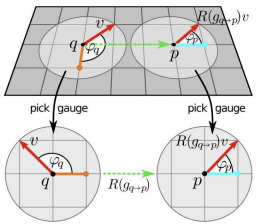
With hindsight…



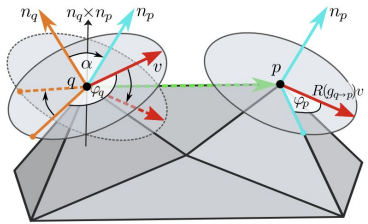
- Meshes come with *a lot* of useful geometry
  - Evident by utility of positional features
  - (Vanilla) GNNs would *discard* that information

- We now have a **wealth** of architectures that are specialising for the mesh domain!
  - Geodesic CNN (Masci *et al.*)
  - MoNet (Monti *et al.*)
  - Gauge Equivariant Mesh CNN (de Haan *et al.*)

- All of the above would make great choices for processing the brain mesh!
  - Perhaps an interesting future project? 👀
  - Reach out to me if you're curious!



(a) Parallel transport on a flat mesh.


(b) Parallel transport along an edge of a general mesh.

...Back to the past 🧠

- This project proved to me the untapped utility that GNNs can have in biological problems
  - We applied the GCN and GAT models pretty much out-of-the-box!

- Now was the time to revisit my earlier collaboration (Parapred) under this lens.

DeepMind

Attentive Cross-modal Paratope Prediction

Andreea Deac, **Petar Veličković** and Pietro Sormanni

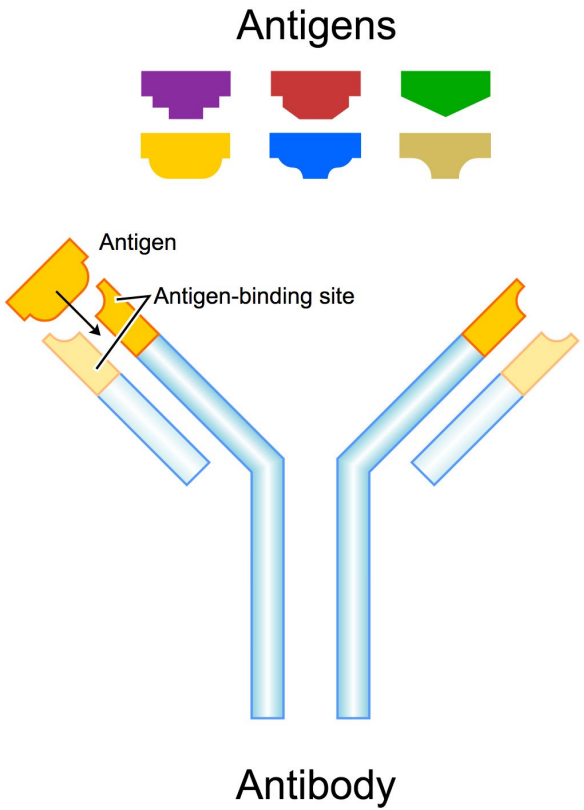
Motivation for antibody design

- Antibodies are
  - Y-shaped proteins
  - a critical part of our immune system

- They neutralise pathogenic bacteria and viruses by tagging the antigen in a “lock and key” system.

- Designing our own arbitrary antibodies would be a big step towards personalised medicine.

- (You’ve probably heard a whole lot about antibodies and antigens in the past year…)



Towards personalised medicine

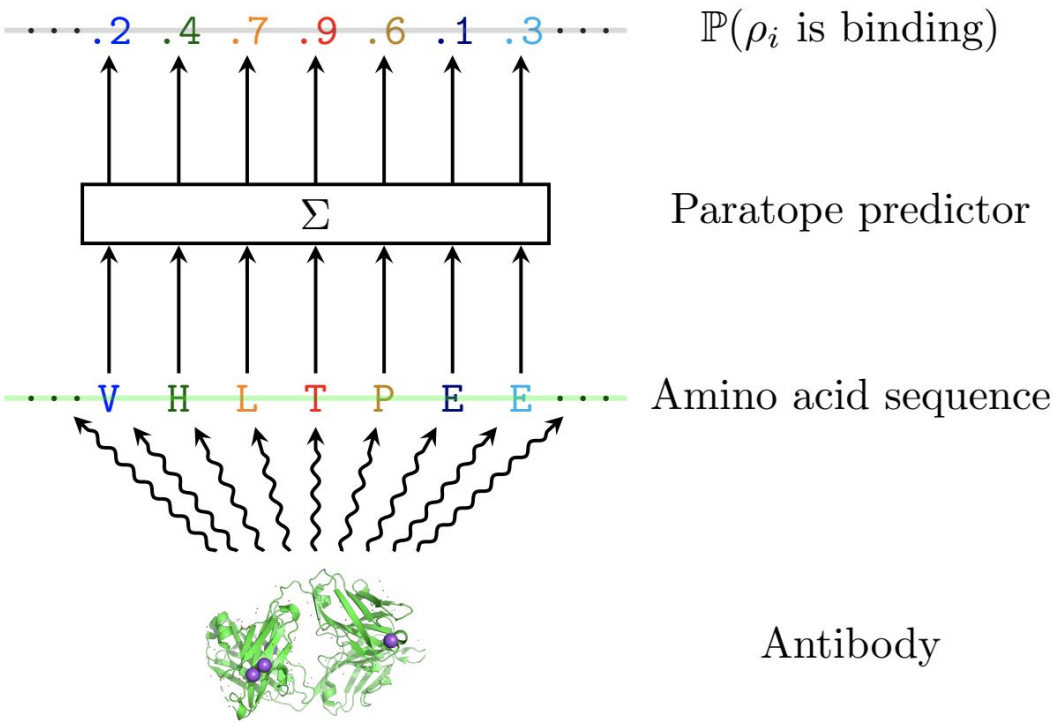
- Generating an antibody requires first predicting the specific amino acids (the **paratope**) which participate in the neutralisation of the antigen.

- **Input:** a sequence of (one-hot encoded) antibody amino acids.

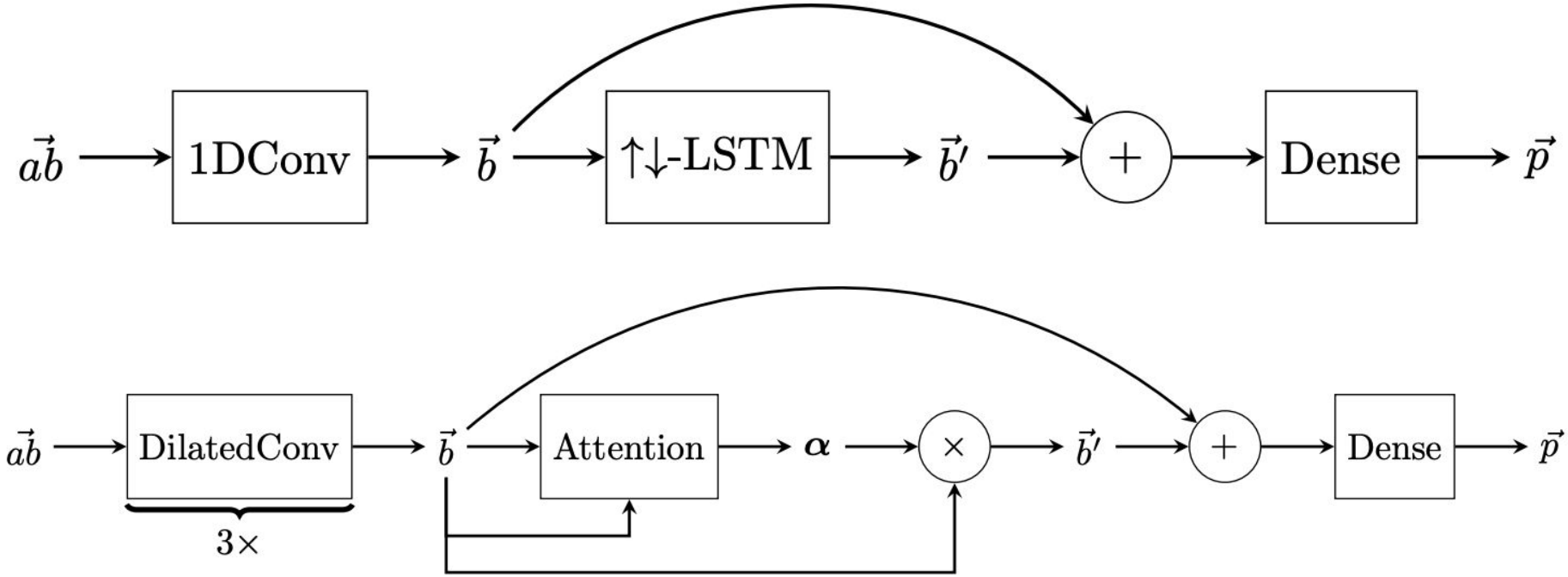
  (+ a sequence of (one-hot encoded) antigen amino acids)

- **Output:** probability for each amino acid to participate in the binding with the antigen.

