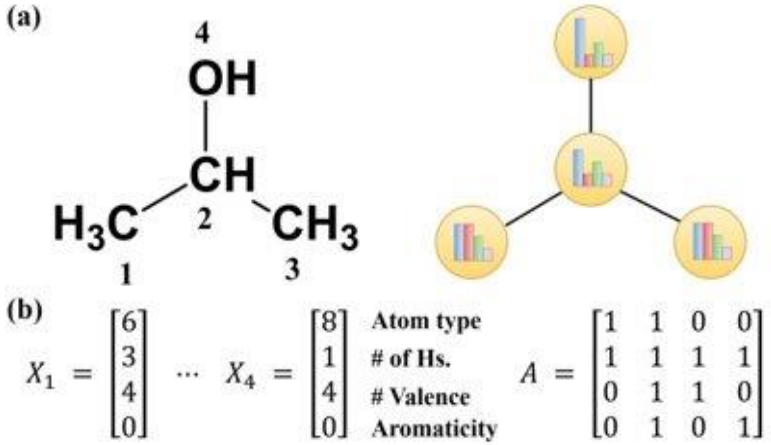
Current Topic of
Graph Neural Network
for cheminformatic and bioinformatics



Jihun Jeung

jihun@gm.gist.ac.kr

Life Data Mining Lab, School of Life science, GIST

Last Update 2021.12.15

Article

Highly accurate protein structure prediction with AlphaFold

https://doi.org/10.1038/s41586-021-03819-2

Received: 11 May 2021

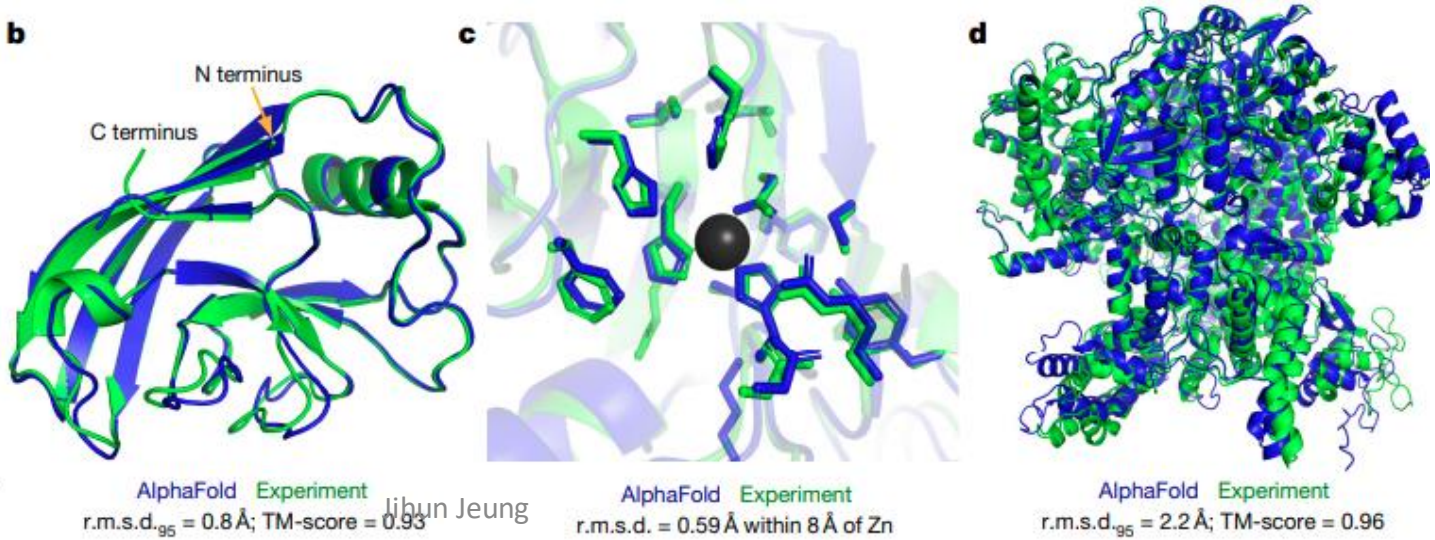
Accepted: 12 July 2021

Published online: 15 July 2021

Open access

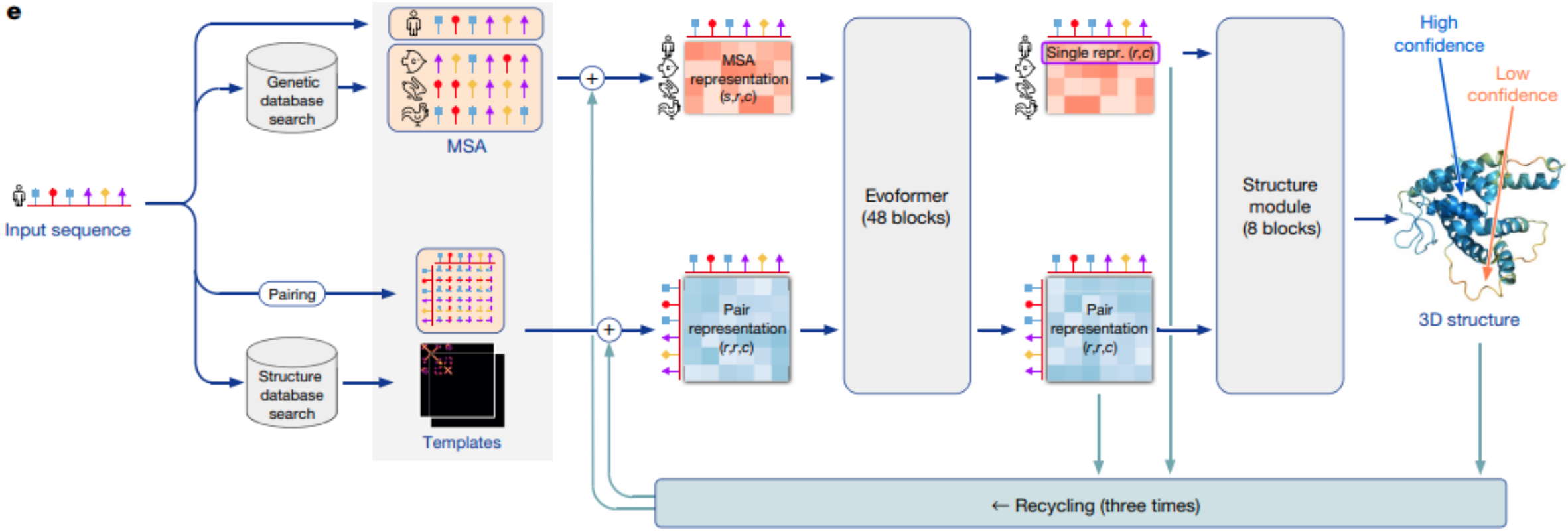
Check for updates

John Jumper[1,4]✉, Richard Evans[1,4], Alexander Pritzel[1,4], Tim Green[1,4], Michael Figurnov[1,4], Olaf Ronneberger[1,4], Kathryn Tunyasuvunakool[1,4], Russ Bates[1,4], Augustin Žídek[1,4], Anna Potapenko[1,4], Alex Bridgland[1,4], Clemens Meyer[1,4], Simon A. A. Kohl[1,4], Andrew J. Ballard[1,4], Andrew Cowie[1,4], Bernardino Romera-Paredes[1,4], Stanislav Nikolov[1,4], Rishub Jain[1,4], Jonas Adler[1], Trevor Back[1], Stig Petersen[1], David Reiman[1], Ellen Clancy[1], Michal Zielinski[1], Martin Steinegger[2,3], Michalina Pacholska[1], Tamas Berghammer[1], Sebastian Bodenstein[1], David Silver[1], Oriol Vinyals[1], Andrew W. Senior[1], Koray Kavukcuoglu[1], Pushmeet Kohli[1] & Demis Hassabis[1,4]✉



