ETH zürich

Applications of AI in Chemistry & Biology

David Graber
david.graber@sam.math.ethz.ch
ETH Zürich

Lecture Outline

1. **Introduction - Proteins and Small Molecules**
2. **Text-based AI models**
3. **Protein Structure Prediction**
   - AlphaFold2
   - Evolutionary Scale Modelling (ESM)
   - Performance Comparison
4. **Structure-based AI models**
   - 3D-Convolutional Neural Networks
   - Introduction to Graphs and Graph Neural Networks
   - Graph-based models

5. **Generative AI for *de novo* protein design**

Introduction

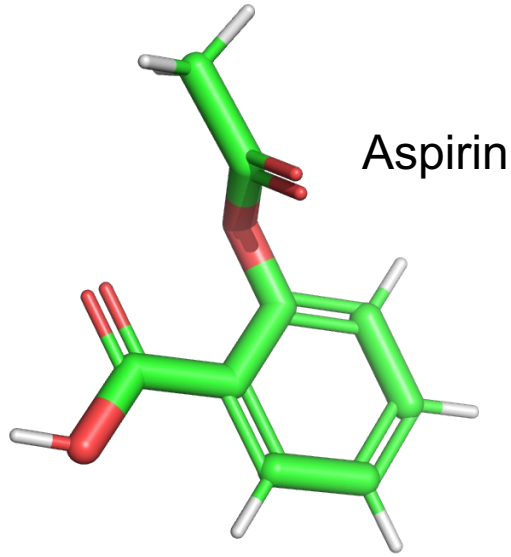
→ Proteins and Small Molecules
→ Engineering of Protein and Small Molecules

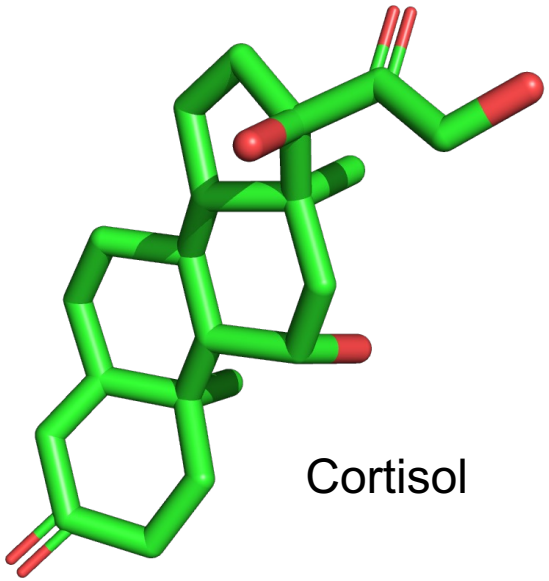
Introduction

Small molecules

- Organic compounds with a low molecular weight, usually less than 100 atoms
- Organisms produce small molecules naturally, as metabolites or signalling molecules (e.g. cortisol)
- Can easily diffuse across cell membranes due to their small size
- Often interact with proteins, influencing their function
- Most drugs are small molecules