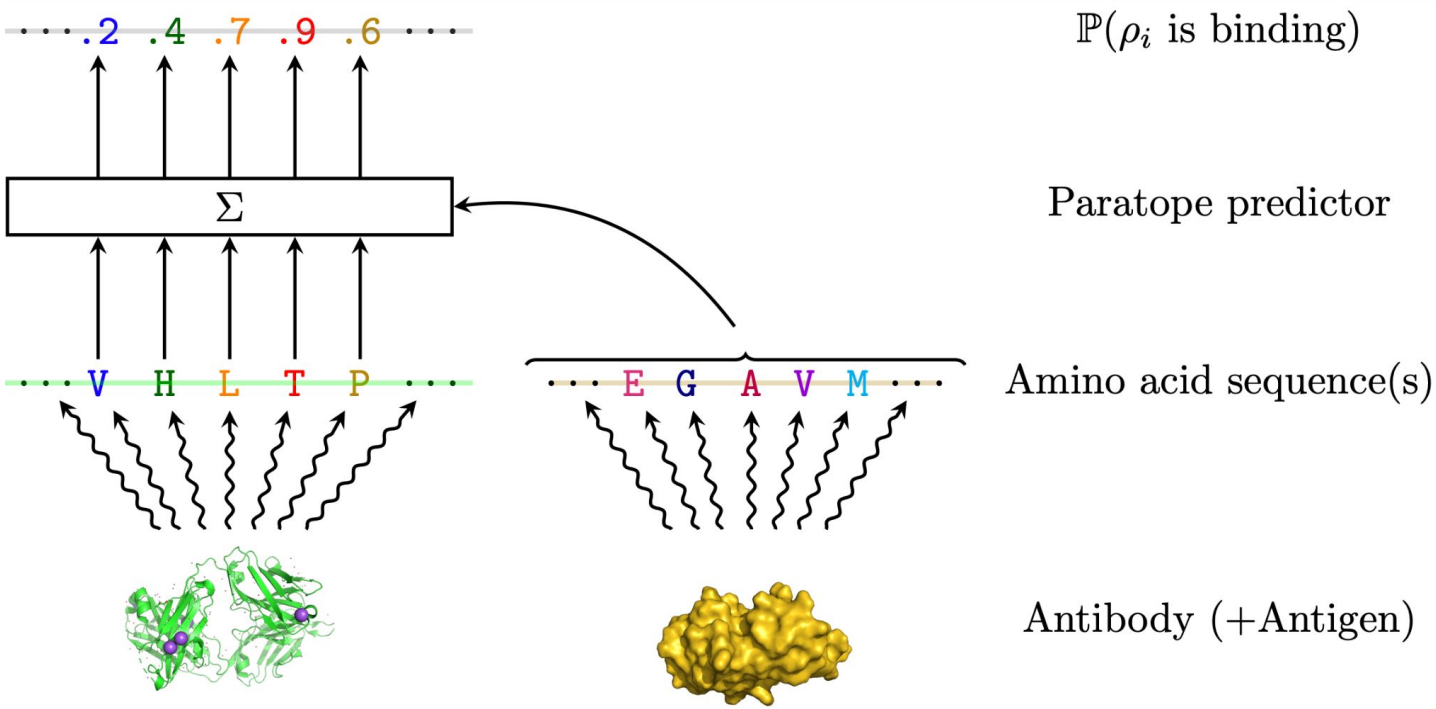
Paratope prediction



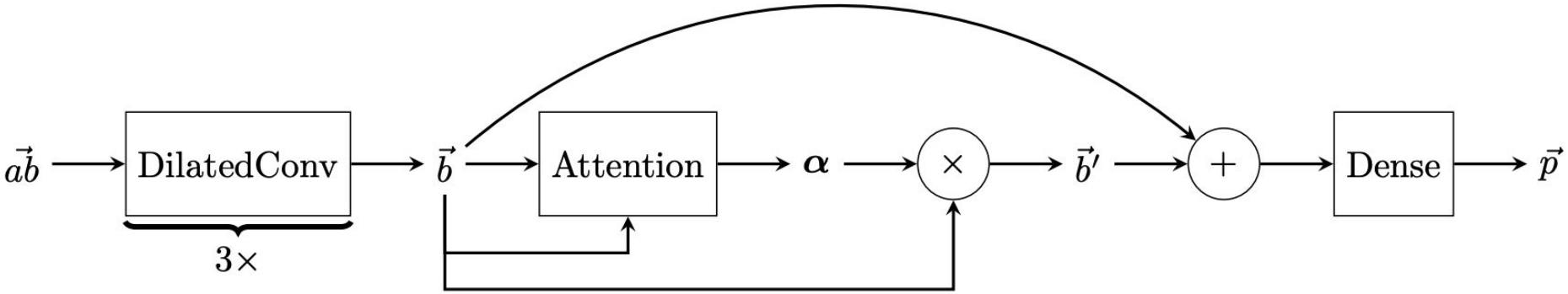
Parapred and Fast-Parapred architecture



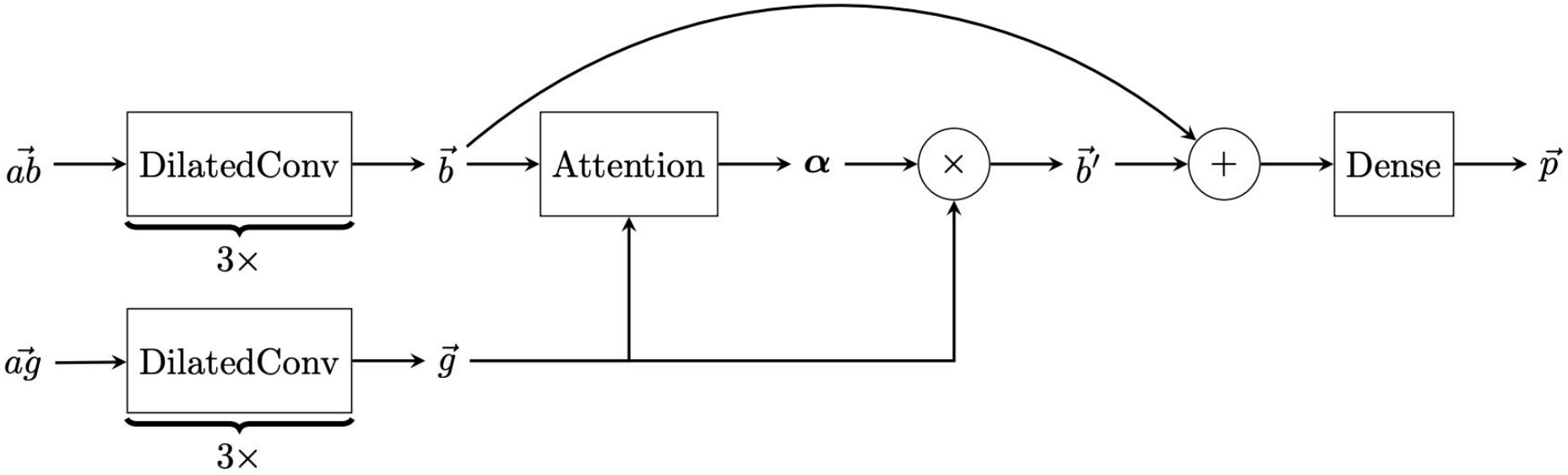
Paratope prediction (+ antigen)



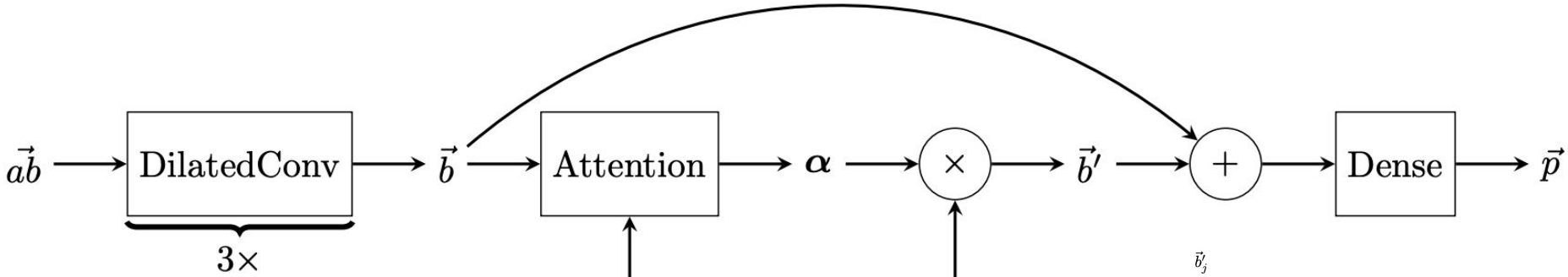
Fast-Parapred



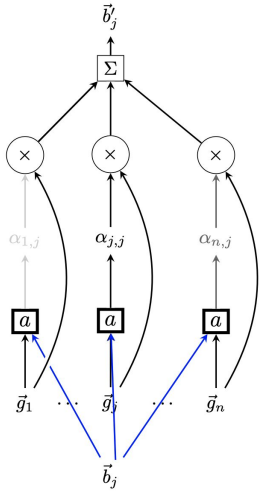
AG-Fast-Parapred



AG-Fast-Parapred

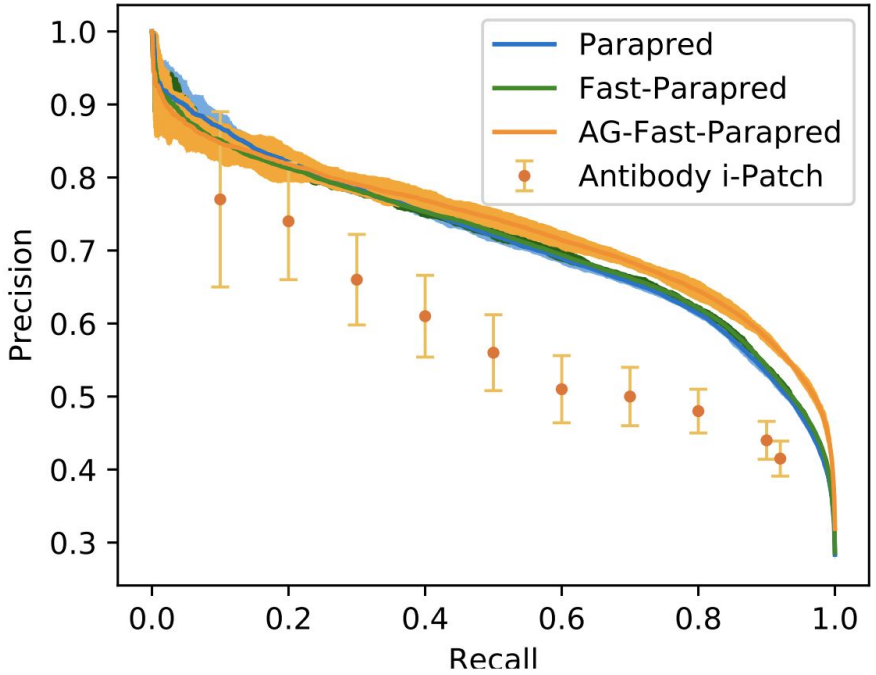


$$\vec{b'} = \sigma\left(\sum_{j \in \nu_i} a(\vec{b}_i, \vec{g}_j)\vec{g}_j\right)$$



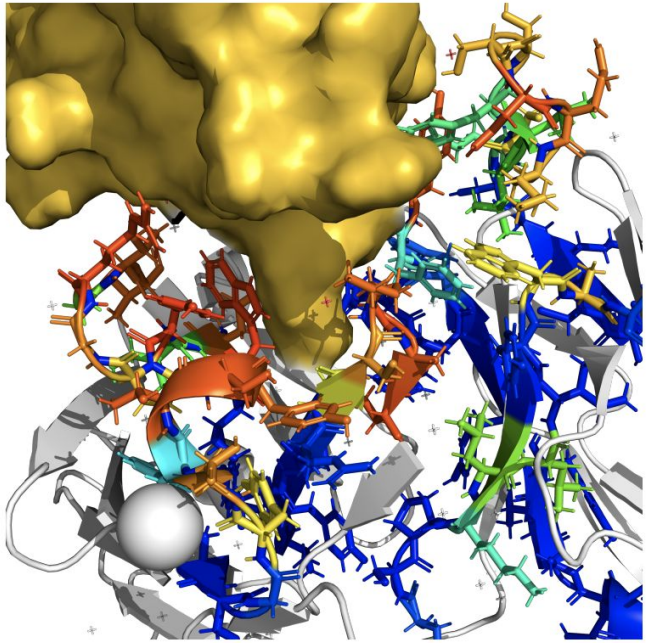
~ GAT over **fully-connected** antibody/antigen graph!