https://www.nature.com/articles/s41586-021-03819-2

GNN increase the predicton accuracy in AlphaFold

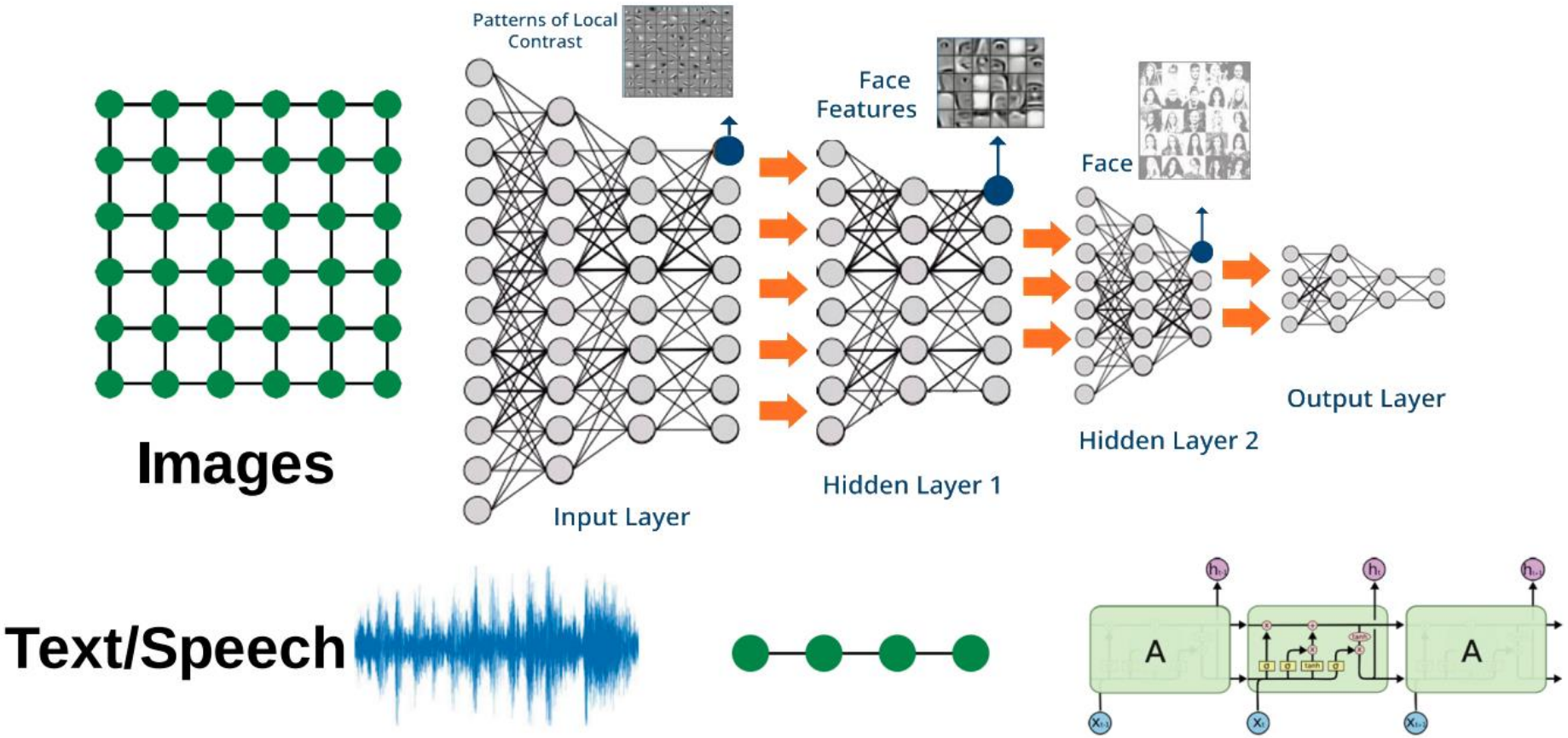


https://www.nature.com/articles/s41586-021-03819-2

Outline

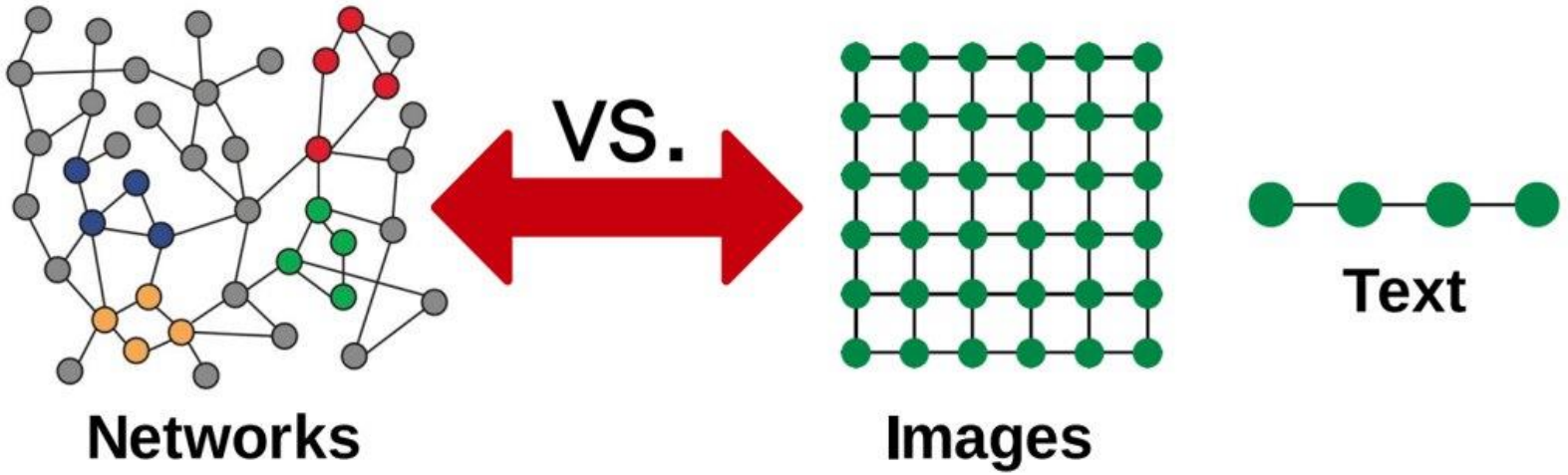
- Deep learning
- Graph Neural Network
  - GraphSAGE
  - GG-NN
  - GAT
- Application of GNN
  - Cheminformatics
  - bioinformatics

Deep Learning



Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

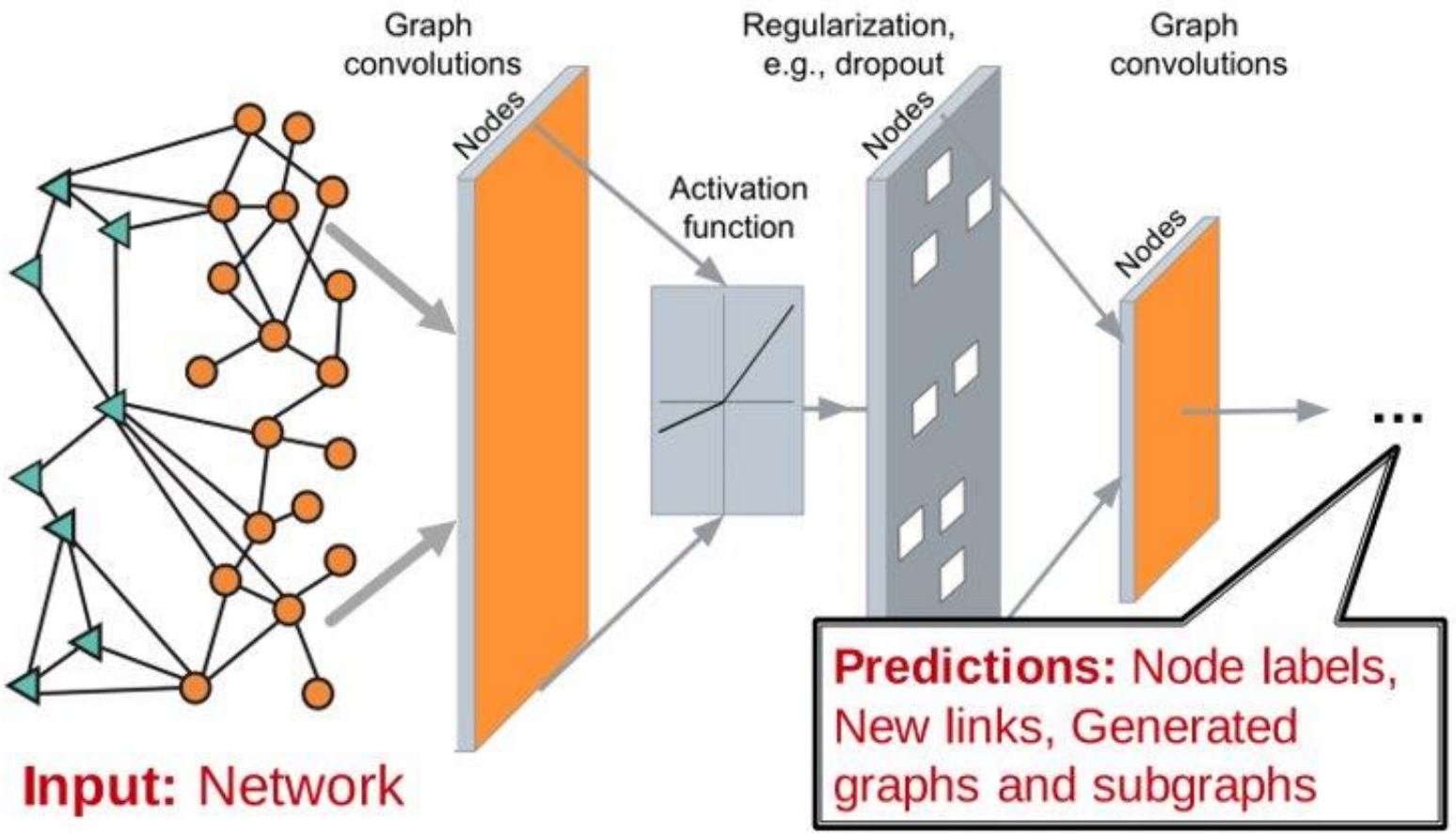
Why Graph Neural Network?



- Arbitrary size and complex topological structure (i.e., no spatial locality like grids)
- No fixed node ordering or reference point

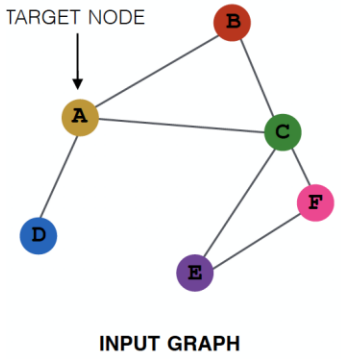
Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

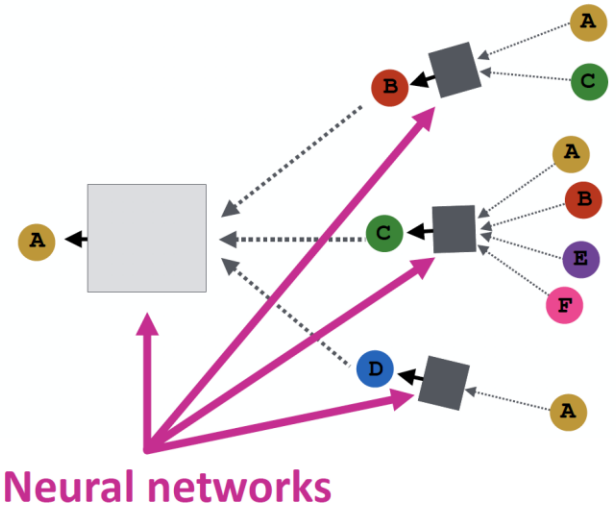
Graph Neural Network (GNN)

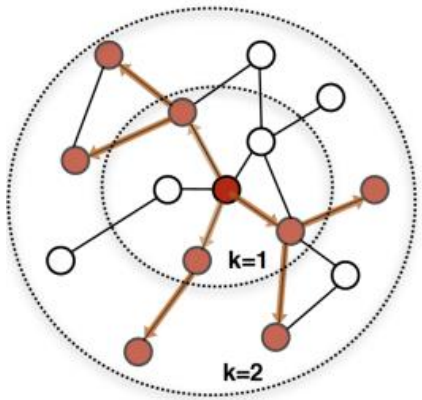


Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

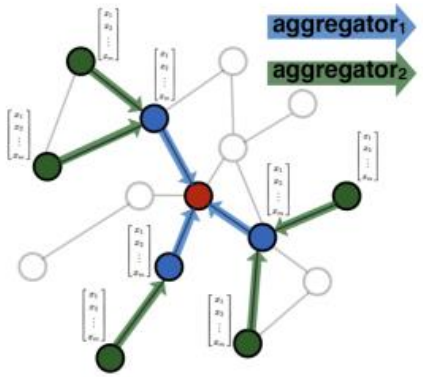
GraphSAGE : Inductive representation learning