- Their structure usually allows for **easy synthesis in the laboratory**


Aspirin

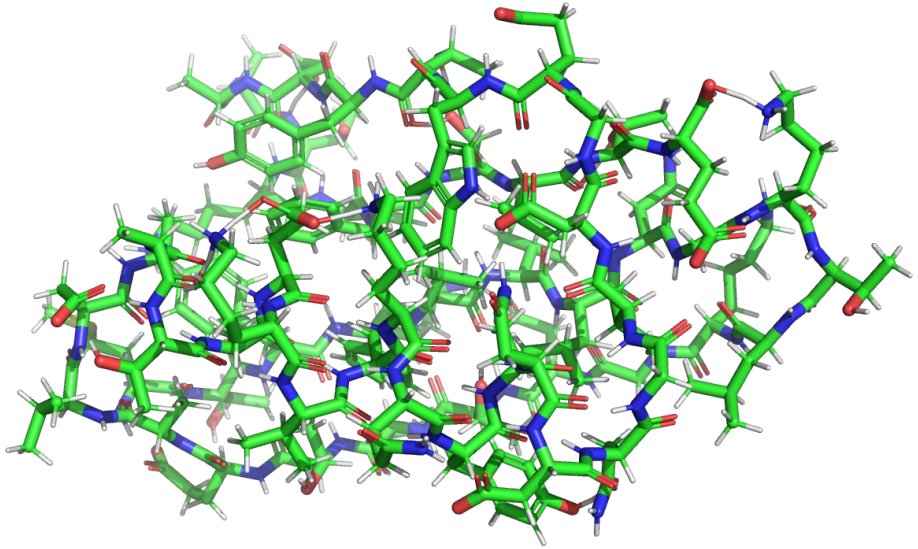

Cortisol

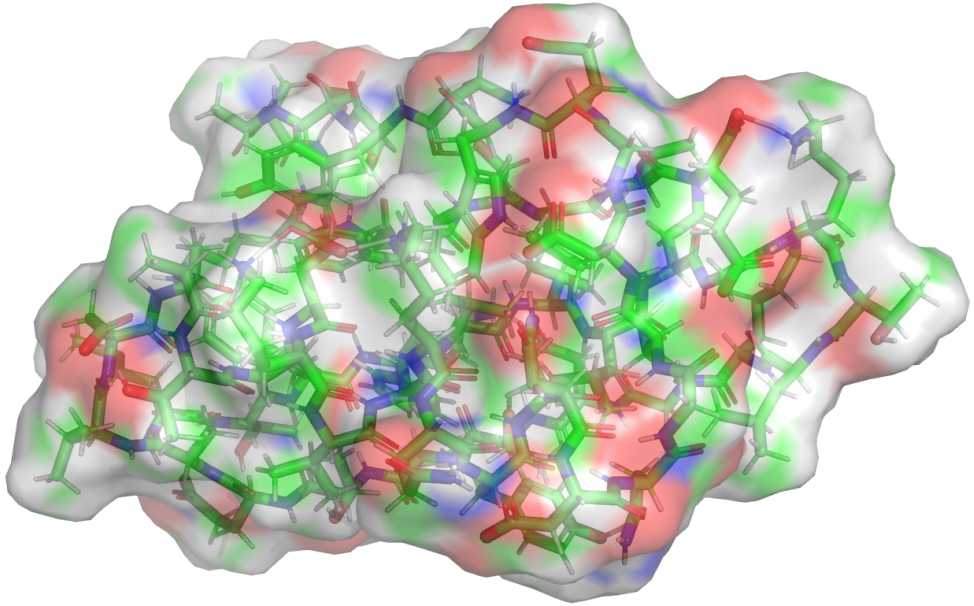
Introduction

Proteins

- Large, chain-like organic molecules made of a fixed set of building blocks
- These chains fold into a defined 3D structure known as the native conformation
- Proteins have many different functions
  - Enzymes catalyse chemical reactions
  - Structural components
  - Signalling molecules
  - Transporters

The building blocks of cells and play key roles in almost every function necessary for life





Proteins

Each protein is a linear chain of amino acids

There are twenty standard amino acids, which are the fundamental building blocks of all proteins

Backbone
Identical for all amino acids (the links of the chain)

- Amino group ($-H_2N$)
- Carboxyl group (-COOH)

Side chains
Varies between amino acids and determines their chemical properties

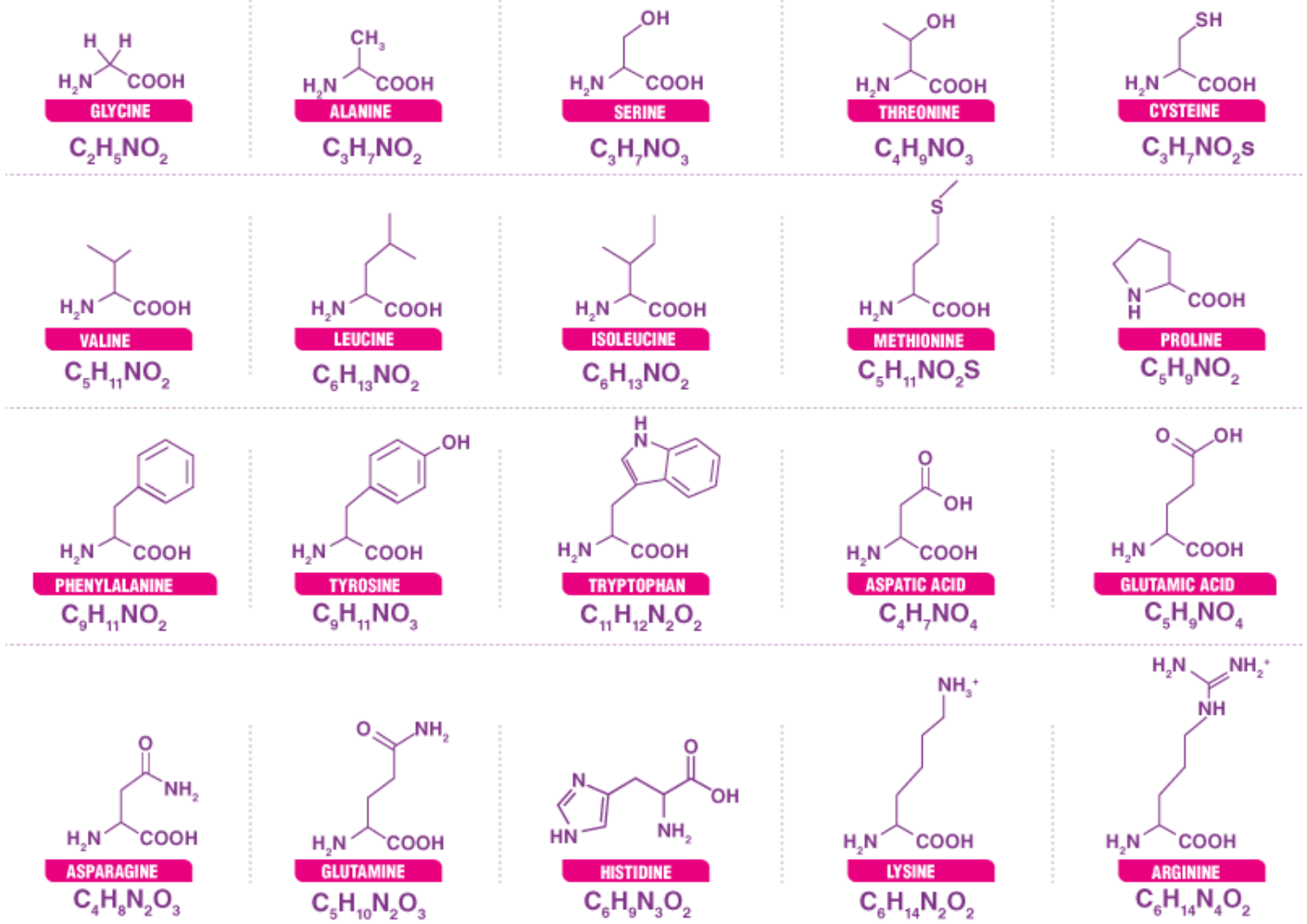


Image Credits: https://byjus.com/biology/amino-acids/

From DNA to Proteins

The genetic code of an organism dictates the precise sequence of amino acids in proteins