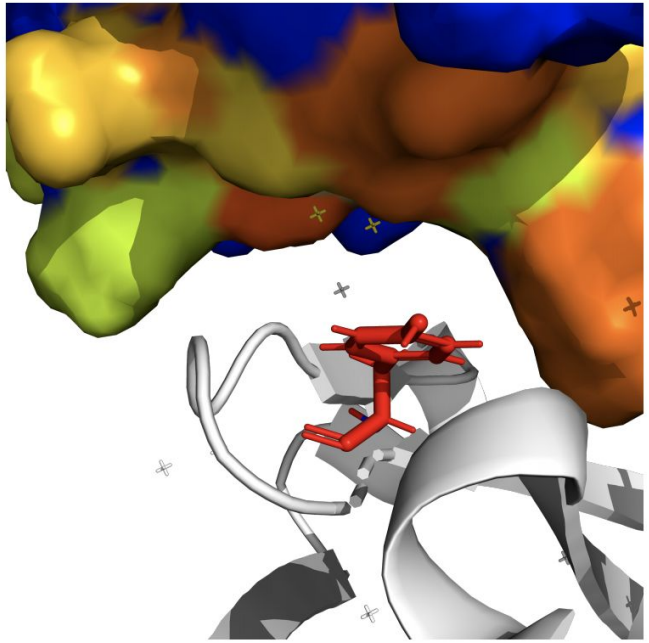
Quantitative results



	ROC AUC	MCC	Epoch time
proABC	0.851	0.522	—
Parapred	0.880 ± 0.002	0.564 ± 0.005	0.190 ± 0.019s
Fast-Parapred	0.883 ± 0.001	0.572 ± 0.004	**0.085** ± 0.015s
AG-Fast-Parapred	**0.899** ± 0.004	**0.598** ± 0.012	0.178 ± 0.020s

Qualitative results





The model learns the antibody/antigen **geometry** without being given any positional information!

...Back to the past 🦠

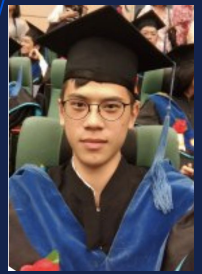
- Now it was apparent that stitching GNNs into protein-protein interaction made sense!

- Could we explore some other cases of molecular interaction?

DeepMind

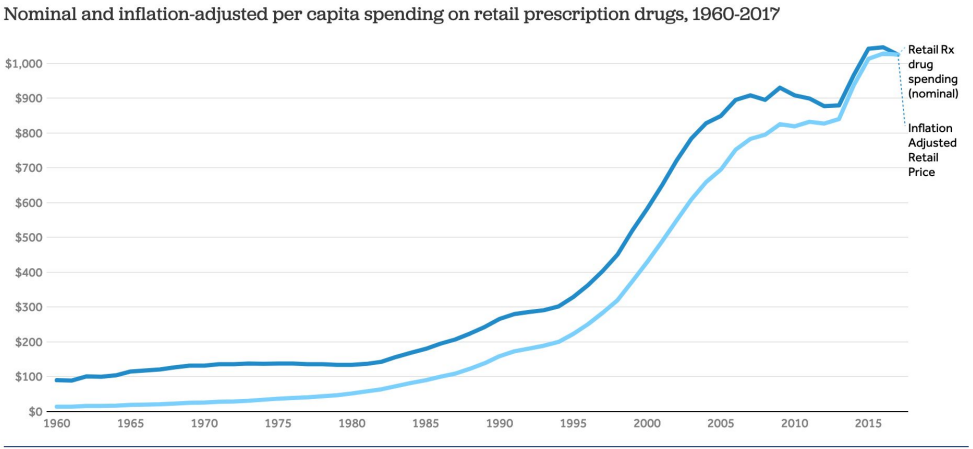
Drug-Drug Adverse Effect Prediction with Graph Co-Attention

Andreea Deac, Yu-Hsiang Huang, **Petar Veličković**, Pietro Liò and Jian Tang

Drug use is increasing

	2000	2011
Prescription Drug Use	51%	59%
>5 drugs	8.2%	15%



Polypharmacy

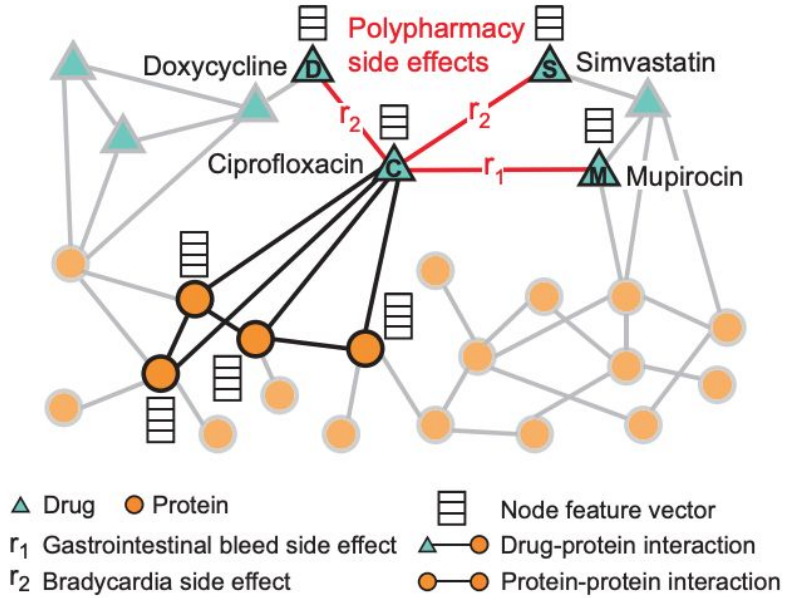
- Polypharmacy is the concurrent use of multiple medications by a patient.
- It is necessary for chronic, complex or multiple conditions and most of the increase in cost comes from treating these.
- “Hulk & Iron Man” analogy: drugs correspond to ‘heroes’, but putting them together can destroy the surrounding city!

Adverse side-effects

- Side effects affecting 15% of the population, treatment costs exceeding $177 billion/year
- Some found in Phase IV of clinical trials
- But plenty are undiscovered when the drugs are put on the market

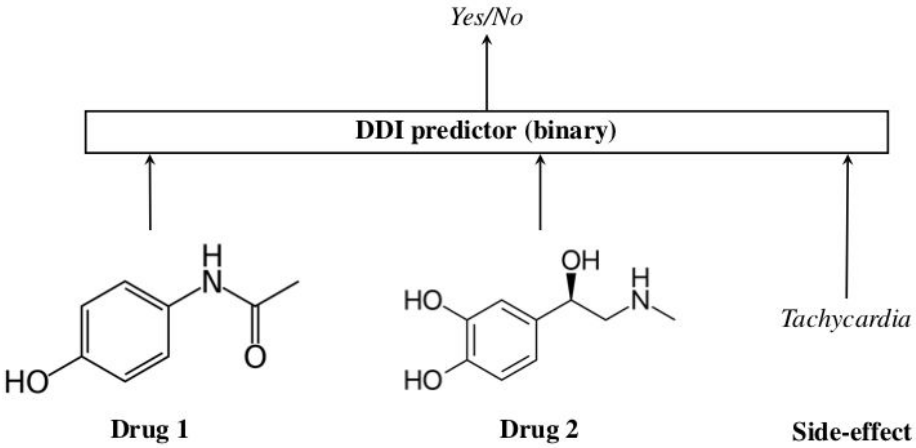
Related work

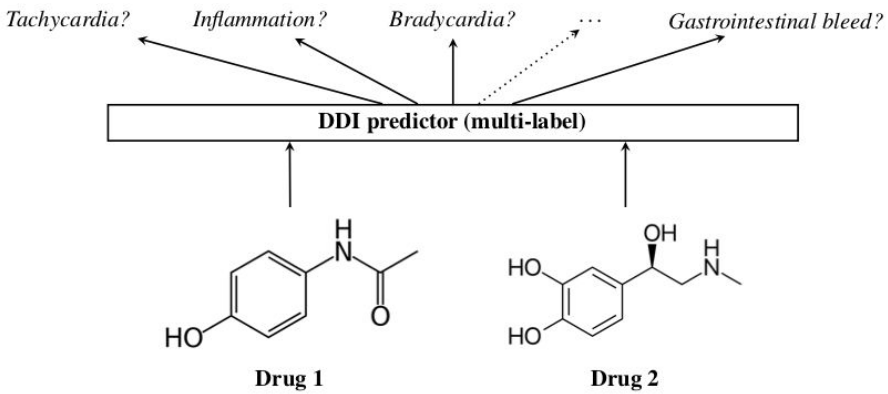
- Most models predict if a side-effect exists or not (using drug-drug similarity: chemical substructures, individual drug side effects, interaction profile fingerprints)
- Others model the interactions between pairs of drugs, pairs of proteins and drug-protein pairs to predict “missing” links between them.
- We, instead, represents **molecules as graphs**!