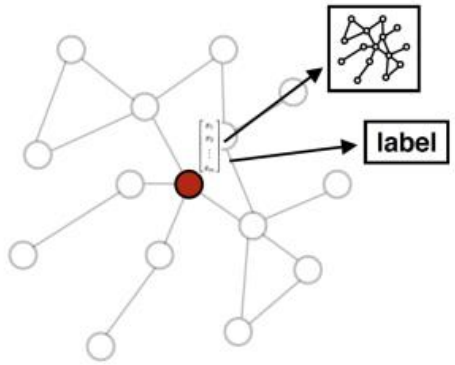




1. Sample neighborhood



2. Aggregate feature information from neighbors



3. Predict graph context and label using aggregated information

https://arxiv.org/abs/1706.02216

Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

GraphSAGE : Inductive representation learning

Algorithm 1: GraphSAGE embedding generation (i.e., forward propagation) algorithm

Input : Graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$; input features $\{\mathbf{x}_v, \forall v \in \mathcal{V}\}$; depth K; weight matrices $\mathbf{W}^k, \forall k \in \{1, ..., K\}$; non-linearity σ; differentiable aggregator functions $\text{AGGREGATE}_k, \forall k \in \{1, ..., K\}$; neighborhood function $\mathcal{N} : v \rightarrow 2^{\mathcal{V}}$

Output : Vector representations $\mathbf{z}_v$ for all $v \in \mathcal{V}$

$\mathbf{h}_v^0 \leftarrow \mathbf{x}_v, \forall v \in \mathcal{V}$;

for $k = 1...K$ **do**

  for $v \in \mathcal{V}$ **do**

    $\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow \text{AGGREGATE}_k(\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\})$; message

    $\mathbf{h}_v^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k)\right)$ update

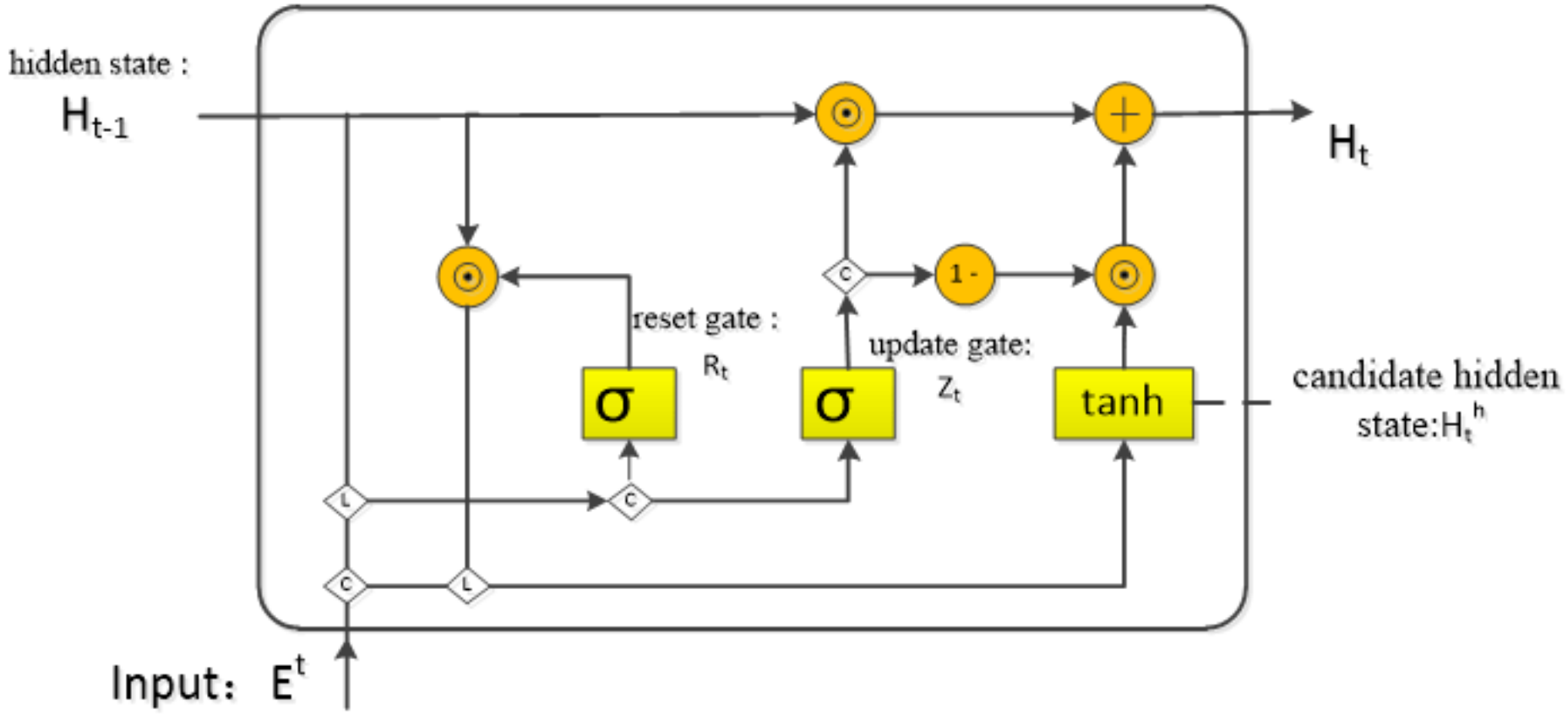
  end

  $\mathbf{h}_v^k \leftarrow \mathbf{h}_v^k / \|\mathbf{h}_v^k\|_2, \forall v \in \mathcal{V}$

end

$\mathbf{z}_v \leftarrow \mathbf{h}_v^K, \forall v \in \mathcal{V}$

Structure diagram of Gate Recurrent Unit (GRU)



Gated Graph Sequence Neural Networks (GG-NN)

The basic recurrence of the propagation model is

initialization $\mathbf{h}_v^{(1)} = [\boldsymbol{x}_v^\top, \mathbf{0}]^\top$ (1)

message $\mathbf{a}_v^{(t)} = \mathbf{A}_{v:}^\top \left[\mathbf{h}_1^{(t-1)\top} \dots \mathbf{h}_{|\mathcal{V}|}^{(t-1)\top}\right]^\top + \mathbf{b}$ (2)

Update gate $\mathbf{z}_v^t = \sigma\left(\mathbf{W}^z \mathbf{a}_v^{(t)} + \mathbf{U}^z \mathbf{h}_v^{(t-1)}\right)$ (3)

Reset gate $\mathbf{r}_v^t = \sigma\left(\mathbf{W}^r \mathbf{a}_v^{(t)} + \mathbf{U}^r \mathbf{h}_v^{(t-1)}\right)$ (4)

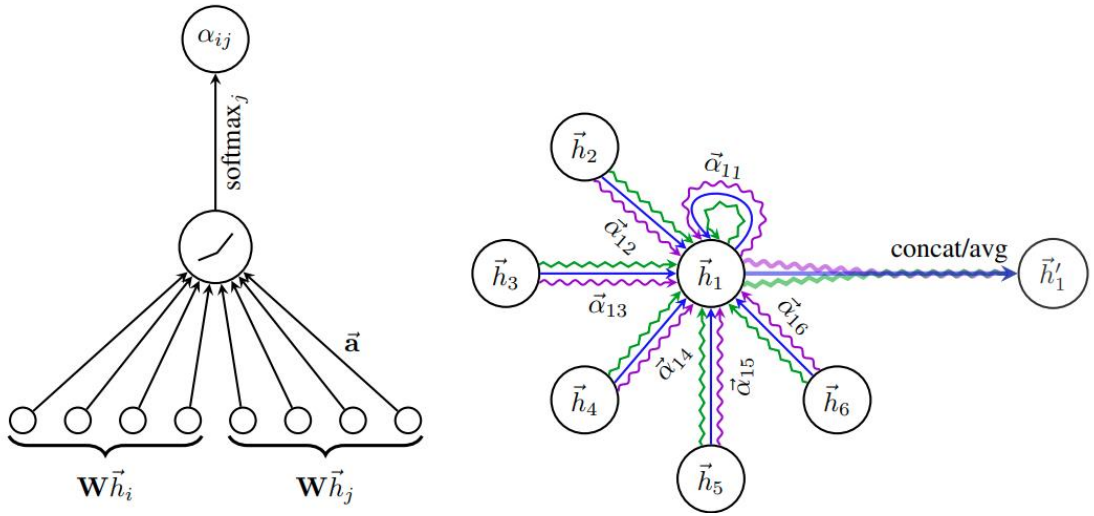
$\widetilde{\mathbf{h}_v^{(t)}} = \tanh\left(\mathbf{W}\mathbf{a}_v^{(t)} + \mathbf{U}\left(\mathbf{r}_v^t \odot \mathbf{h}_v^{(t-1)}\right)\right)$ (5)

$\mathbf{h}_v^{(t)} = (1 - \mathbf{z}_v^t) \odot \mathbf{h}_v^{(t-1)} + \mathbf{z}_v^t \odot \widetilde{\mathbf{h}_v^{(t)}}.$ (6)

- GG-NN adjusts **the update rate** of node states at each convolution layer

https://arxiv.org/abs/1511.05493

Graph Attention Network (GAT)