- Each triplet of DNA nucleotides (codons) encodes for a specific amino acid

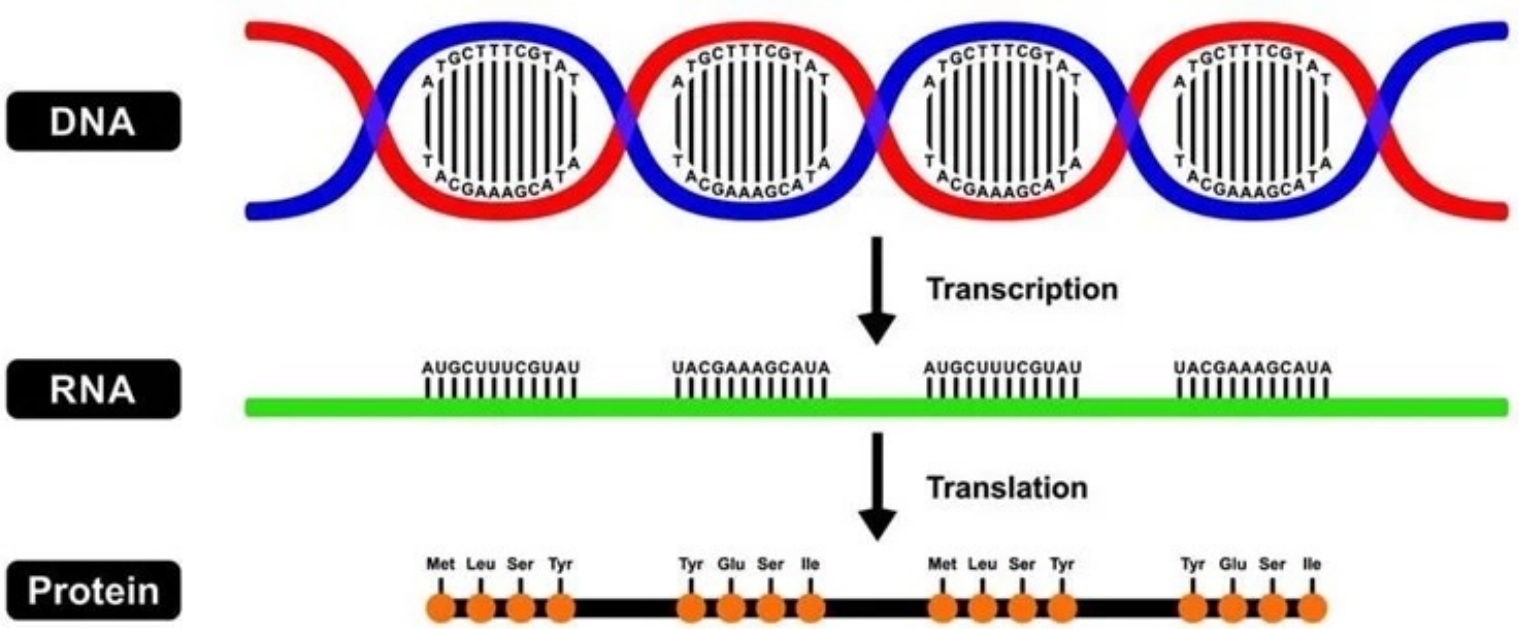
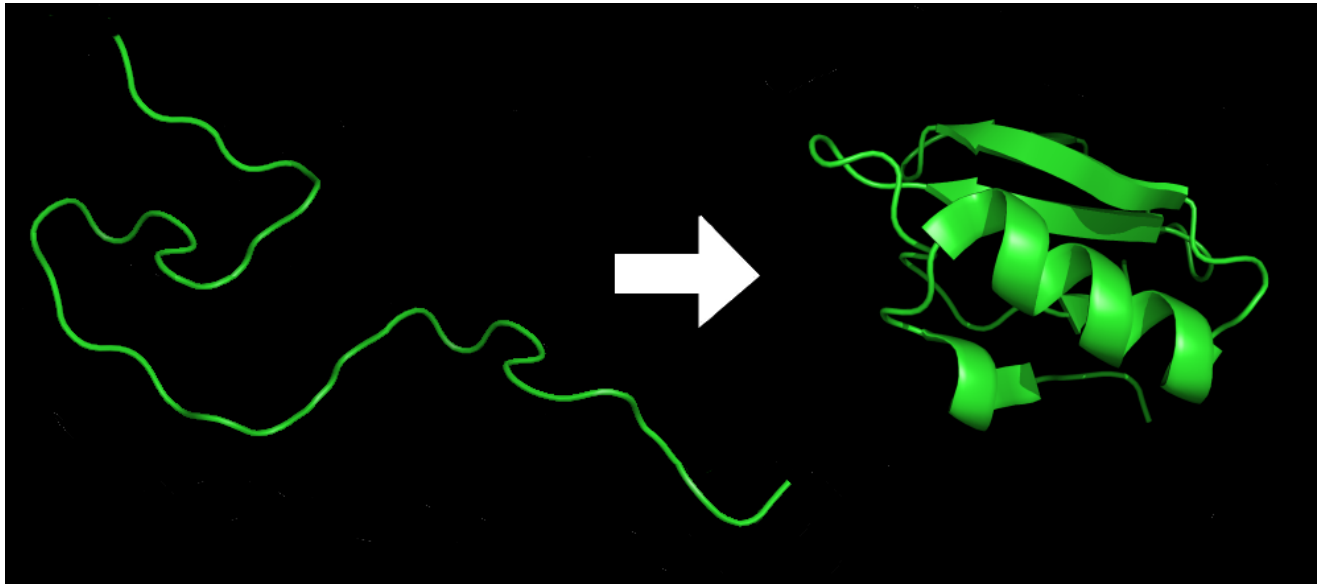