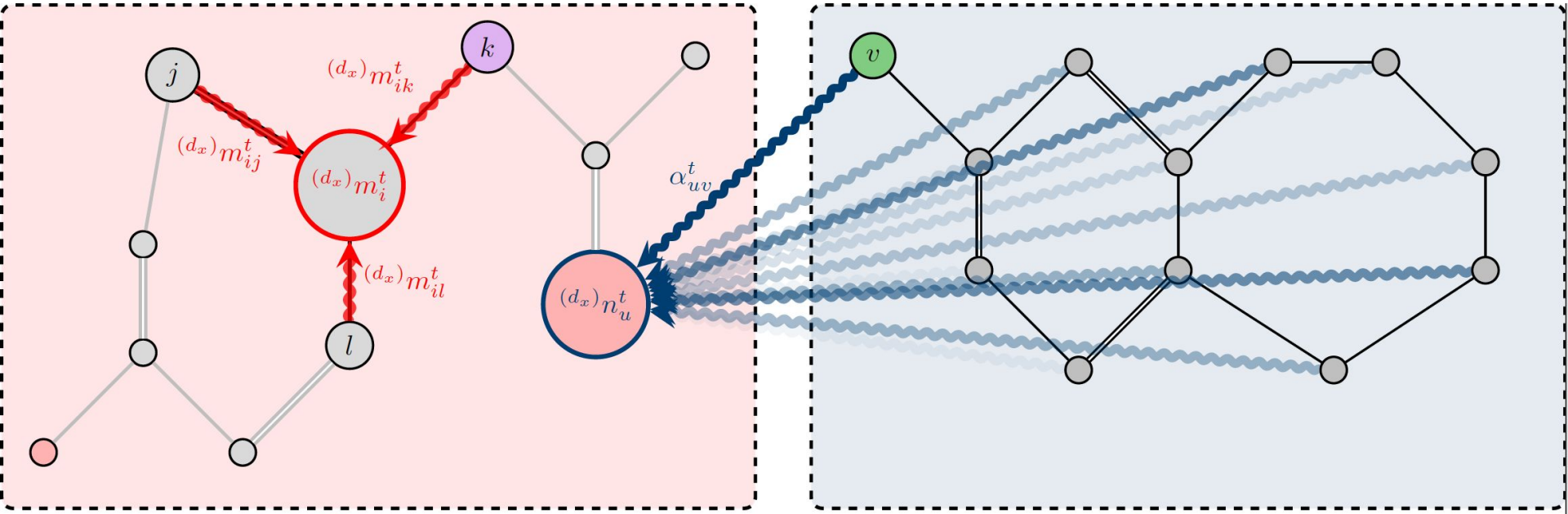
* Modeling polypharmacy side effects with graph convolutional networks, Žitnik et al, 2018

DDI - Tasks


Yes/No
DDI predictor (binary)
Tachycardia
Drug 1
Drug 2
Side-effect


Tachycardia?
Inflammation?
Bradycardia?
Gastrointestinal bleed?
DDI predictor (multi-label)
Drug 1
Drug 2

Graph co-attention



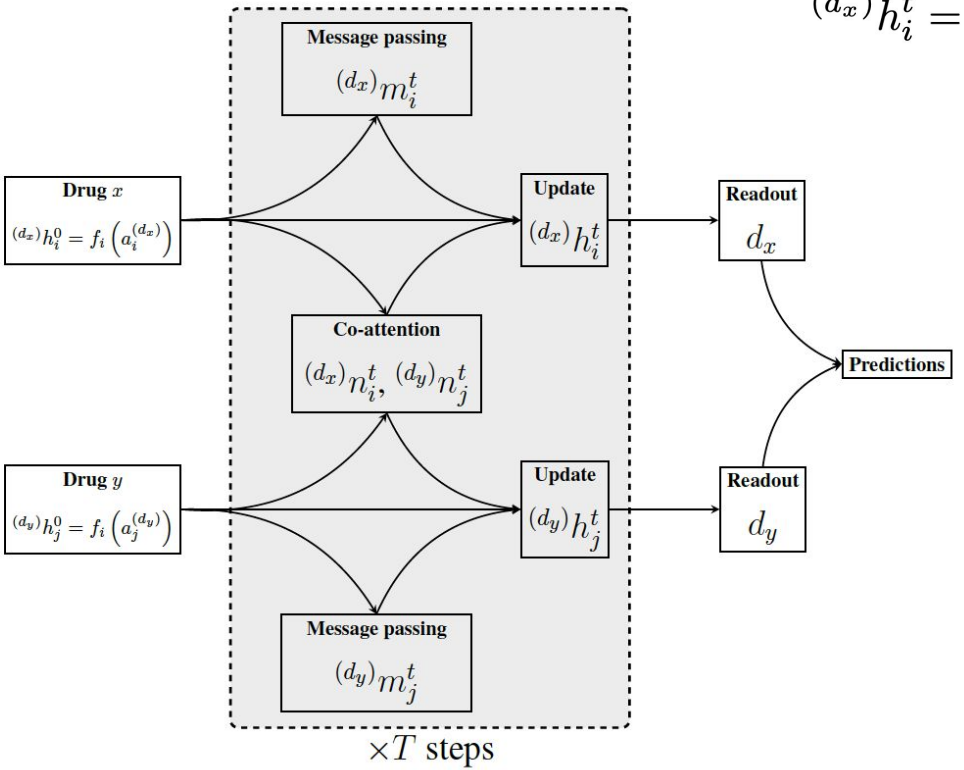
Drug x

Drug y

$$^{(d_x)}m_{ij}^t = f_e^t\left(e_{ij}^{(d_x)}\right) \odot f_v^t\left(^{(d_x)}h_j^{t-1}\right)$$

$$^{(d_x)}n_i^t = f_o^t\left(\Big\Vert_{k=1}^{K} \sum_{\forall j \in d_y} {}^{(k)}\alpha_{ij}^t \cdot {}^{(k)}\mathbf{W}_v^t \, {}^{(d_y)}h_j^{t-1}\right)$$

The (MH)CADDI Architecture



$$({}^{d_x})h_i^t = \text{LayerNorm}\left(({}^{d_x})h_i^{t-1} + ({}^{d_x})m_i^t + ({}^{d_x})n_i^t\right)$$

$$d_x = \sum_{\forall j \in d_x} f_r\left(({}^{d_x})h_j^T\right)$$

Variants considered

- *MPNN-Concat*: removing co-attention, i.e. learning drug representations independently;
- *Late-Outer*: where co-attention messages are not aggregated until the last layer;
- *CADDI*: only $K = 1$ attention head.

Quantitative results

Table 1: Comparative evaluation results after stratified 10-fold crossvalidation.

	AUROC
Drug-Fingerprints [21]	0.744
RESCAL [30]	0.693
DEDICOM [31]	0.705
DeepWalk [32]	0.761
Concatenated features [46]	0.793
Decagon [46]	0.872
MHCADDI (ours)	**0.882**
MHCADDI-ML (ours)	0.819

Table 2: Ablation study for various aspects of the MHCADDI model.

	AUROC
MPNN-Concat	0.661
Late-Outer	0.724
CADDI	0.778
MHCADDI	**0.882**

...Back to the past 💊

- It was ~at this point I graduated from my PhD, and joined DeepMind

- Gradually oriented back towards classical algorithms, and away from biology
  - Luckily, biology is ***packed*** with interesting classical algorithms :)

- The following three works (time permitting) represent a medley of biological approaches I was involved in during this time.
  - Two of these opportunities came *not far from home* :)
  - The third one was **years** in the making!

DeepMind

Hierarchical Protein Function Prediction with Tail-GNNs

Stefan Spalević, **Petar Veličković**, Jovana Kovačević and Mladen Nikolić

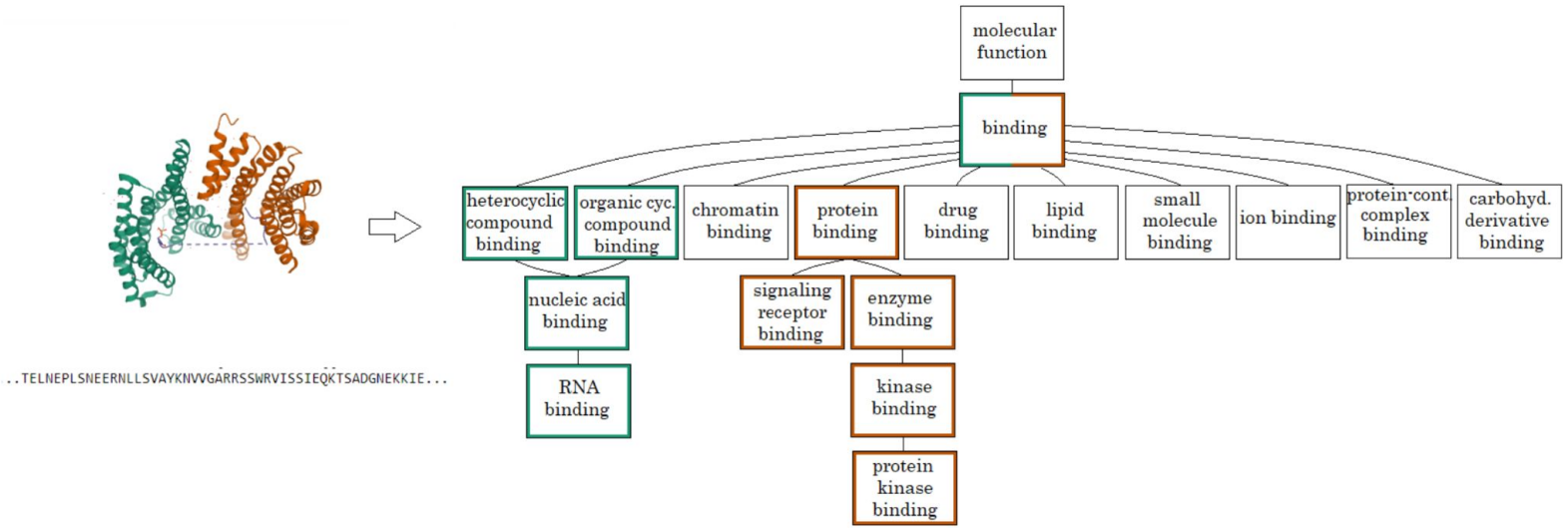
Protein function prediction

- Detecting mechanisms of action for proteins is a highly relevant task!

- It is also an area ripe with graphs!
  - Protein itself can be represented as a graph (if known structure; Gligorijević et al.)
  - Protein-protein interaction networks are graphs (standard **PPI** benchmark for GNNs)
  - In this particular domain, a graph comes up in one more place…

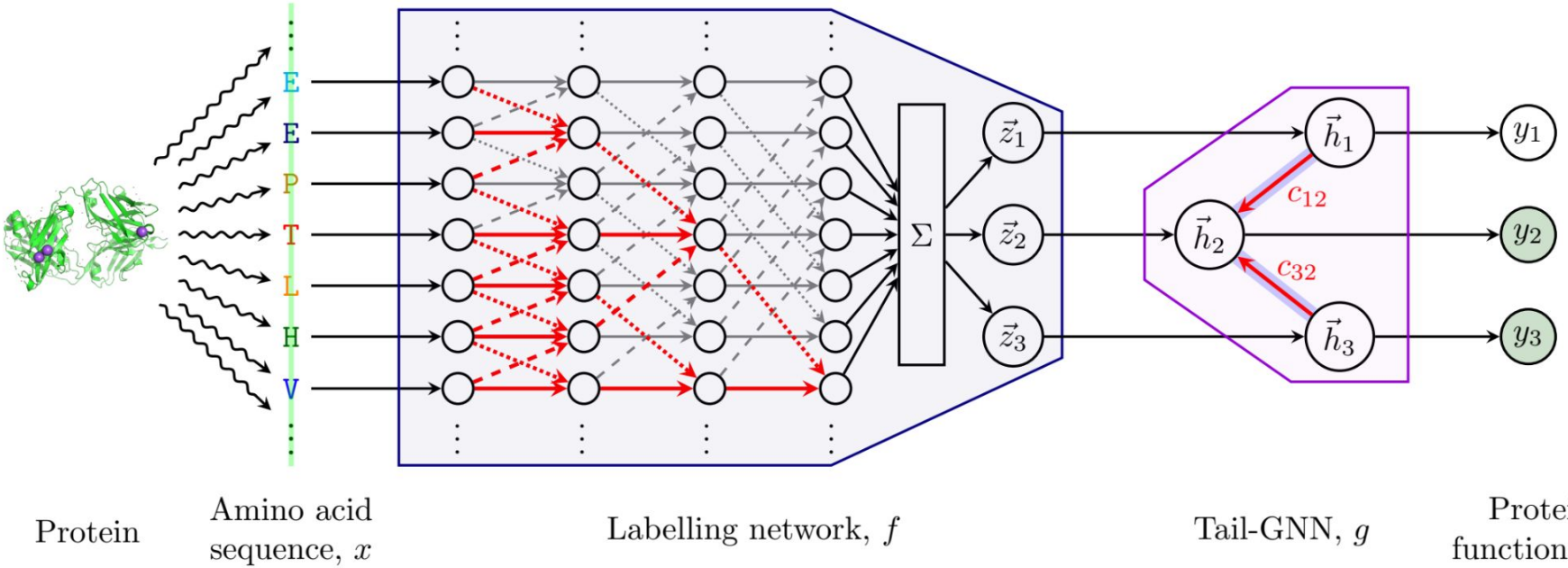
Protein function prediction

- The label space of functions is itself a graph! (**gene ontology**)