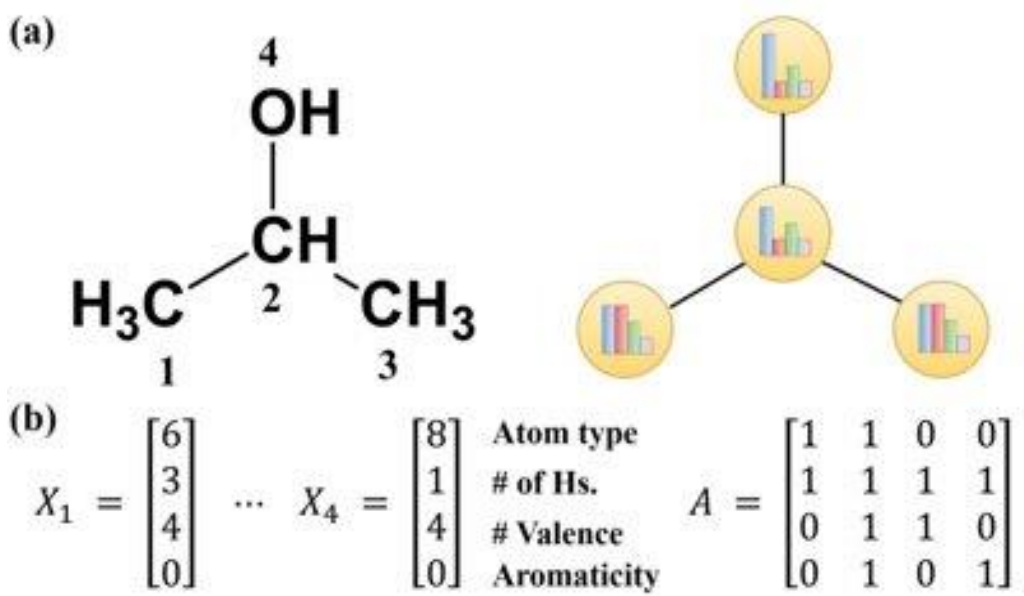


$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\vec{\mathbf{a}}^T[\mathbf{W}\vec{h}_i \| \mathbf{W}\vec{h}_j]\right)\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(\text{LeakyReLU}\left(\vec{\mathbf{a}}^T[\mathbf{W}\vec{h}_i \| \mathbf{W}\vec{h}_k]\right)\right)}$$

aij denotes an attention coefficient, which measures the **importance** of the j-th node in updating the I-th state

https://arxiv.org/abs/1710.10903

Application of GNN



Cheminformatics

- quantum/molecular property
- antibiotics
- Toxicity

Bioinformatics

- Protein structure prediction
- Protein function prediction
- Kcat prediction

A Message Passing Neural Network predicts **quantum properties** of an organic molecule

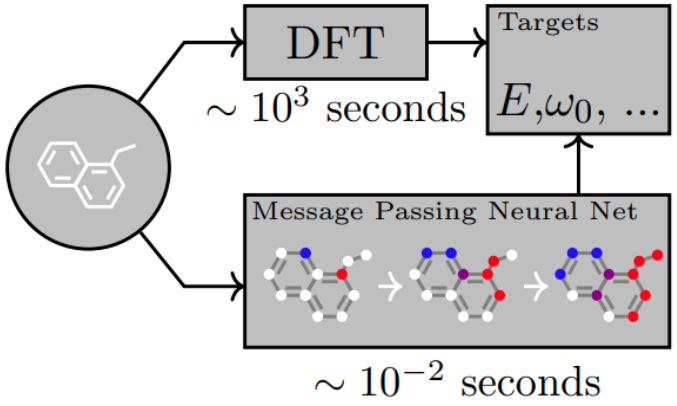


Table 2. Comparison of Previous Approaches (left) with MPNN baselines (middle) and our methods (right)

Target	BAML	BOB	CM	ECFP4	HDAD	GC	GG-NN	DTNN	enn-s2s	enn-s2s-ens5
mu	4.34	4.23	4.49	4.82	3.34	0.70	1.22	-	**0.30**	0.20
alpha	3.01	2.98	4.33	34.54	1.75	2.27	1.55	-	**0.92**	0.68
HOMO	2.20	2.20	3.09	2.89	1.54	1.18	1.17	-	**0.99**	0.74
LUMO	2.76	2.74	4.26	3.10	1.96	1.10	1.08	-	**0.87**	0.65
gap	3.28	3.41	5.32	3.86	2.49	1.78	1.70	-	**1.60**	1.23
R2	3.25	0.80	2.83	90.68	1.35	4.73	3.99	-	**0.15**	0.14
ZPVE	3.31	3.40	4.80	241.58	1.91	9.75	2.52	-	**1.27**	1.10
U0	1.21	1.43	2.98	85.01	0.58	3.02	0.83	-	**0.45**	0.33
U	1.22	1.44	2.99	85.59	0.59	3.16	0.86	-	**0.45**	0.34
H	1.22	1.44	2.99	86.21	0.59	3.19	0.81	-	**0.39**	0.30
G	1.20	1.42	2.97	78.36	0.59	2.95	0.78	.84[2]	**0.44**	0.34
Cv	1.64	1.83	2.36	30.29	0.88	1.45	1.19	-	**0.80**	0.62
Omega	0.27	0.35	1.32	1.47	0.34	0.32	0.53	-	**0.19**	0.15
Average	2.17	2.08	3.37	53.97	1.35	2.59	1.36	-	**0.68**	0.52

https://arxiv.org/abs/1704.01212

Challenge in molecular graph representation

Challenges

- Different molecular properties depends on their **local chemical environments**

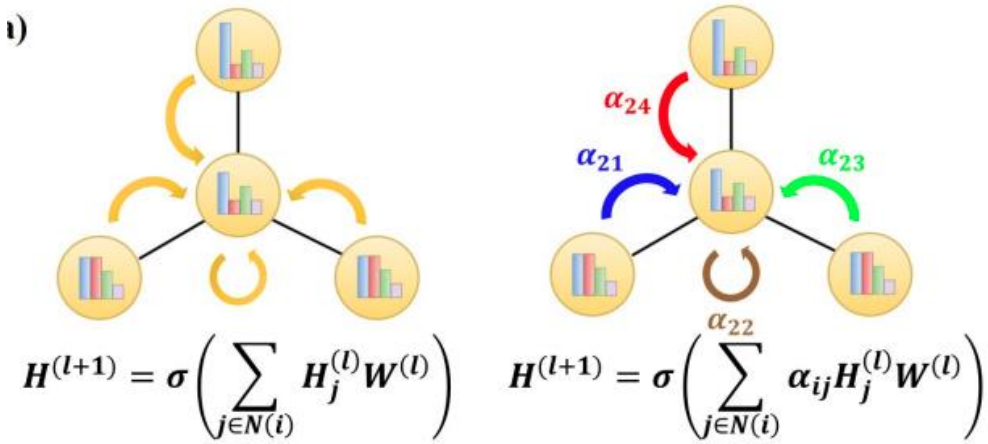
Solution

- Add adaptive **attention weights** depends on their chemical environments
- Adjust **the update rate** of node states at each convolution layers

Attention mechanism

$$\alpha_{ij} = \frac{e_{ij}}{\sum_{k \in N(i)} e_{ik}} = \frac{\sigma(MLP[H_i W, H_j W])}{\sum_{k \in N(i)} \sigma(MLP[H_i W, H_k W])},$$

the importance rate of each adjacent node



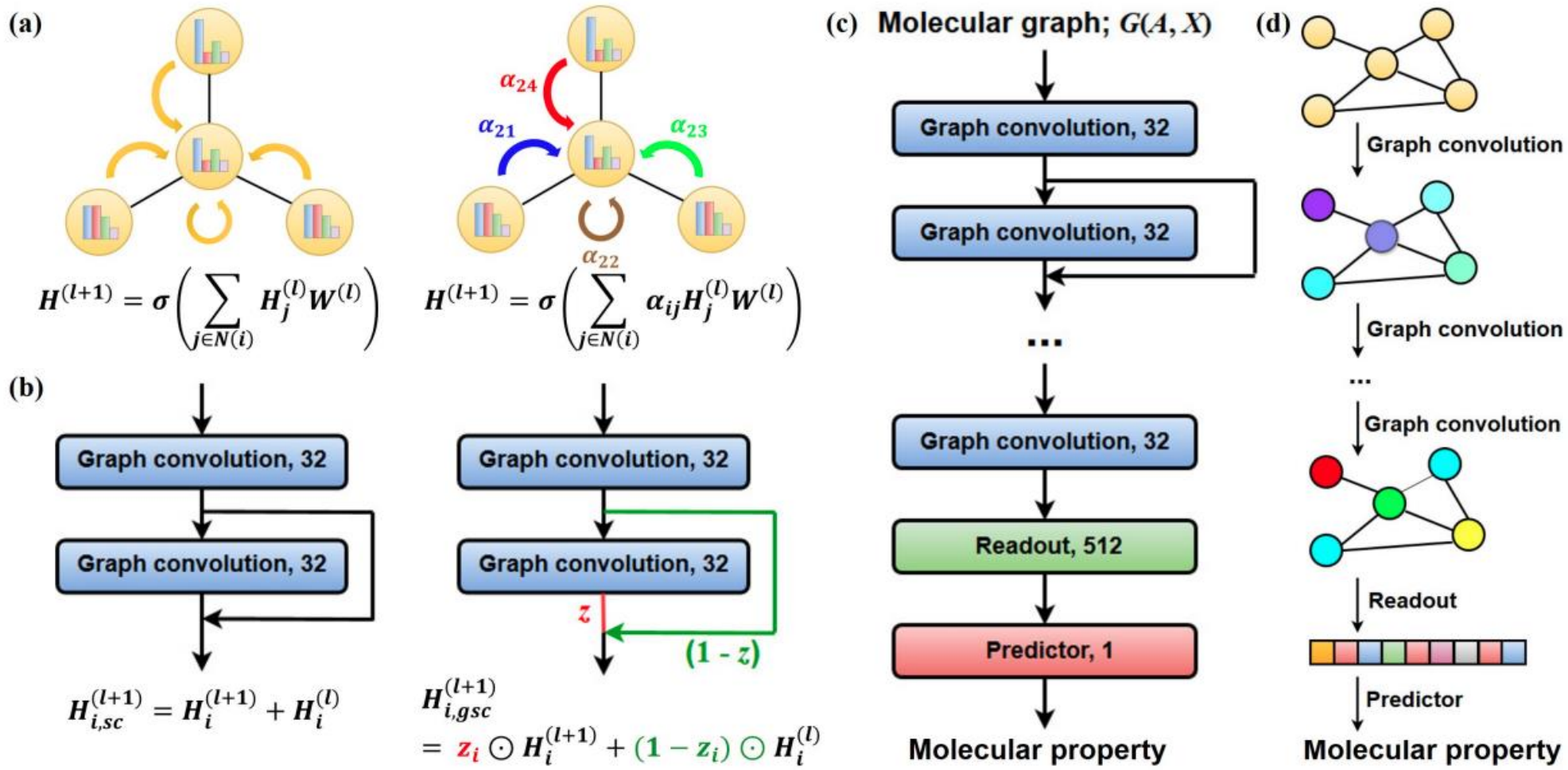
$$\alpha_{ij}^{(l)} = \sigma((H_i^{(l)} W^{(l)}) C^{(l)} (H_j^{(l)} W^{(l)})^T),$$

the attention coefficient should be analogous to **the interaction strength** between an atom pair (i, j)

* C (l) is a coupling matrix. Note that the coupling matrix may correspond to the dictionaries containing pairwise interactions in Shang's model.