Image Credits: https://www.azolifesciences.com/article/From-DNA-to-Protein3b-The-Central-Dogma-of-Molecular-Biology.aspx

Protein Folding

After synthesis, the linear chain of amino acids folds into a more ordered three-dimensional structure, which defined the proteins biological function



- The folded structure is defined by the sequence of amino acids
- Folding is a spontaneous process of energy minimization that is guided by interactions between the amino acid such as hydrophobic interactions, hydrogen bonds and van der Waals forces

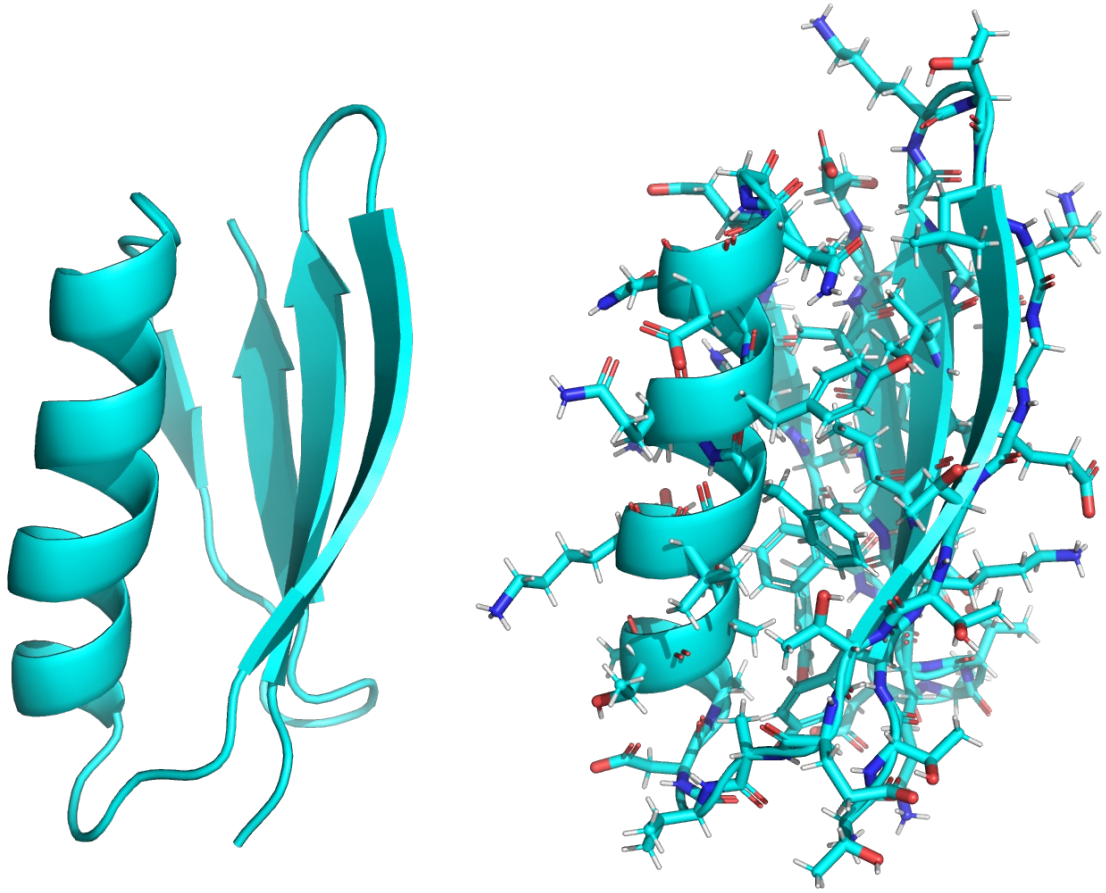
Image Credits: https://en.wikipedia.org/wiki/Protein_folding

Protein Structure

Primary Structure
The sequence of amino acids in the chain

Secondary Structure
Regularly repeating local structures stabilized by hydrogen bonds. The most common examples are the α-helix and the β-sheet

Tertiary Structure
The spatial relationship of the secondary structures to one another, which defines the overall shape of the protein molecule



Protein Structure: Protein 3D-structures are often depicted in a simplified cartoon format, which shows the location of the backbone of the amino acid chain and the presence of characteristic secondary structures like alpha-helices and beta-sheets. The side chains of the amino acids are not visible.