- Requires a GNN in the **label space**
  - Our literature survey suggested no proposals like this!
  - Once again, a biological problem motivates a core architecture

Tail-GNN


E
E
P
T
L
H
V
Σ
$\vec{z}_1$
$\vec{z}_2$
$\vec{z}_3$
$\vec{h}_1$
$\vec{h}_2$
$\vec{h}_3$
c_{12}
c_{32}
y_1
y_2
y_3
Protein
Amino acid sequence, x
Labelling network, f
Tail-GNN, g
Protein functions, $\mathbf{Y}$

Quantitative results

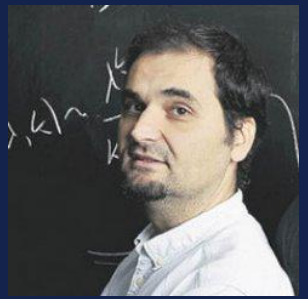
- With the right aggregator choice + spectral features, yields significant benefits!

Model	**Validation** F_1	**Test** F_1
Labelling network	0.582 ± 0.003	0.584 ± 0.003
Tail-GNN-mean	0.583 ± 0.006	0.586 ± 0.004
Tail-GNN-GAT	0.582 ± 0.004	0.587 ± 0.005
Tail-GNN-max	0.581 ± 0.002	0.585 ± 0.004
Tail-GNN-sum	$\mathbf{0.596} \pm 0.003$	$\mathbf{0.600} \pm 0.003$
Tail-GNN-sum (no spectral fts.)	0.587 ± 0.007	0.590 ± 0.008

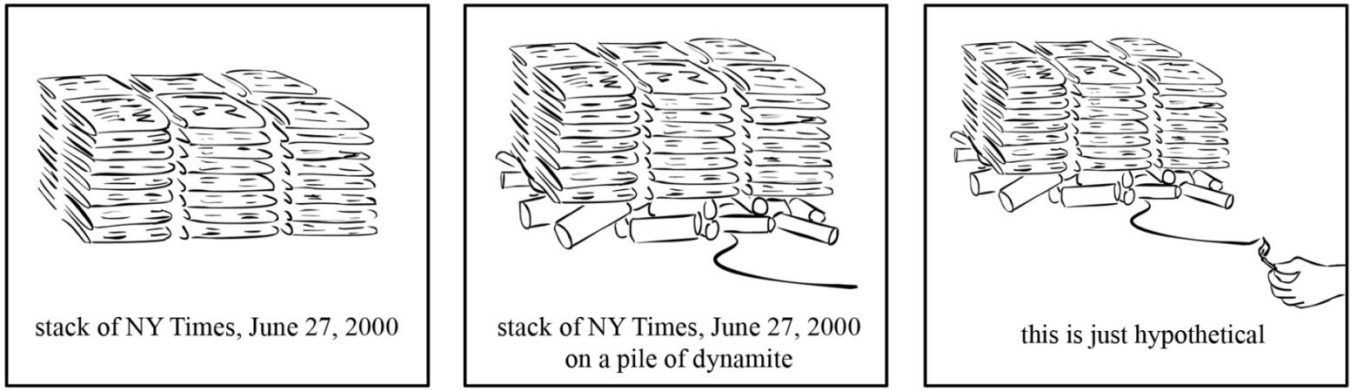
DeepMind

A Step Towards Neural Genome Assembly

Lovro Vrček, **Petar Veličković** and Mile Šikić



Genome assembly



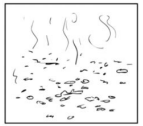
Genome assembly

Multiple Copies of a Genome (Millions of them)

CTGATGATGGACTACGCTACTACTGCTAGCTGTATTACGATCAGCTACCACATCGTAGCTACGATGCATTAGCAAGCTATCGGATCAGCTACCACATCGTAGC
CTGATGATGGACTACGCTACTACTGCTAGCTGTATTACGATCAGCTACCACATCGTAGCTACGATGCATTAGCAAGCTATCGGATCAGCTACCACATCGTAGC
CTGATGATGGACTACGCTACTACTGCTAGCTGTATTACGATCAGCTACCACATCGTAGCTACGATGCATTAGCAAGCTATCGGATCAGCTACCACATCGTAGC
CTGATGATGGACTACGCTACTACTGCTAGCTGTATTACGATCAGCTACCACATCGTAGCTACGATGCATTAGCAAGCTATCGGATCAGCTACCACATCGTAGC

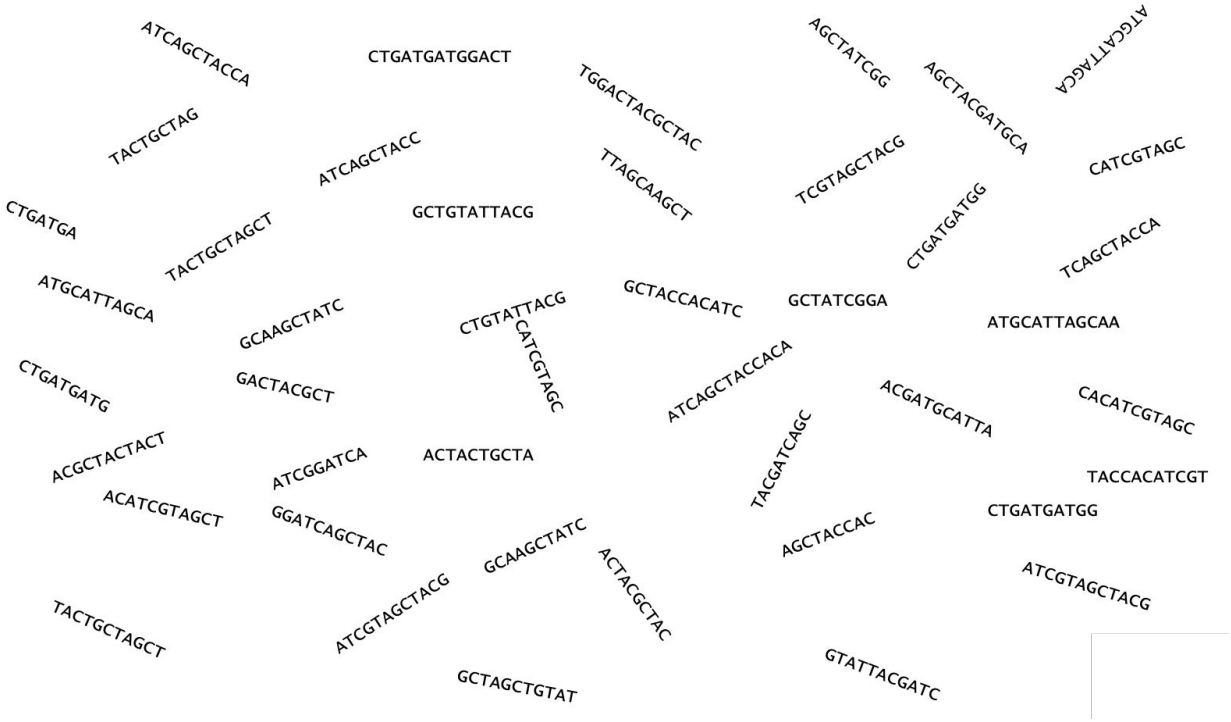
Breaking the Genomes at Random Positions

“Burning” Some Reads

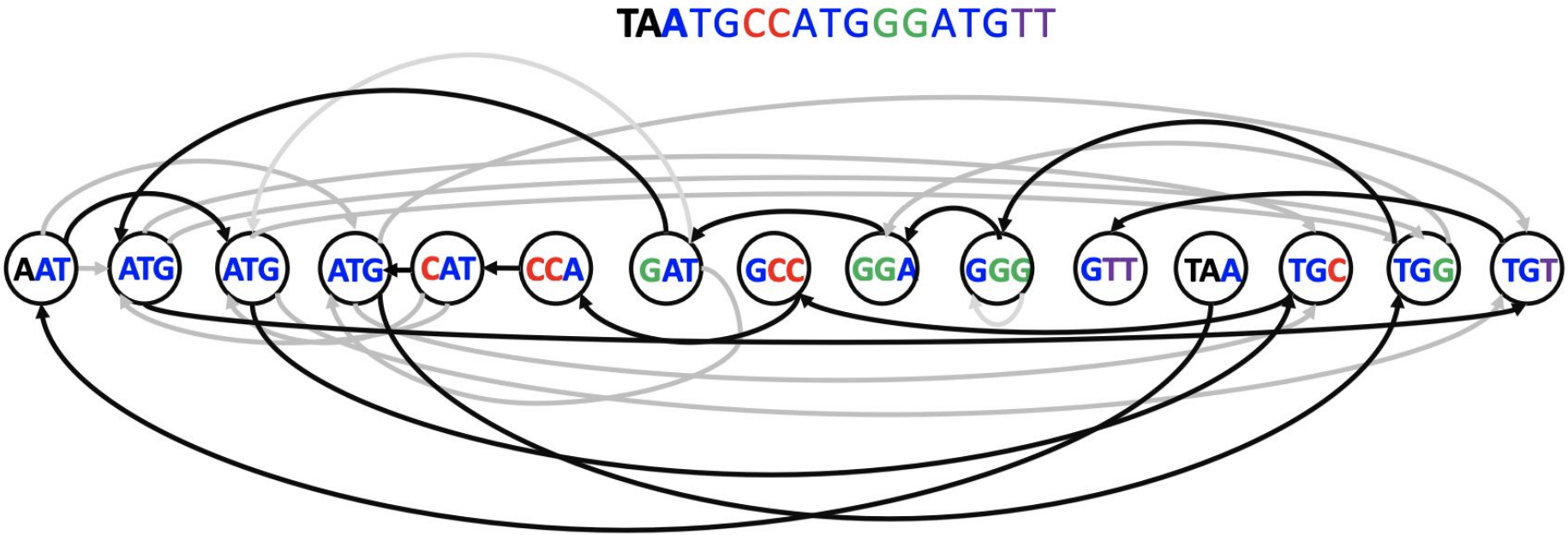


CTGATGA TGGACTACGCTAC TACTGCTAG CTGTATTACG ATCAGCTACCACA TCGTAGCTACG ATGCATTAGCAA GCTATCGGA TCAGCTACCA CATCGTAGC
CTGATGATG GACTACGCT ACTACTGCTA GCTGTATTACG ATCAGCTACC ACATCGTAGCT ACGATGCATTA GCAAGCTATC GGATCAGCTAC CACATCGTAGC
CTGATGATGG ACTACGCTAC TACTGCTAGCT GTATTACGATC AGCTACCAC ATCGTAGCTACG ATGCATTAGCA AGCTATCGG A TCAGCTACCA CATCGTAGC
CTGATGATGGACT ACGCTACTACT GCTAGCTGTAT TACGATCAGC TACCACATCGT AGCTACGATGCA TTAGCAAGCT ATCGGATCA GCTACCACATC GTAGC