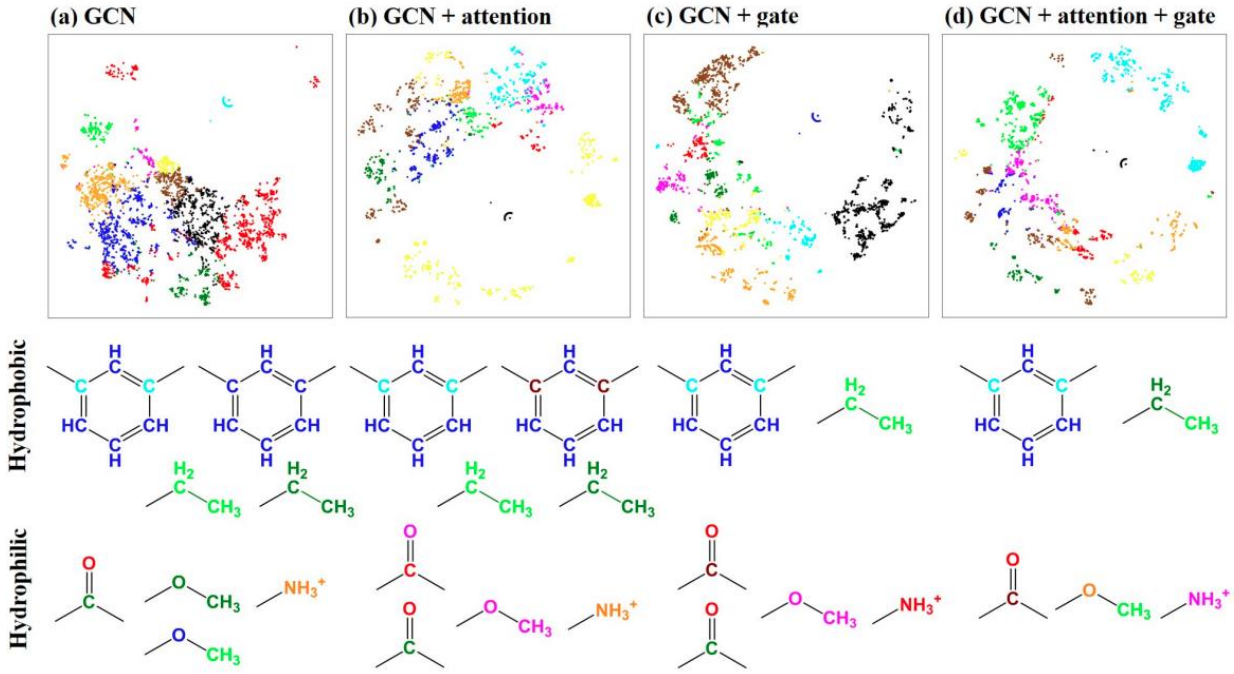
https://arxiv.org/abs/1805.10988

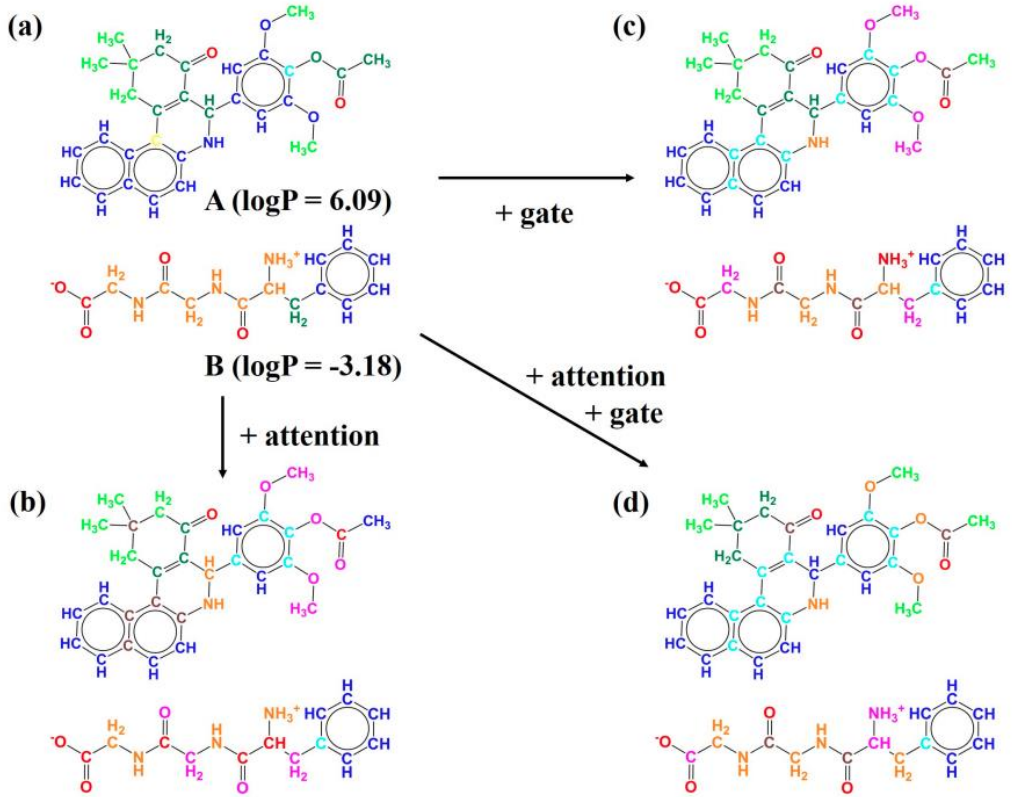
GNN learns **molecular structure-property relationships**



https://arxiv.org/abs/1805.10988

GNN learns **molecular structure-property relationships**





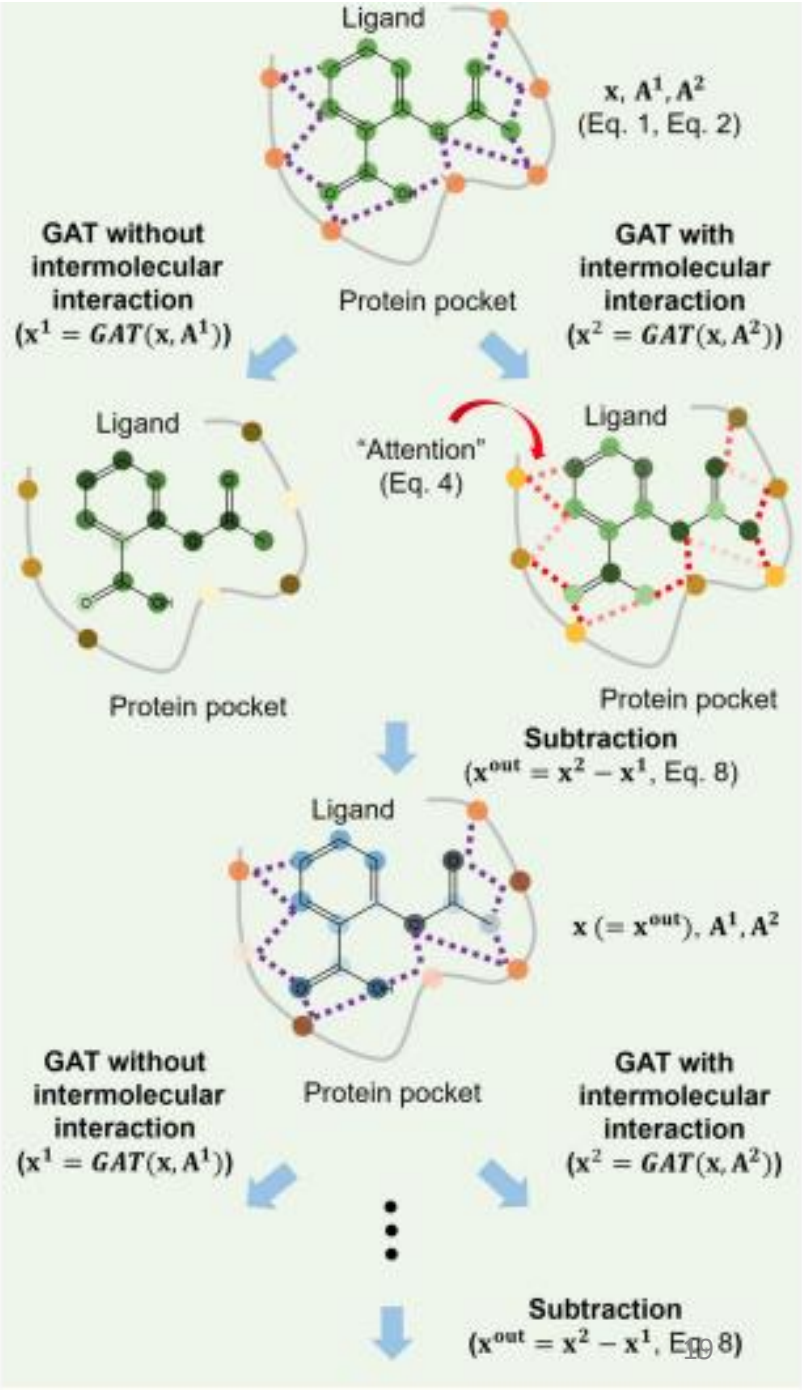
https://arxiv.org/abs/1805.10988

Drug-Target interaction

Table 3. AUROC, Adjusted LogAUC, PRAUC, Sensitivity, Specificity, and Balanced Accuracy of Our Model, Docking, and Other Deep Learning Models[a]

	AUROC	adjusted LogAUC	PRAUC	sensitivity	specificity	balanced accuracy
ours	**0.968**	**0.633**	**0.697**	**0.826**	0.967	**0.909**
ours w/o attention	0.936	0.577	0.623	0.758	**0.970**	0.888
docking	0.689	0.153	0.016			
Atomnet[19]	0.855	0.321				
Ragoza et al.[22]	0.868					
Torng et al.[40]	0.886					
Gonczarek et al.[17]	0.904					

[a]We note that the division of the training and test sets may be different for each model.



https://pubs.acs.org/doi/10.1021/acs.jcim.9b00387

Challenge in molecular graph representation

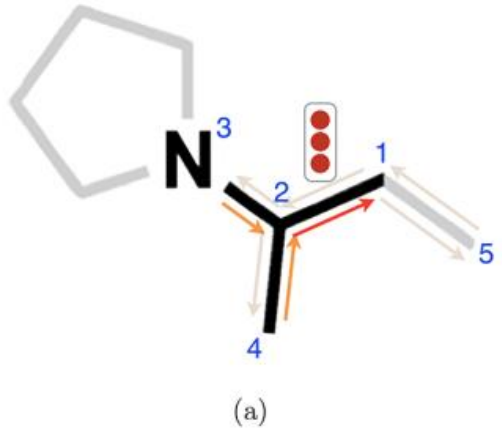
Challenges

- a poor molecular representation by memorizing the molecular scaffolds in training data **fails to generalize** to new ones.