Introduction

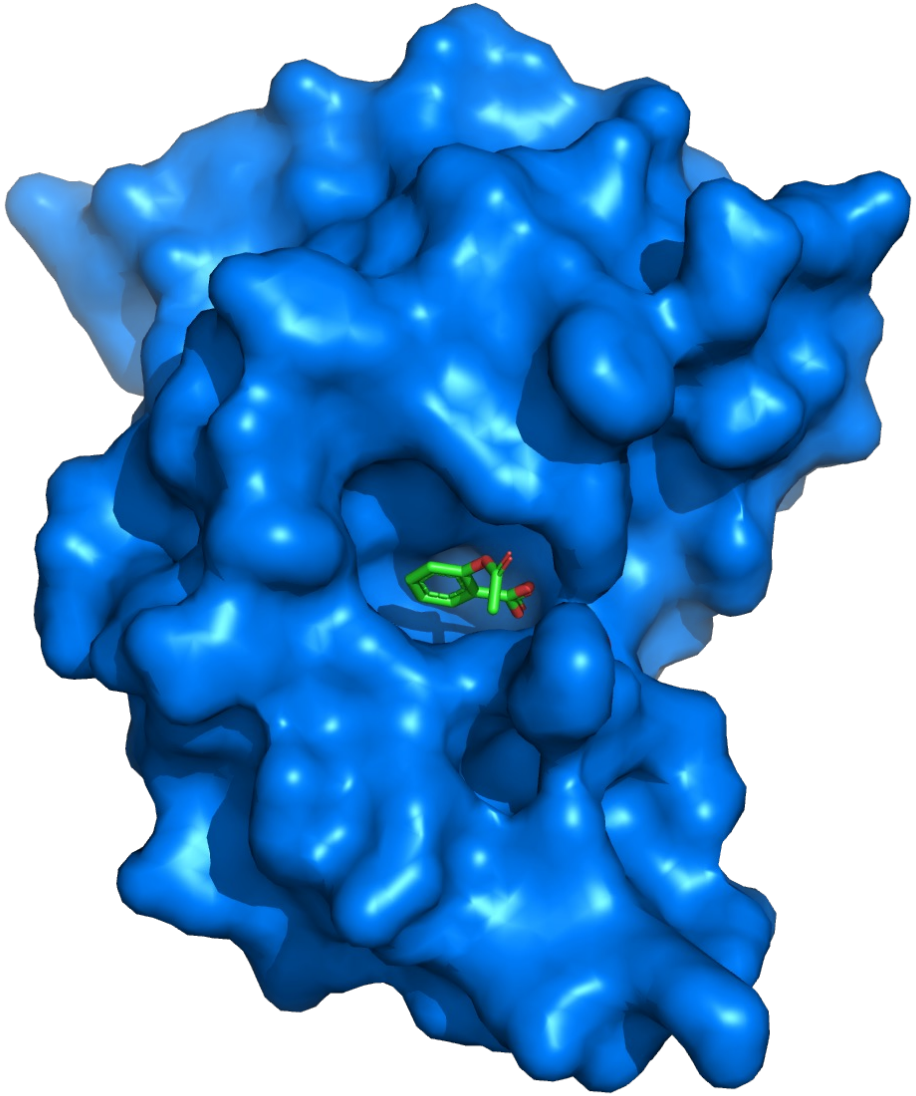
→ Proteins and Small Molecules
→ Engineering of Protein and Small Molecules

Engineering of Small Molecules

Most drugs are small molecules

- Can bind to proteins and affect a biological process
- The chemical structure and composition of the small molecule defines its function and binding preferences
- Small molecule drugs are engineered to interact with a specific target to modulate disease pathways
  - Inhibit enzymes
  - Block receptors

→ **Engineering of Small Molecules:**
Altering the size and composition of the organic compound to get desired properties (e.g. different carbon backbone and change functional groups)



Aspirin (in green and red) inhibits the activity of the enzyme cyclooxygenase (COX) which leads to the formation of prostaglandins (PGs) that cause inflammation, swelling, pain and fever (PDB 1OXR)

Engineering of Proteins

Altering the amino acid sequence of proteins to achieve specific functions or properties

Biocatalysis
Example: Change amino acid sequence of an enzyme to accept a new substrate

Therapeutics
Example: Change amino acid sequence of an antibody to bind to SARS-CoV-2 spike protein