Genome assembly



Genome assembly using **Hamiltonian paths**



But… the reads are **faulty!**

- Learn **algorithms** to prune errors

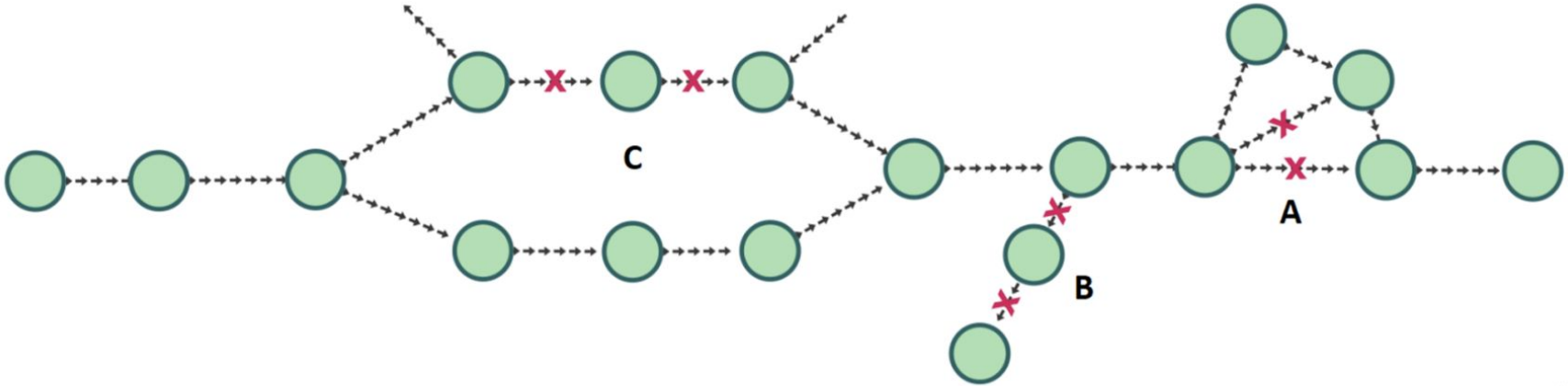


Figure 1: Example of structures in the assembly graph, before all the simplification steps. Letter **A** marks transitive edges, a short tip is marked with **B**, and a bubble which cannot be fully resolved is marked with **C**. Red crosses show which edges can be removed from the assembly graph.

Towards **neural** genome assembly

Pre-train on **synthetic** graphs...

...generalises to **real** organisms!

(Still preliminary, but **encouraging**!)

Table 1: Scaling of algorithm execution for isolated learning of algorithms.

	Scaling				
Algorithm	1x	2x	4x	8x	20x
Transitive removal	98.10%	99.00%	99.52%	99.76%	99.91%
Tips trimming	98.05%	98.96%	99.49%	99.70%	99.87%
Bubble popping	98.16%	99.03%	99.53%	99.77%	99.90%

Table 2: Scaling of algorithm execution for parallel learning of algorithms.

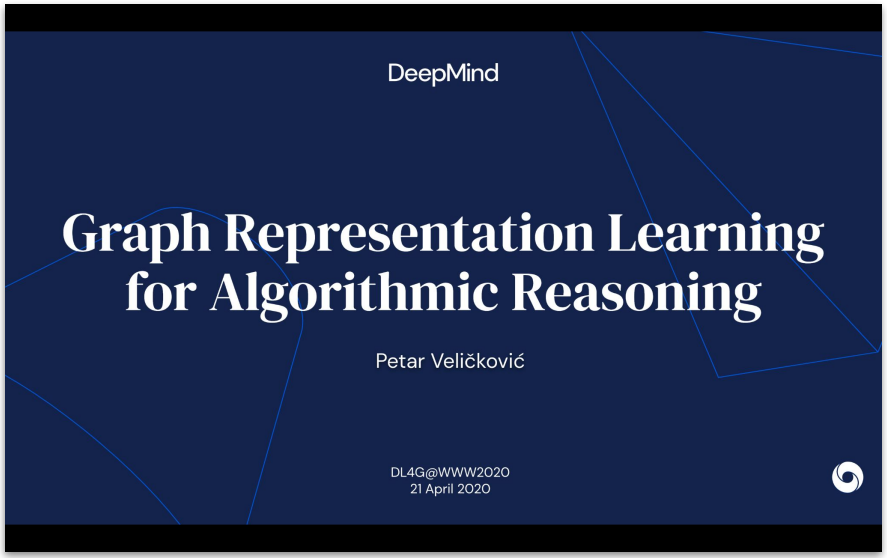
	Scaling				
Algorithm	1x	2x	4x	8x	20x
Transitive removal	98.21%	99.07%	99.50%	99.89%	99.92%
Tips trimming	98.45%	99.11%	99.46%	99.76%	99.89%
Bubble popping	98.17%	99.02%	99.51%	99.78%	99.90%

Table 3: Parallel algorithm execution on the assembly graph of lambda phage.

	Transitive removal	Tips trimming	Bubble popping
Lambda phage	98.04%	93.33%	97.47%
E. coli	99.67%	98.84%	99.26%

Further insight: Algorithmic reasoning

If you would like to know more details about teaching GNNs to be more “algorithmic”:



https://www.youtube.com/watch?v=IPQ6CPoluok

https://drive.google.com/file/d/1_EQ9Yu7VEkvrHaVHl_WbT5ABvxrSNY-s/view?usp=sharing

Broader context: Combinatorial Optimisation

Combinatorial optimization and reasoning with graph neural networks

Quentin Cappart[1], Didier Chételat[2], Elias Khalil[3], Andrea Lodi[2], Christopher Morris[2], and Petar Veličković[*4]

[1]Department of Computer Engineering and Software Engineering, Polytechnique Montréal
[2]CERC in Data Science for Real-Time Decision-Making, Polytechnique Montréal
[3]Department of Mechanical & Industrial Engineering, University of Toronto
[4]DeepMind

Combinatorial optimization is a well-established area in operations research and computer science. Until recently, its methods have focused on solving problem instances in isolation, ignoring the fact that they often stem from related data distributions in practice. However, recent years have seen a surge of interest in using machine learning, especially graph neural networks (GNNs), as a key building block for combinatorial tasks, either as solvers or as helper functions. GNNs are an inductive bias that effectively encodes combinatorial and relational input due to their permutation-invariance and sparsity awareness. This paper presents a conceptual review of recent key advancements in this emerging field, aiming at both the optimization and machine learning researcher.

Our *43-page* survey on GNNs for CO!

https://arxiv.org/abs/2102.09544

Section 3.3. details algorithmic reasoning, with comprehensive references.

DeepMind

Predicting Patient Outcomes with Graph Representation Learning

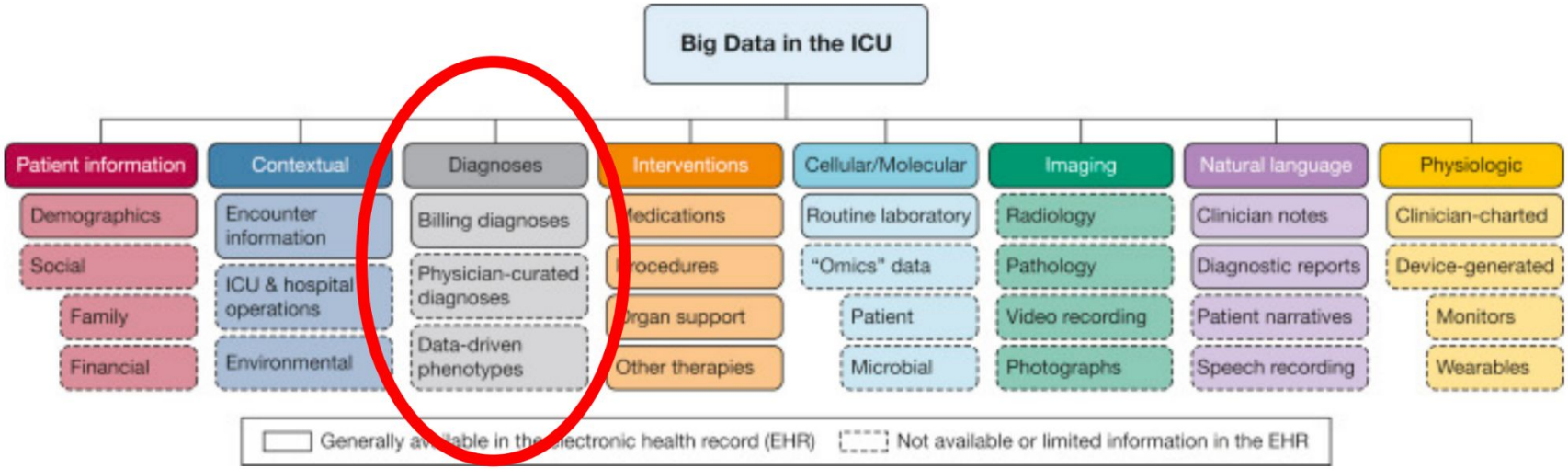
Emma Rocheteau*, Catherine Tong*, **Petar Veličković**, Nicholas Lane and Pietro Liò




Electronic Health Records (EHRs) in the ICU

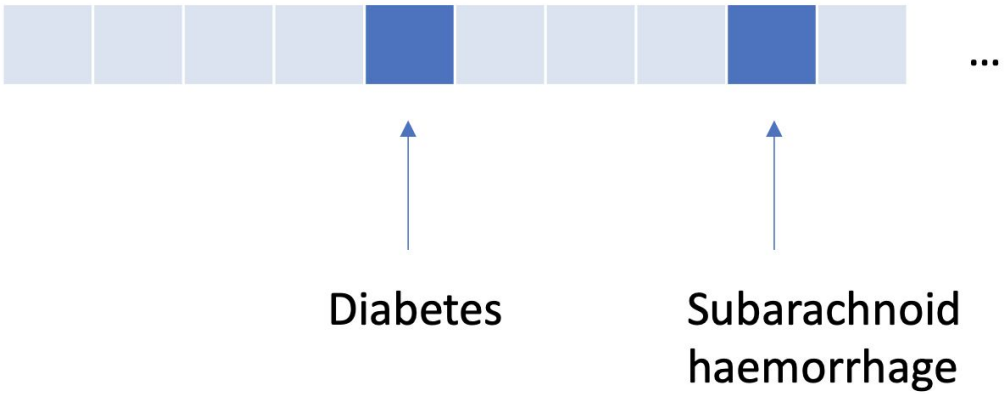
- EHRs can provide plentiful information about a patient's progression
  - But not all data contained in there are easy to leverage by deep learning systems!