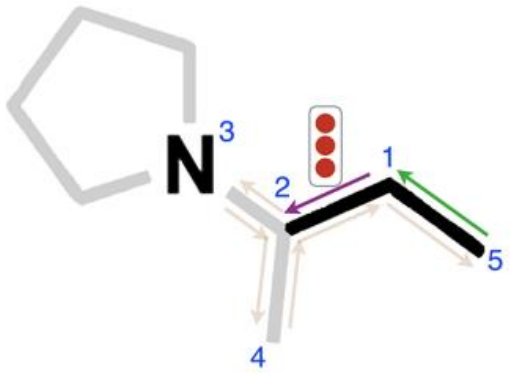
Solution

- A hybrid representation that **combines** convolutions and descriptors
- Convolutions centered on **bonds (edge)**, instead of atoms (node)

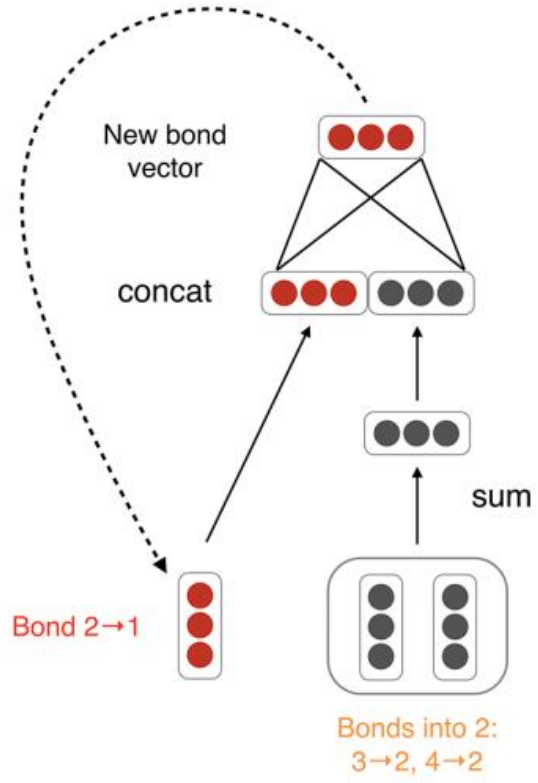
Edge-based molecular representation learning for chemical property prediction



(a)



(b)



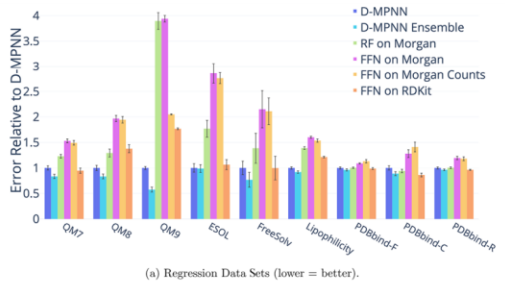
(c)

Edge-based aggregation

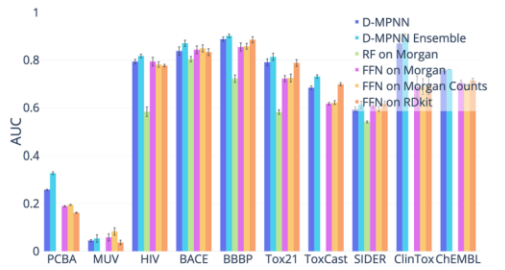
$$m_{vw}^{t+1} = \sum_{k \in \{N(v) \backslash w\}} M_t(x_v, x_k, h_{kv}^t)$$

Edge-based update

$$h_{vw}^{t+1} = U_t(h_{vw}^t, m_{vw}^{t+1})$$



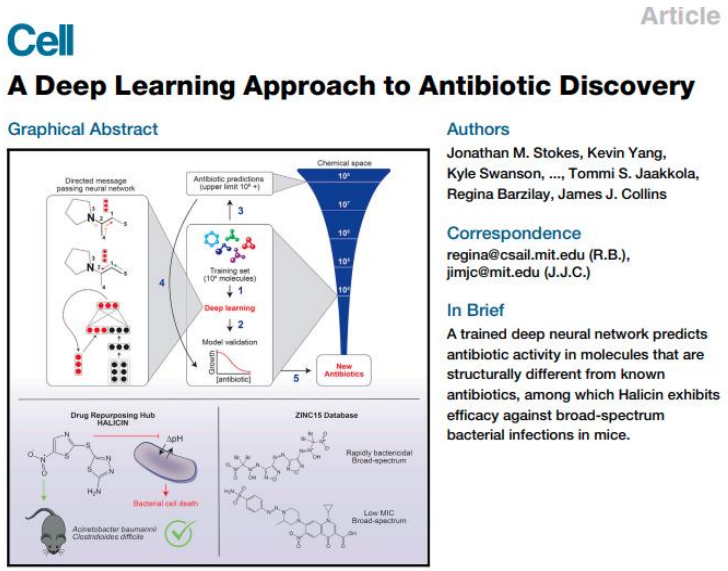
(a) Regression Data Sets (lower = better).



(b) Classification Data Sets (higher = better).

https://pubs.acs.org/doi/abs/10.1021/acs.jcim.9b00237

Edge-based molecular representation learning results in **antibiotic discovery**



Article

Cell

A Deep Learning Approach to Antibiotic Discovery

Graphical Abstract

Authors

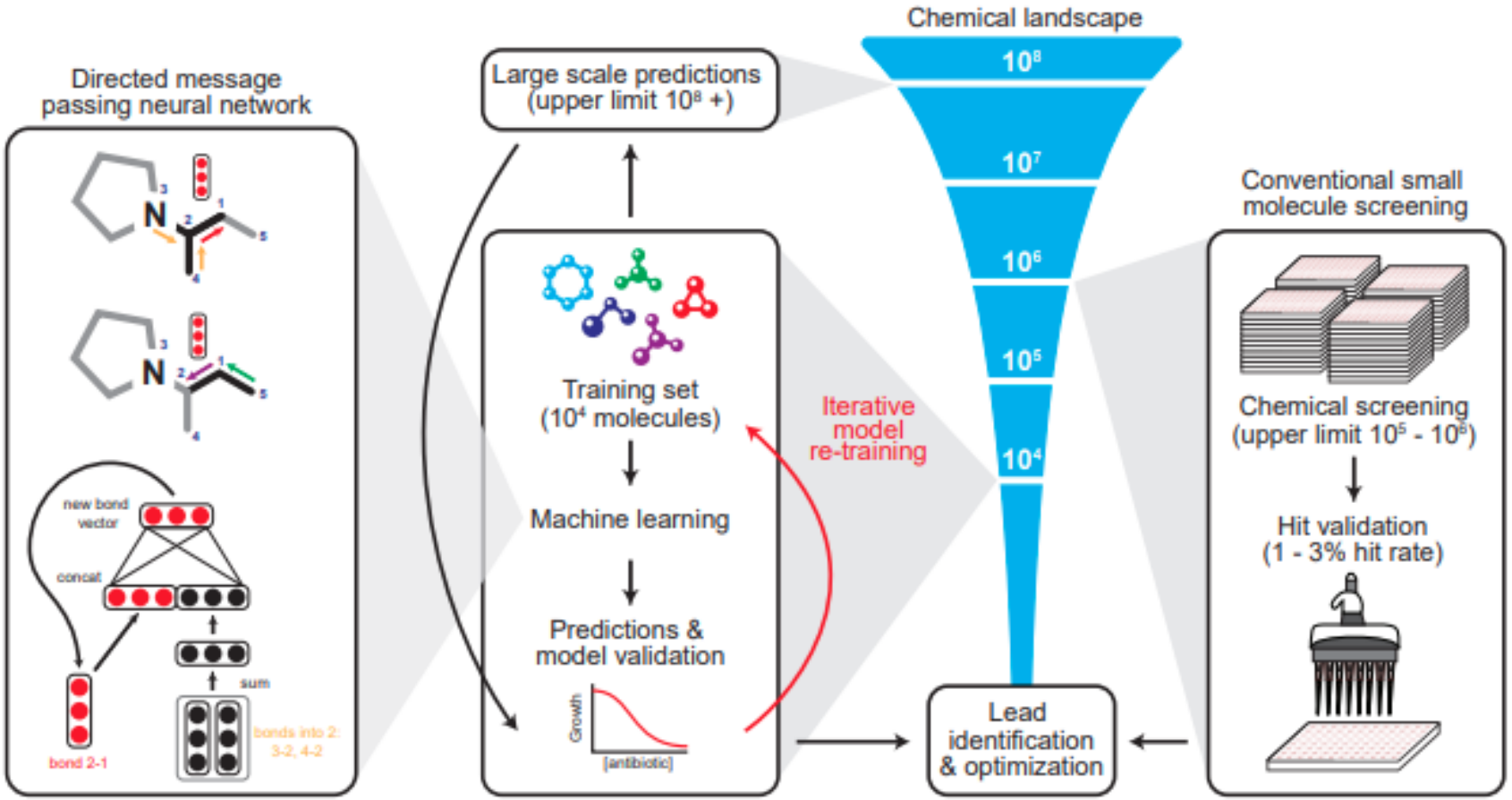
Jonathan M. Stokes, Kevin Yang, Kyle Swanson, ..., Tommi S. Jaakkola, Regina Barzilay, James J. Collins

Correspondence

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

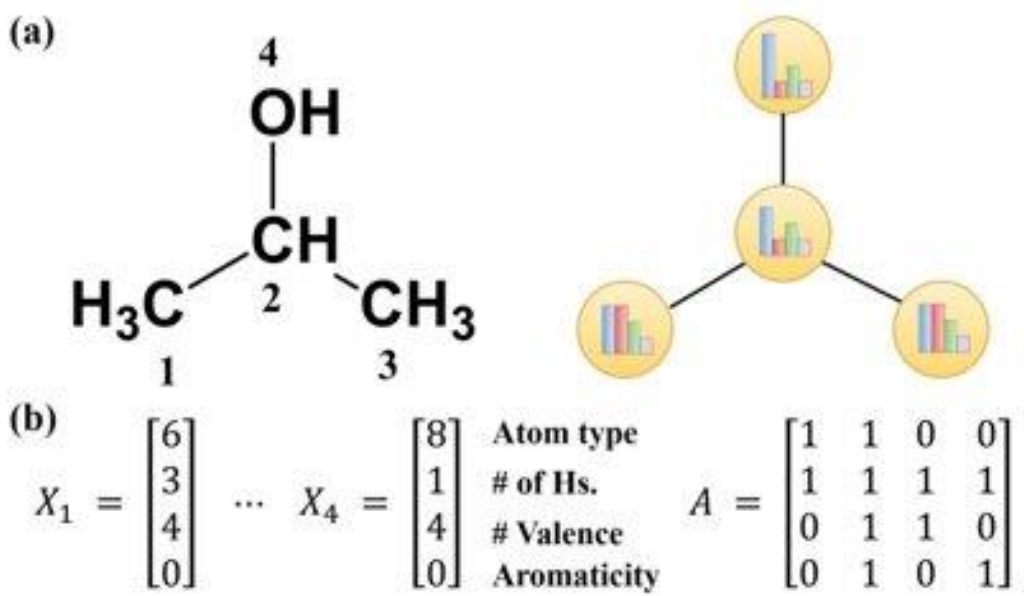
In Brief

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.