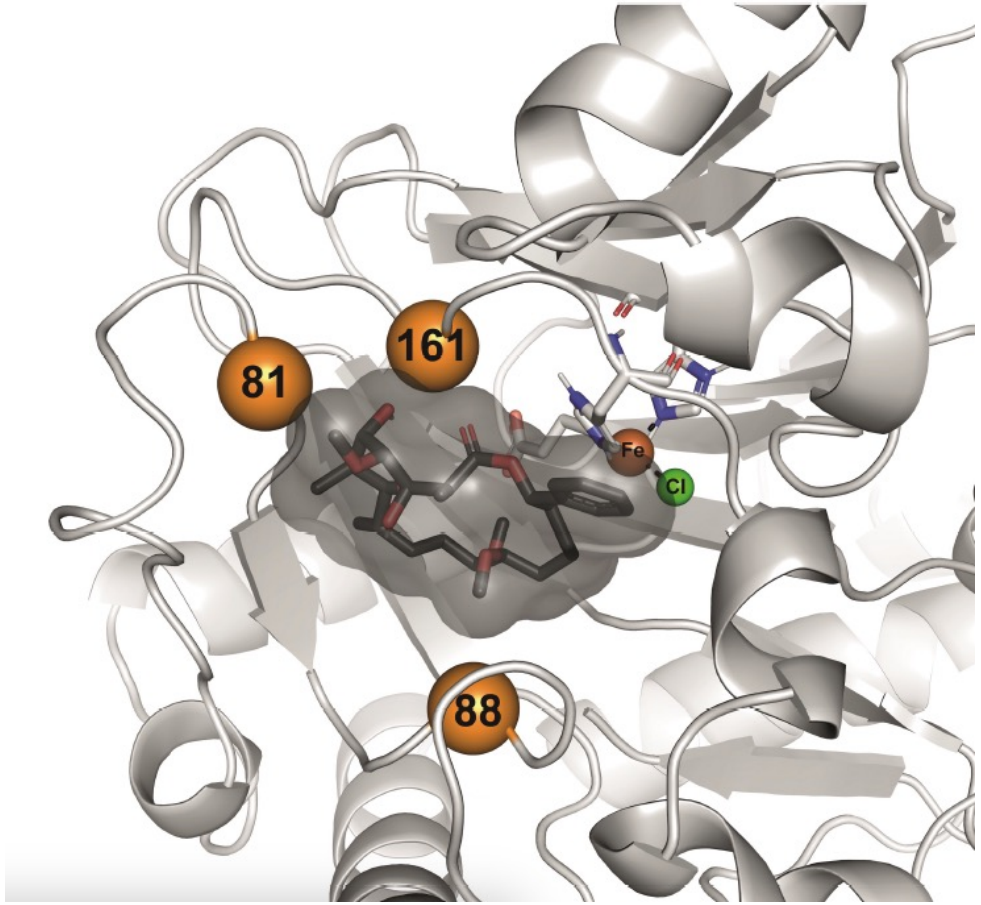


Image Credits: Büchler, J. *et al.* Algorithm-aided engineering of aliphatic halogenase WelO5* for the asymmetric late-stage functionalization of soraphens. *Nat Commun* **13**, 371 (2022).

Strategies for Engineering of Proteins

(Semi-)Rational Design:

- Use detailed knowledge of the structure of a protein to make desired changes
- Generate combinatorial libraries of a protein with varying amino acids at a specific location

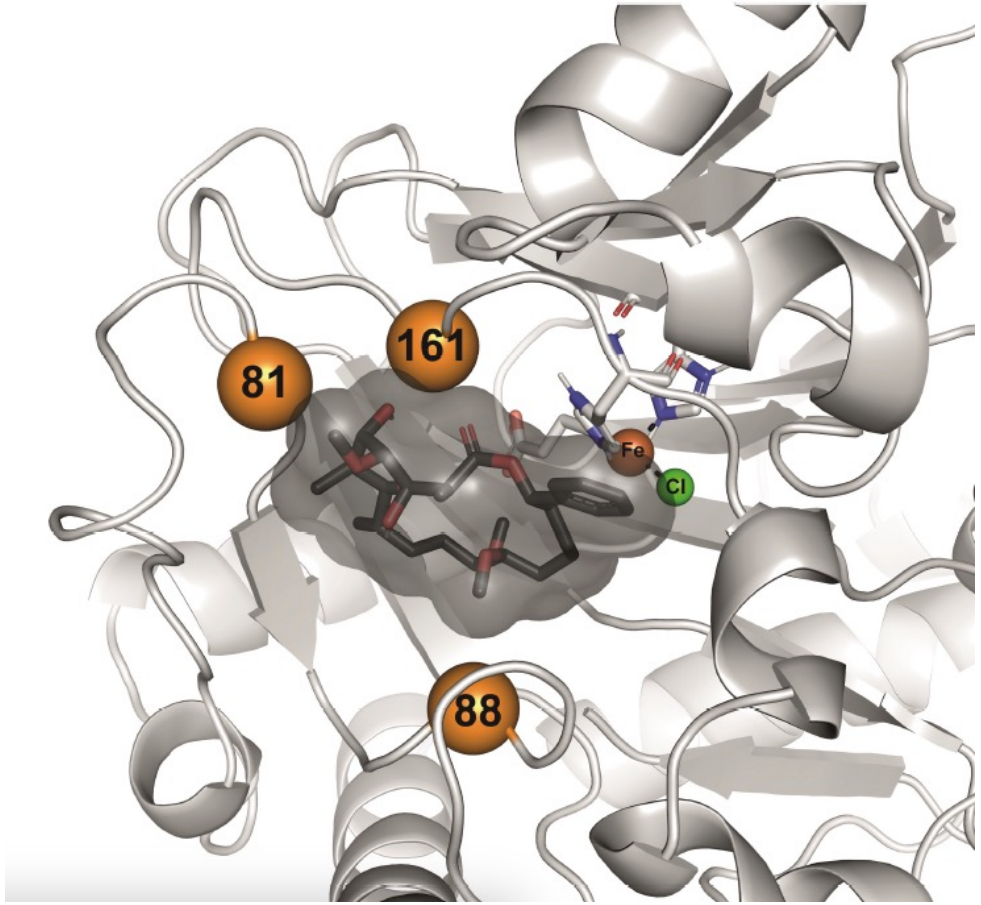


Image Credits: Büchler, J. *et al.* Algorithm-aided engineering of aliphatic halogenase WelO5* for the asymmetric late-stage functionalization of soraphens. *Nat Commun* **13**, 371 (2022).