- Today, we focus on **diagnoses**.



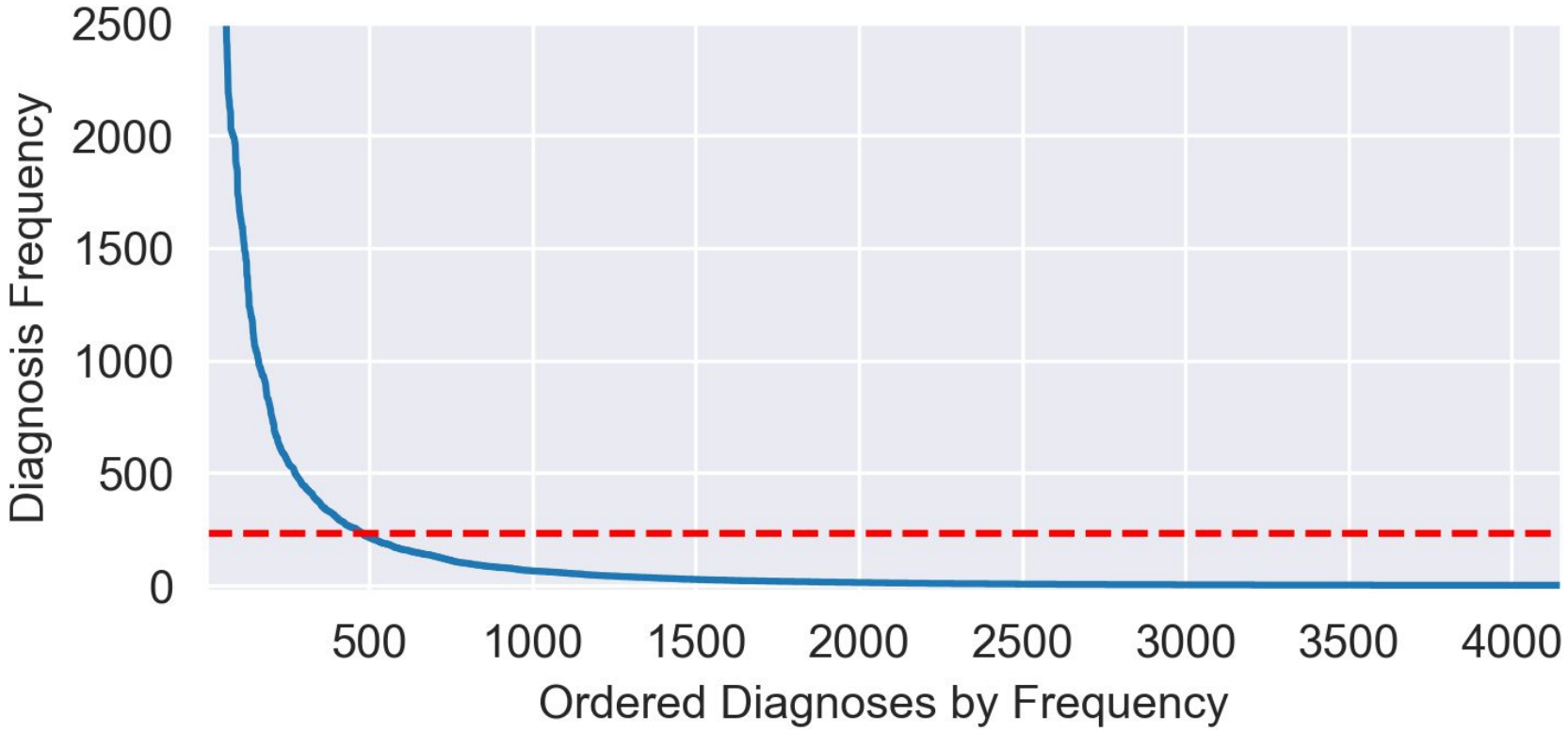
Diagnosis information is hard to use

- Large number of possibilities makes distinguishing *patterns* of comorbidity **difficult**.



- There is a lack of data for rarer combinations.
  - A long tail of ***rare*** diagnoses, difficult for deep learning models to leverage!

Distribution of diagnoses in eICU



The “pattern recognition” method

- Commonly, the “long tail” of diagnoses is *discarded* and the rest embedded.
  - But this long tail often holds the most **useful** cues, which diagnosticians regularly use!

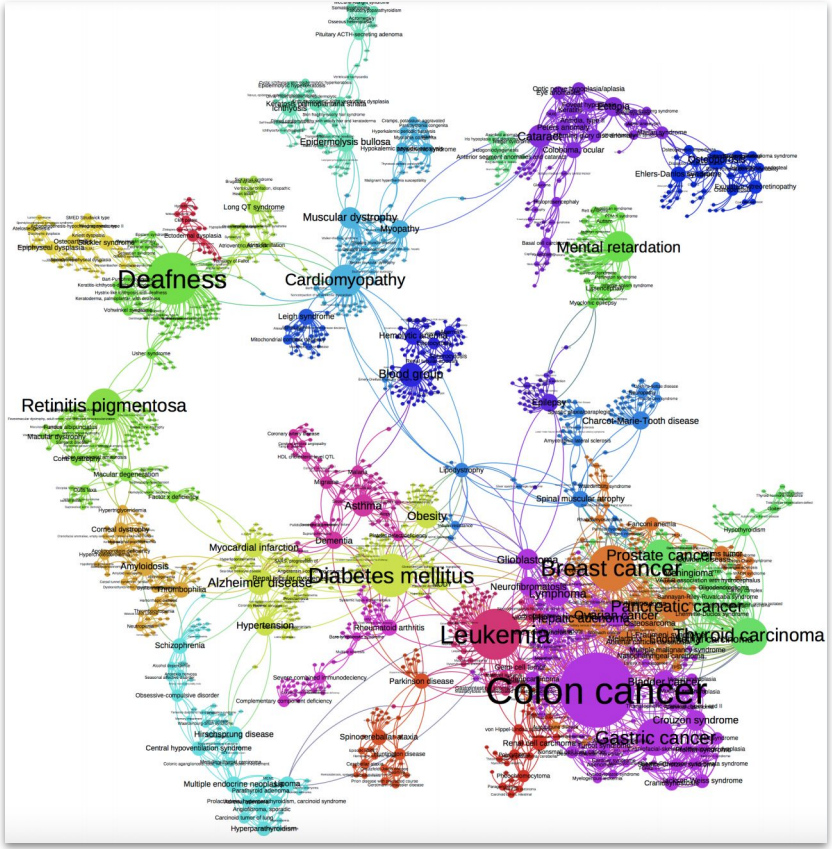
- How do **clinicians** often make decisions about diagnoses or prognoses?

- The *pattern recognition* diagnostic method, as described by Wikipedia:

*“In a pattern recognition method the provider uses **experience** to recognize a pattern of clinical characteristics… This may be the primary method used in cases where diseases are "obvious", or the provider's **experience** may enable him or her to **recognize** the condition **quickly**.”*

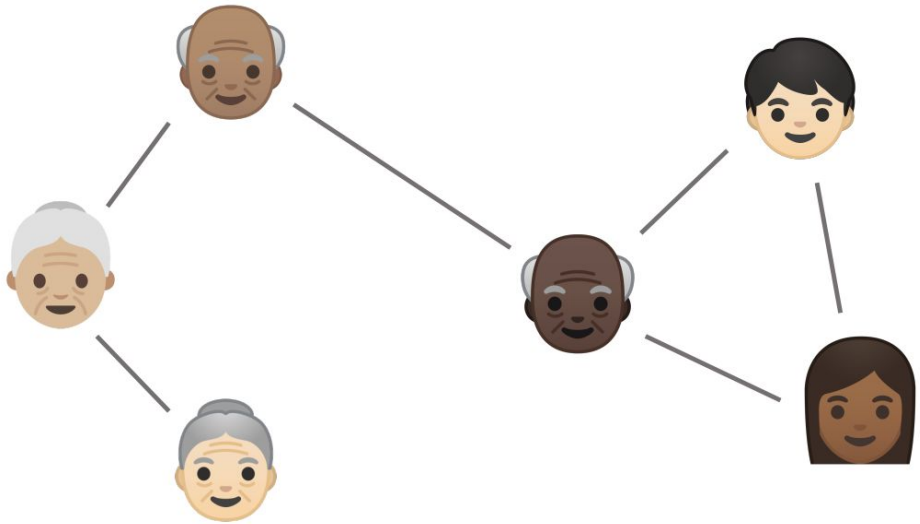
- We interpret experience as exploitation of ***related*** cases the clinician treated in the past.
  - Hence, the cases form a **graph**!

These links definitely exist :)



The graph of patients

- Key assumption: patients with **related diagnoses** will likely have **related prognoses**!



- If we use this signal wisely, it can be a great way to regularise our model **and** make advantage of sparse diagnosis data.

How to **build** the graph?