Cell, Volume 180, Issue 4, 20 February 2020, Pages 688-702.e13

Application of GNN



Cheminformatics

- quantum/molecular property
- antibiotics
- Toxicity

Bioinformatics

- Protein structure prediction
- Protein function prediction
- Kcat prediction

Article

Highly accurate protein structure prediction with AlphaFold

https://doi.org/10.1038/s41586-021-03819-2

Received: 11 May 2021

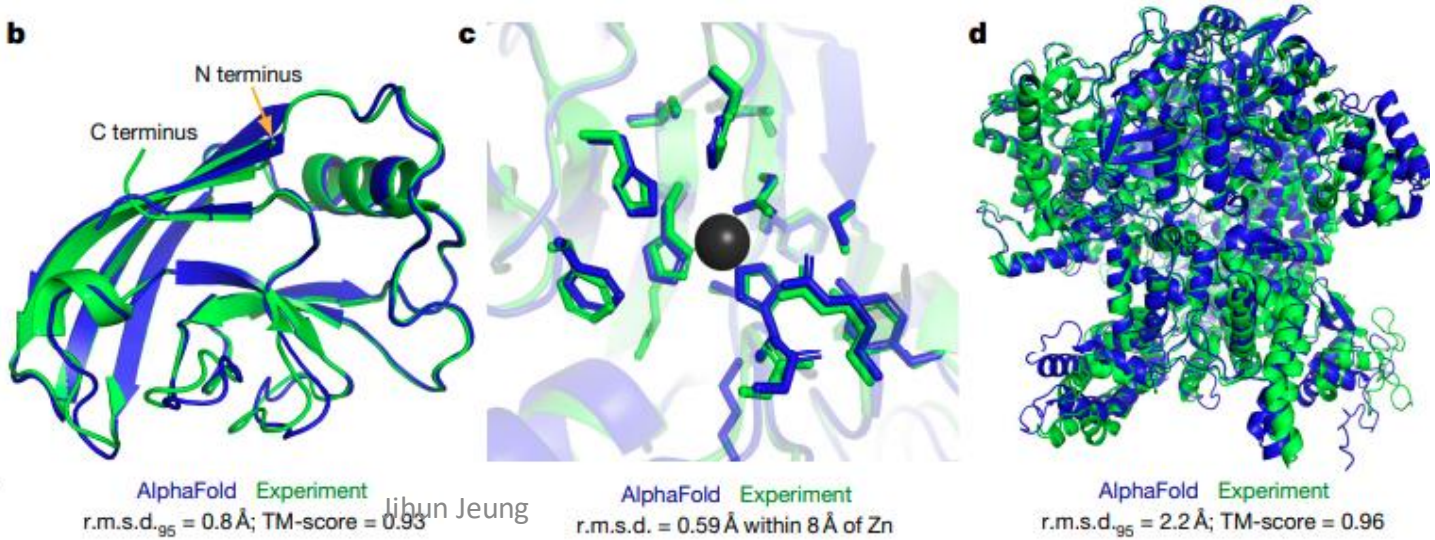
Accepted: 12 July 2021

Published online: 15 July 2021

Open access

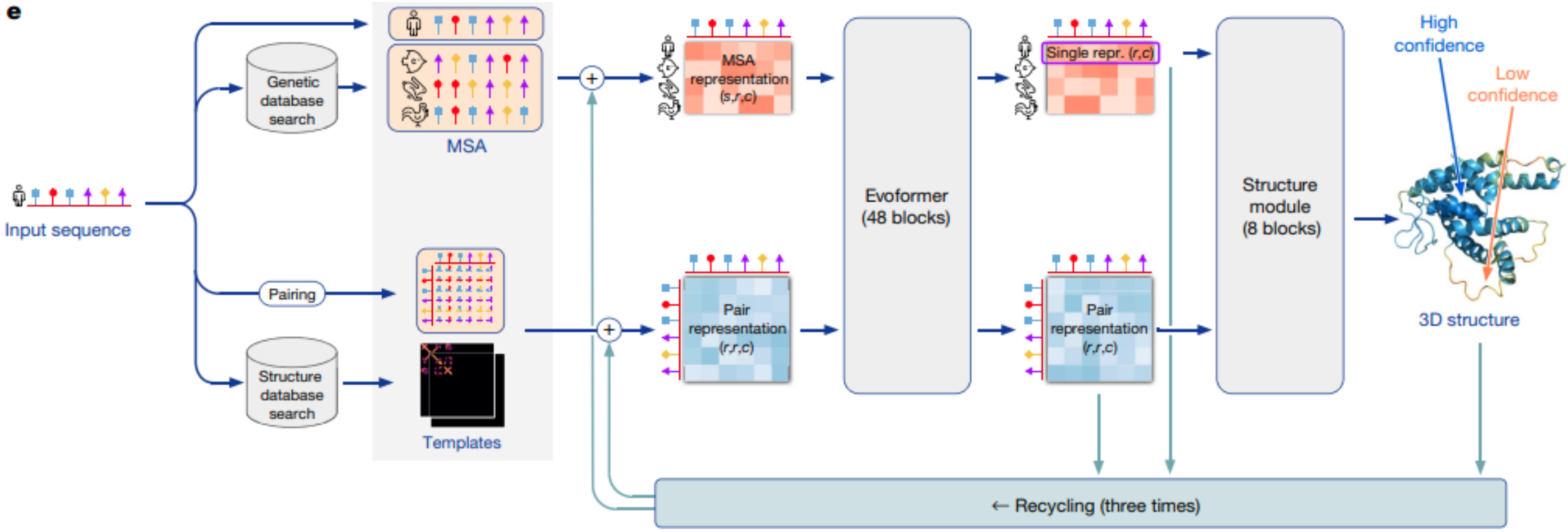
Check for updates

John Jumper[1,4]✉, Richard Evans[1,4], Alexander Pritzel[1,4], Tim Green[1,4], Michael Figurnov[1,4], Olaf Ronneberger[1,4], Kathryn Tunyasuvunakool[1,4], Russ Bates[1,4], Augustin Žídek[1,4], Anna Potapenko[1,4], Alex Bridgland[1,4], Clemens Meyer[1,4], Simon A. A. Kohl[1,4], Andrew J. Ballard[1,4], Andrew Cowie[1,4], Bernardino Romera-Paredes[1,4], Stanislav Nikolov[1,4], Rishub Jain[1,4], Jonas Adler[1], Trevor Back[1], Stig Petersen[1], David Reiman[1], Ellen Clancy[1], Michal Zielinski[1], Martin Steinegger[2,3], Michalina Pacholska[1], Tamas Berghammer[1], Sebastian Bodenstein[1], David Silver[1], Oriol Vinyals[1], Andrew W. Senior[1], Koray Kavukcuoglu[1], Pushmeet Kohli[1] & Demis Hassabis[1,4]✉



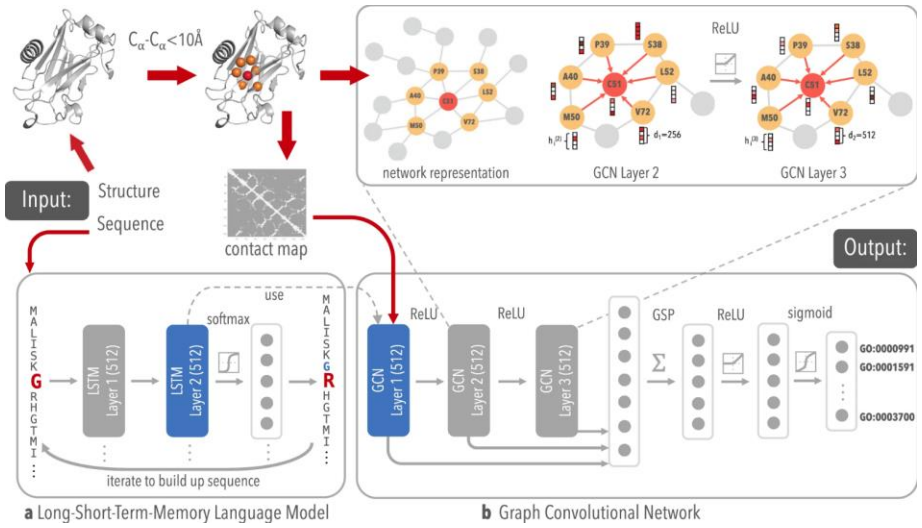
https://www.nature.com/articles/s41586-021-03819-2