Strategies for Engineering of Proteins

(Semi-)Rational Design:

- Use detailed knowledge of the structure of a protein to make desired changes
- Generate combinatorial libraries of a protein with varying amino acids at a specific location

Directed Evolution:

- Cycles of introducing random mutations and selecting most performant variant from the pool of mutants

Very time consuming and expensive and are only suited for optimizing existing proteins

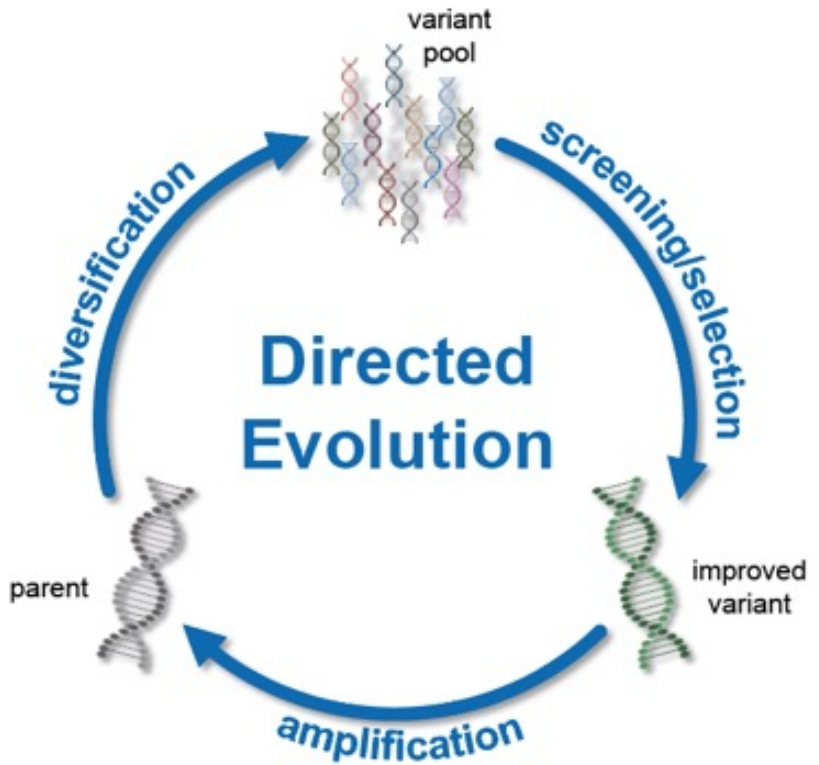


Image Credits: Bioprocess Laboratory, ETH Zürich (https://bsse.ethz.ch/bpl/research/directed-evolution.html)

Combinatorial Search Space

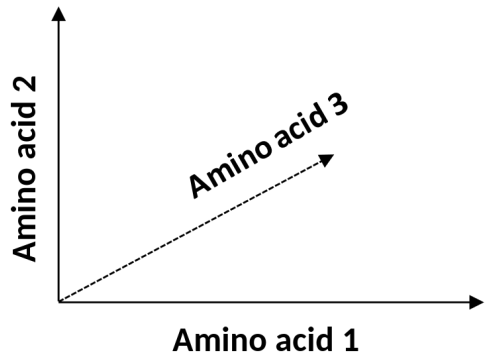
The search space for optimizing amino acid sequences in protein engineering is extremely large

Considering that there are 20 naturally occurring amino acids, each of which can occupy any position within the protein sequence, a protein of just 100 amino acids has 20^{100} possible sequences

All dipeptides in 2D space = 400

Amino acid 2 \ Amino acid 1	Ala	Arg	Asn	Asp	Cys	Glu	Gln	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	AA	RA	NA	DA	CA	EA	QA	GA	HA	IA	LA	KA	MA	FA	PA	SA	TA	WA	YA	VA
Arg	AR	RR	NR	DR	CR	ER	QR	GR	HR	IR	LR	KR	MR	FR	PR	SR	TR	WR	YR	VR
Asn	AN	RN	NN	DN	CN	EN	QN	GN	HN	IN	LN	KN	MN	FN	PN	SN	TN	WN	YN	VN
Asp	AD	RD	ND	DD	CD	ED	QD	GD	HD	ID	LD	KD	MD	FD	PD	SD	TD	WD	YD	VD
Cys	AC	RC	NC	DC	CC	EC	QC	GC	HC	IC	LC	KC	MC	FC	PC	SC	TC	WC	YC	VC
Glu	AE	RE	NE	DE	CE	EE	QE	GE	HE	IE	LE	KE	ME	FE	PE	SE	TE	WE	YE	VE
Gln	AQ	RQ	NQ	DQ	CQ	EQ	QQ	GQ	HQ	IQ	LQ	KQ	MQ	FQ	PQ	SQ	TQ	WQ	YQ	VQ
Gly	AG	RG	NG	DG	CG	EG	QG	GG	HG	IG	LG	KG	MG	FG	PG	SG	TG	WG	YG	VG
His	AH	RH	NH	DH	CH	EH	QH	GH	HH	IH	LH	KH	MH	FH	PH	SH	TH	WH	YH	VH
Ile	AI	RI	NI	DI	CI	EI	QI	GI	HI	II	LI	KI	MI	FI	PI	SI	TI	WI	YI	VI
Leu	AL	RL	NL	DL	CL	EL	QL	GL	HL	IL	LL	KL	ML	FL	PL	SL	TL	WL	YL	VL
Lys	AK	RK	NK	DK	CK	EK	QK	GK	HK	IK	LK	KK	MK	FK	PK	SK	TK	WK	YK	VK
Met	AM	RM	NM	DM	CM	EM	QM	GM	HM	IM	LM	KM	MM	FM	PM	SM	TM	WM	YM	VM
Phe	AF	RF	NF	DF	CF	EF	QF	GF	HF	IF	LF	KF	MF	FF	PF	SF	TF	WF	YF	VF
Pro	AP	RP	NP	DP	CP	EP	QP	GP	HP	IP	LP	KP	MP	FP	PP	SP	TP	WP	YP	VP
Ser	AS	RS	NS	DS	CS	ES	QS	GS	HS	IS	LS	KS	MS	FS	PS	SS	TS	WS	YS	VS
Thr	AT	RT	NT	DT	CT	ET	QT	GT	HT	IT	LT	KT	MT	FT	PT	ST	TT	WT	YT	VT
Trp	AW	RW	NW	DW	CW	EW	QW	GW	HW	IW	LW	KW	MW	FW	PW	SW	TW	WW	YW	VW
Tyr	AY	RY	NY	DY	CY	EY	QY	GY	HY	IY	LY	KY	MY	FY	PY	SY	TY	WY	YY	VY
Val	AV	RV	NV	DV	CV	EV	QV	GV	HV	IV	LV	KV	MV	FV	PV	SV	TV	WV	YV	VV