- The “relatedness” score between two patients *i* and *j* is given by:

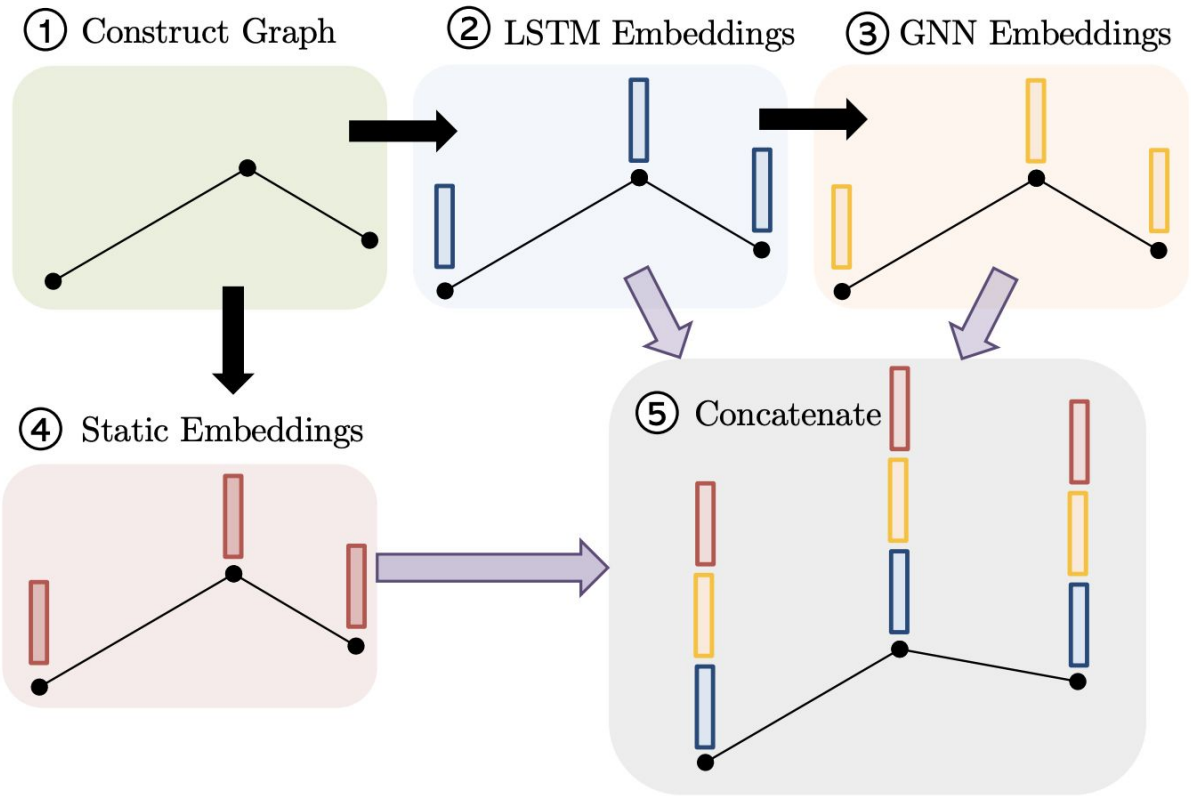
$$\mathcal{M}_{ij} = \alpha \sum_{\mu=1}^{m} \mathcal{D}_{i\mu}\mathcal{D}_{j\mu}\left(d_{\mu}^{-1} + \gamma\right) - \sum_{\mu=1}^{m} \mathcal{D}_{i\mu} + \mathcal{D}_{j\mu}$$

  where:

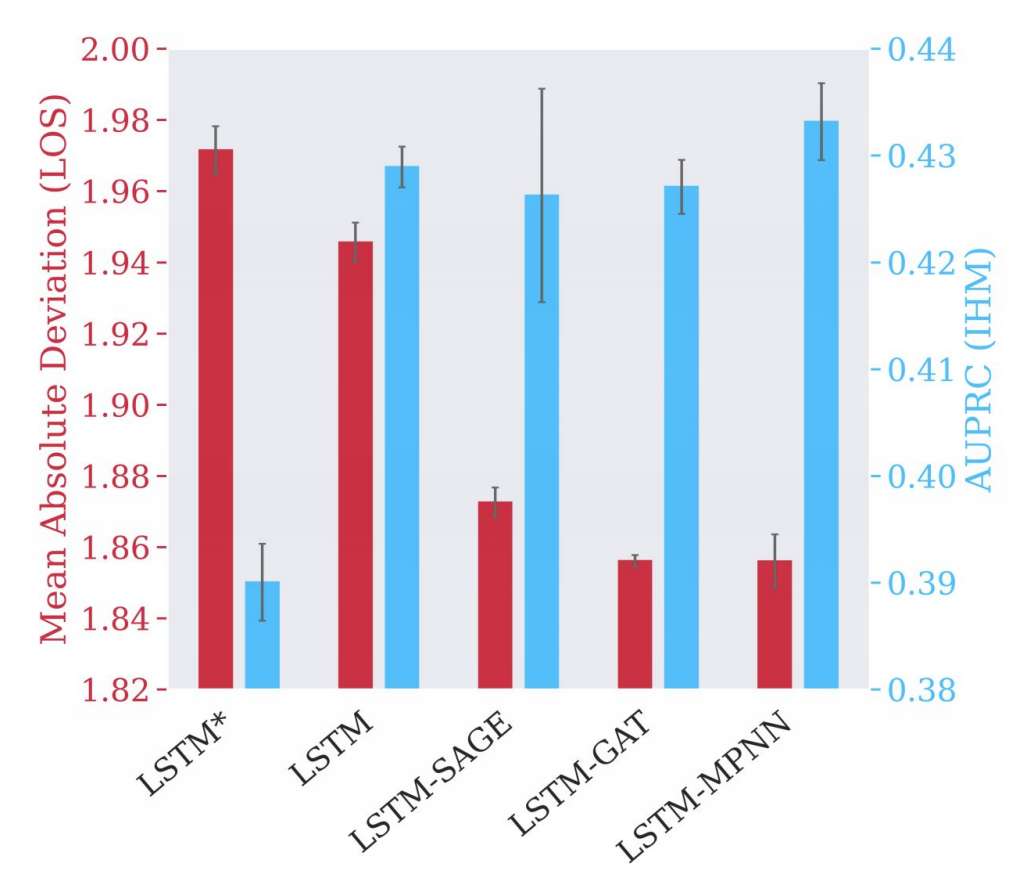
  - **D** is a *diagnosis matrix* (s.t. $\mathbf{D}_{i\mu}$ means “does patient *i* have diagnosis μ”?)
  - d_{μ} is the *frequency* of diagnosis μ
  - m is the *number* of diagnoses
  - α and γ are *hyperparameters*

- Can **threshold** based on the relatedness scores

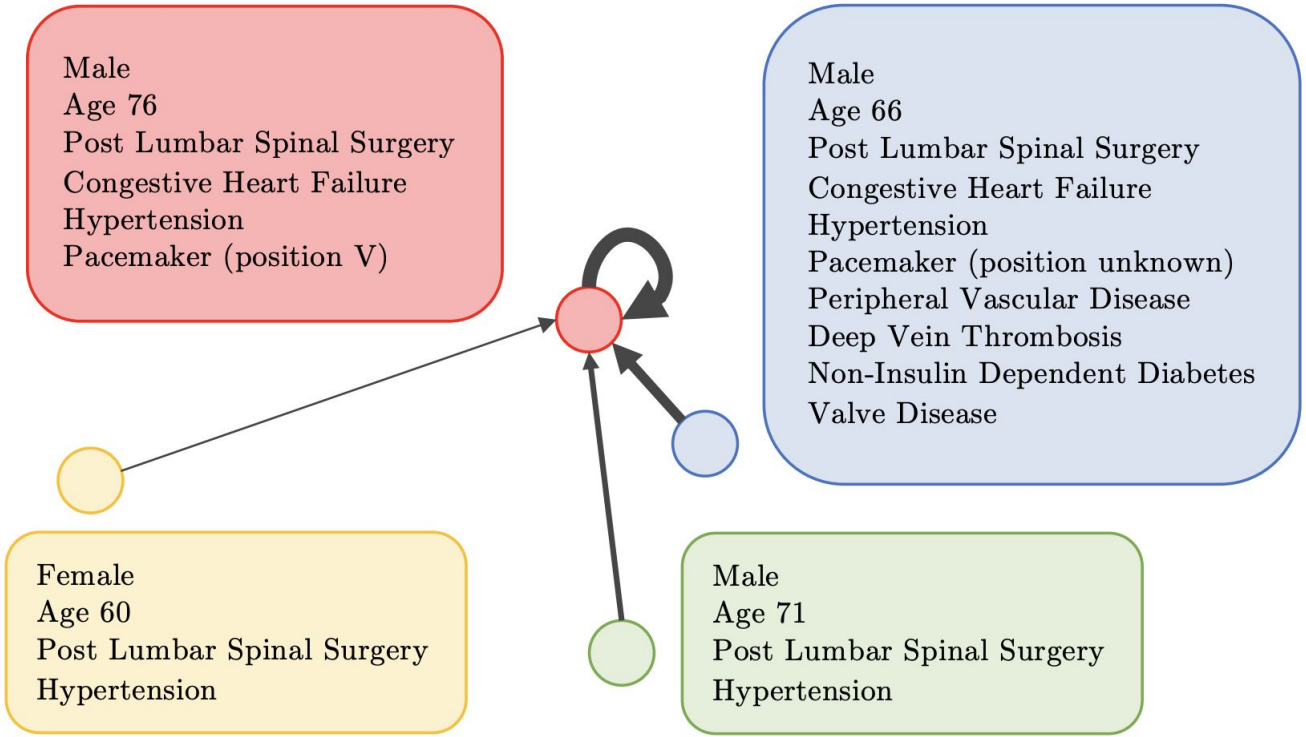
Hybrid LSTM-GNN model



Our results


Mean Absolute Deviation (LOS)
2.00
1.98
1.96
1.94
1.92
1.90
1.88
1.86
1.84
1.82
AUPRC (IHM)
0.44
0.43
0.42
0.41
0.40
0.39
0.38
LSTM*
LSTM
LSTM-SAGE
LSTM-GAT
LSTM-MPNN

Qualitative: LSTM-GAT Attention weights



AAAI'21 Workshop Recognition

In conclusion…

- Studying biological problems with graph representation learning is likely here to stay
  - Abundance of data “sitting and waiting to be processed”
  - In many problems of interest, state-of-the-art is still a **shallow** method
  - Often, biological problems can give rise to **core** methodological progress.

- With the right mindset, no proper biological training is needed!
  - Just the ability to carefully listen, and work **together** with biologists.

- For biologists: I hope I’ve convinced you that GNNs could be a useful tool!

- But ultimately, I would love to stimulate, and see even more of, **interdisciplinary** research.

DeepMind

Thank you!

Questions?

petarv@google.com | https://petar-v.com

With many thanks to:
Lead authors: Guillem Cucurull, Andreea Deac, Stefan Spalević, Lovro Vrček, Emma Rocheteau, Catherine Tong
Bio/chem: Konrad Wagstyl, Estrid Jakobsen, Alan Evans, Pietro Sormanni, Pietro Lio, Mile Šikić, Jovana Kovačević
ML collaborators: Arantxa Casanova, Yu-Hsiang Huang
ML advisors: Adriana Romero, Michal Drozdzal, Yoshua Bengio, Jian Tang,, Mladen Nikolić, Nic Lane