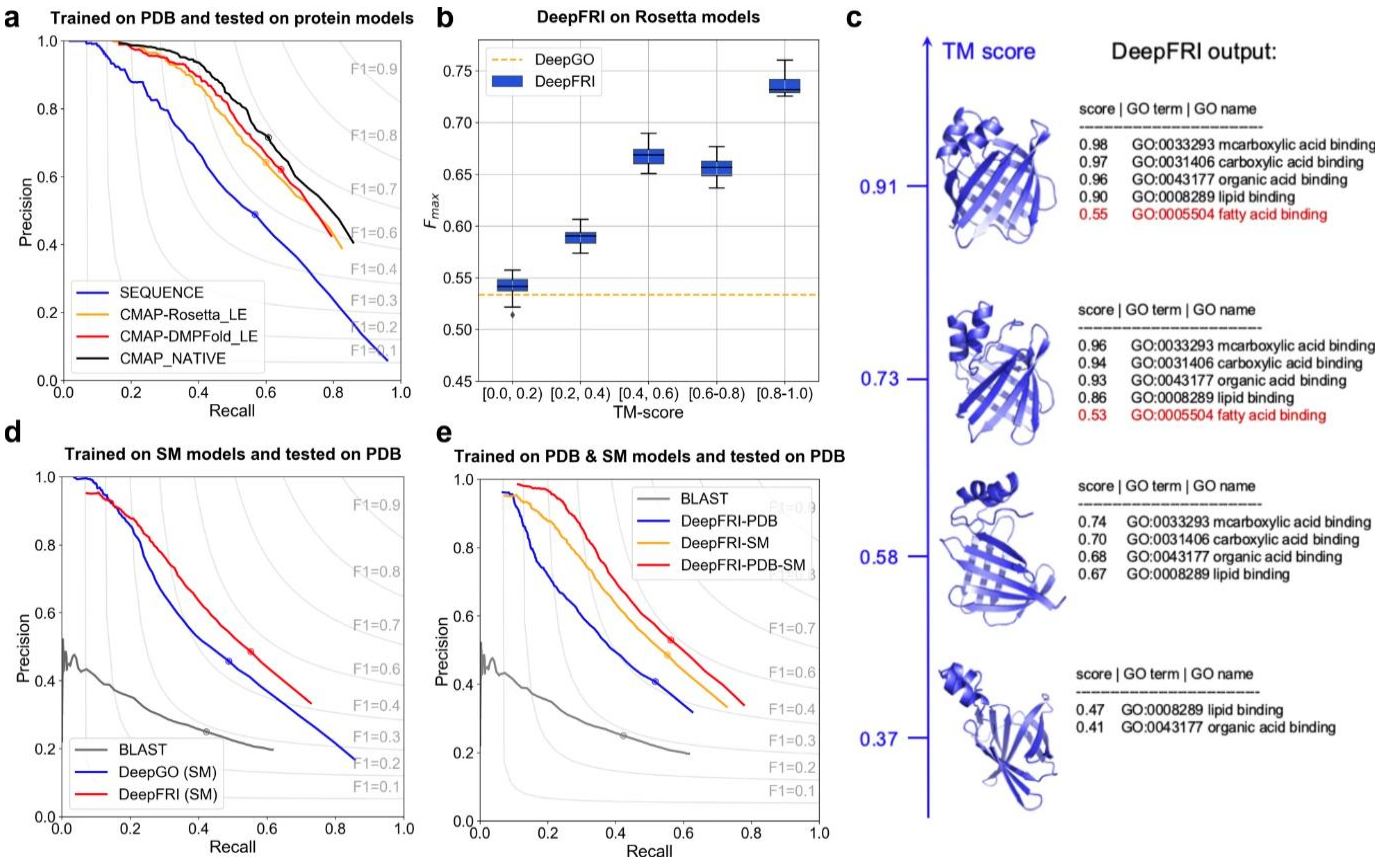
GNN increase the predicton accuracy in AlphaFold



https://www.nature.com/articles/s41586-021-03819-2

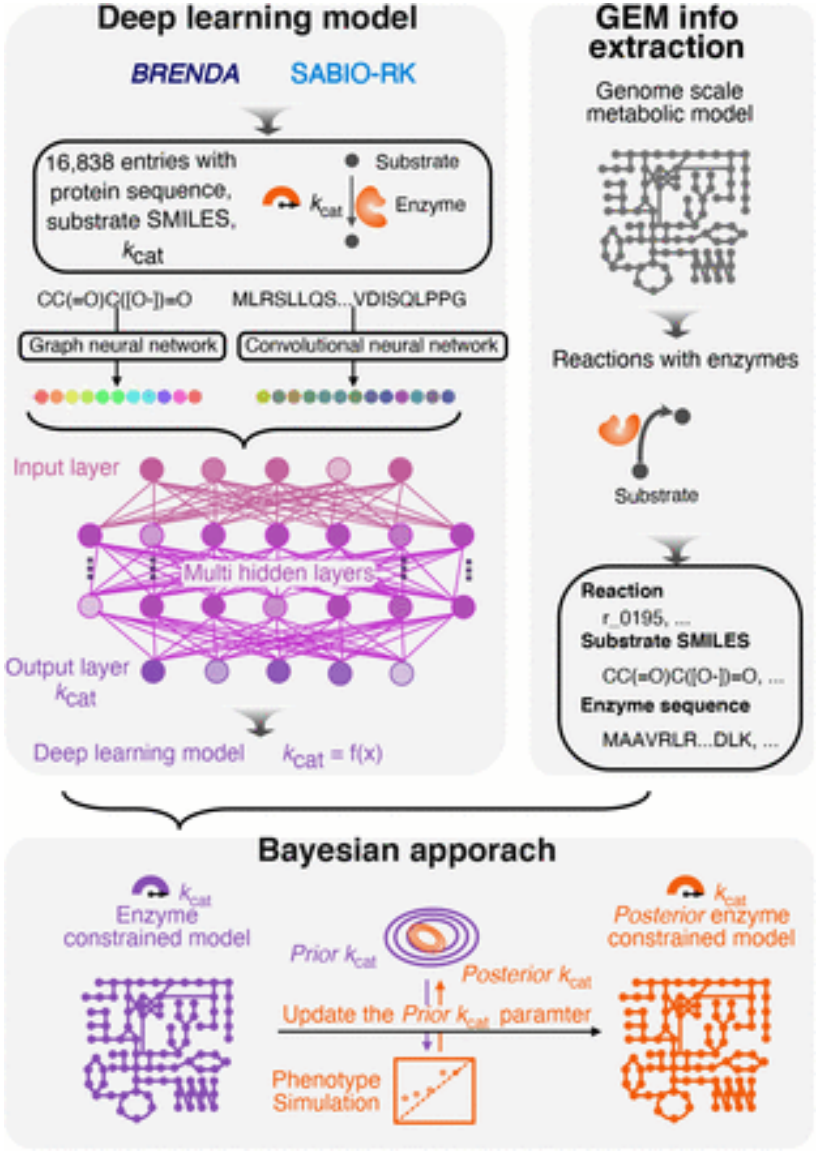
Structure-based **protein function** prediction

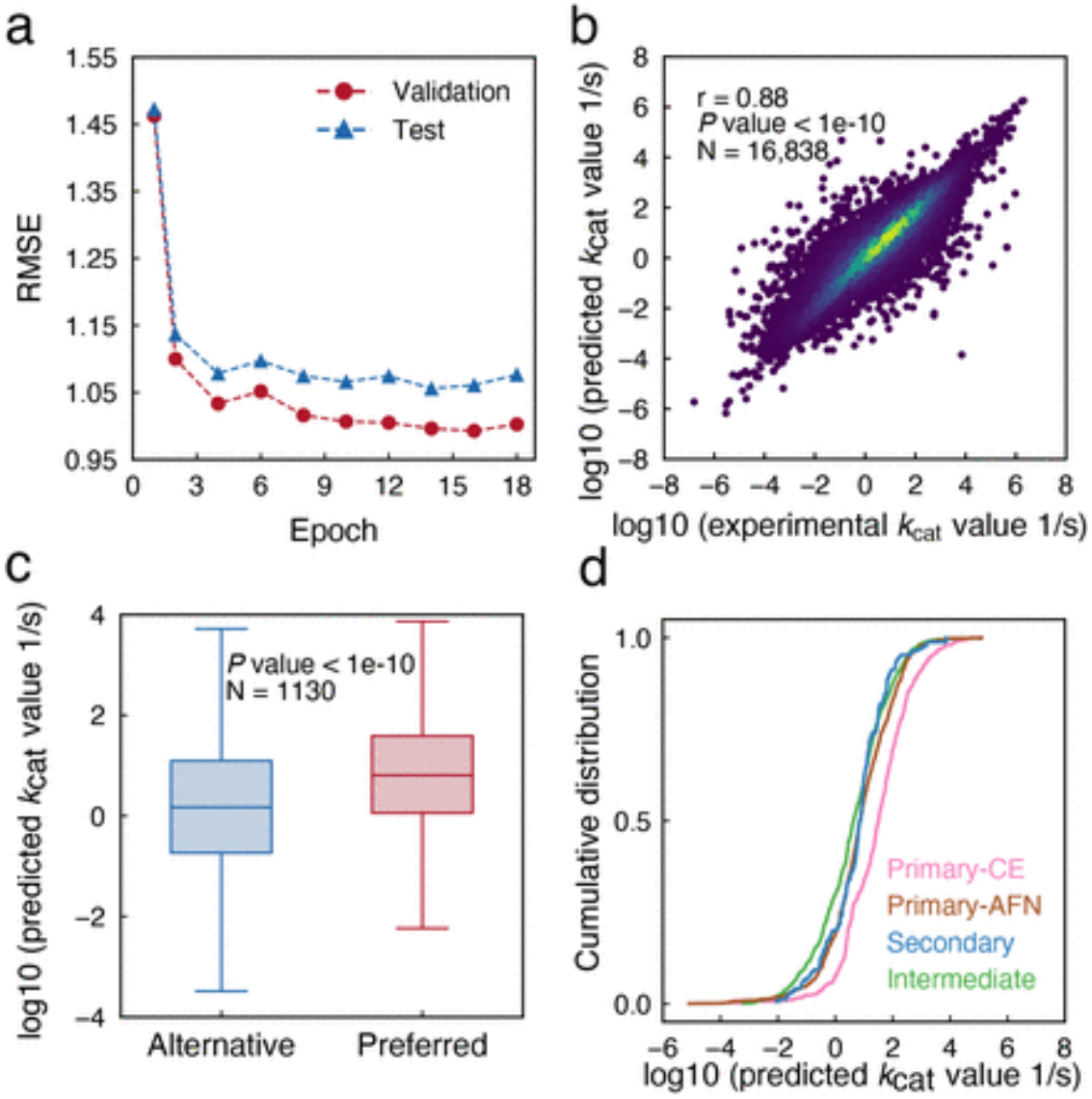




https://www.nature.com/articles/s41467-021-23303-9

Structure-based **Kcat** prediction





https://www.biorxiv.org/content/10.1101/2021.08.06.455417v1.full

Conclusion

- Graph Neural Network (GNN) has **arbitrary size** and **complex topological structure** (i.e., no spatial locality like grids).
- GNN can represent **the molecular properties** from its structure without feature engineering.
- A model of GNN is **adaptive** and **task-dependent**.
  - Understanding **mathematical framework** (learning process) of GNN is crucial to design model architecture.