All tripeptides in 3D space = 8000



All 10 a.a proteins in 10D space ≈ 10^{13}

All 50 a.a proteins in 50D space ≈ 10^{65}

All 100 a.a proteins in 100D space ≈ 10^{130}

All 300 a.a proteins in 300D space ≈ 10^{390}

Image Credits: https://en.wikipedia.org/wiki/Sequence_space_%28evolution%29

Text-based AI models

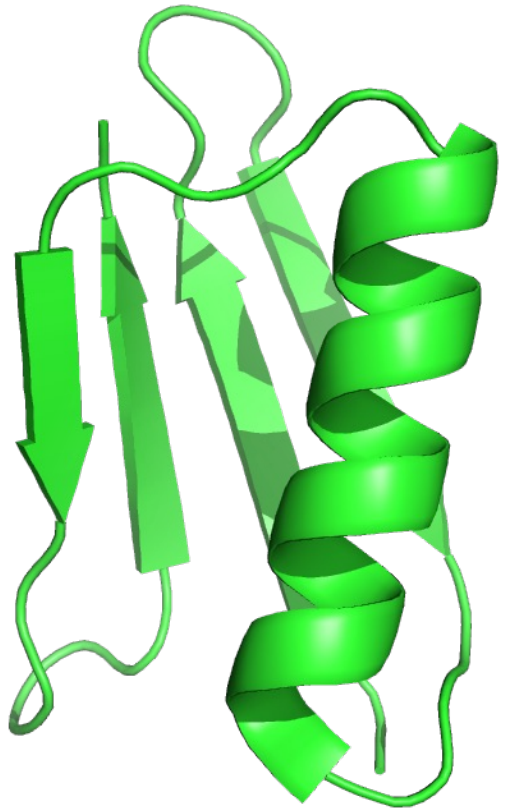
→ Text representation of proteins and small molecules
→ AI model examples

Proteins: text-based representation

Amino Acid Sequences: A linear sequence of letters representing the order of amino acids in the protein

For machine learning, amino acid sequences are encoded into numerical representations by

- One-Hot-Encoding
- Tokenization - Each amino acid is treated as a separate token, like words in text processing.



MTYKLILNGKTLKGETTTEAVDAATAEKVF
KQYANDNGVDGEWTYDAATKTFTVTE

Small molecules: text-based representation

SMILES (Simplified Molecular Input Line Entry System)

- A linear notation representing the molecular structure using ASCII characters
- Atoms are represented by their chemical symbols (C, N, O, P, S, F, Cl, Br, I, H)
- Single bonds are omitted, double bonds are denoted by “=“
- Branches are described using parentheses
- Rings are noted by using numbers to indicate the connection points

For machine learning, SMILES codes are tokenized and mapped to a fixed vocabulary.

Image Credits: https://en.wikipedia.org/wiki/Simplified_molecular-input_line-entry_system

Natural Language Processing in Chemistry and Biology

Analogy between the building blocks of language and those of proteins & small molecules

Proteins: Like natural language, protein sequences have their own kind of grammar and contain long-range dependencies

- Amino acids ≈ words
- Sequences of amino acids ≈ sentences
- Proteins ≈ text paragraph

Small molecules: SMILES codes are also linear sequences that consist of a fixed vocabulary of symbols with patterns and rules

→ **Both protein sequences and SMILES codes are compatible with language model architectures designed to handle sequences**

→ **Methods of Natural Language Processing (NLP) are applied to proteins & small molecules**

Molecular Property Prediction

SMILES-to-Property-Transformer SPT

A transformer model trained to predict temperature-dependent limiting activity coefficients for molecules (a measure of solubility of a molecule and how it behaves in a solvent)

- **Input:** SMILES solute + SMILES solvent
- **Output:** Predicted activity coefficient of the solute

- Input SMILES are tokenized like in NLP
- Temperature information is added to the input matrix
- Transformer block with regression head

Supervised training on experimental data. Performance superior to traditional physical predictive models

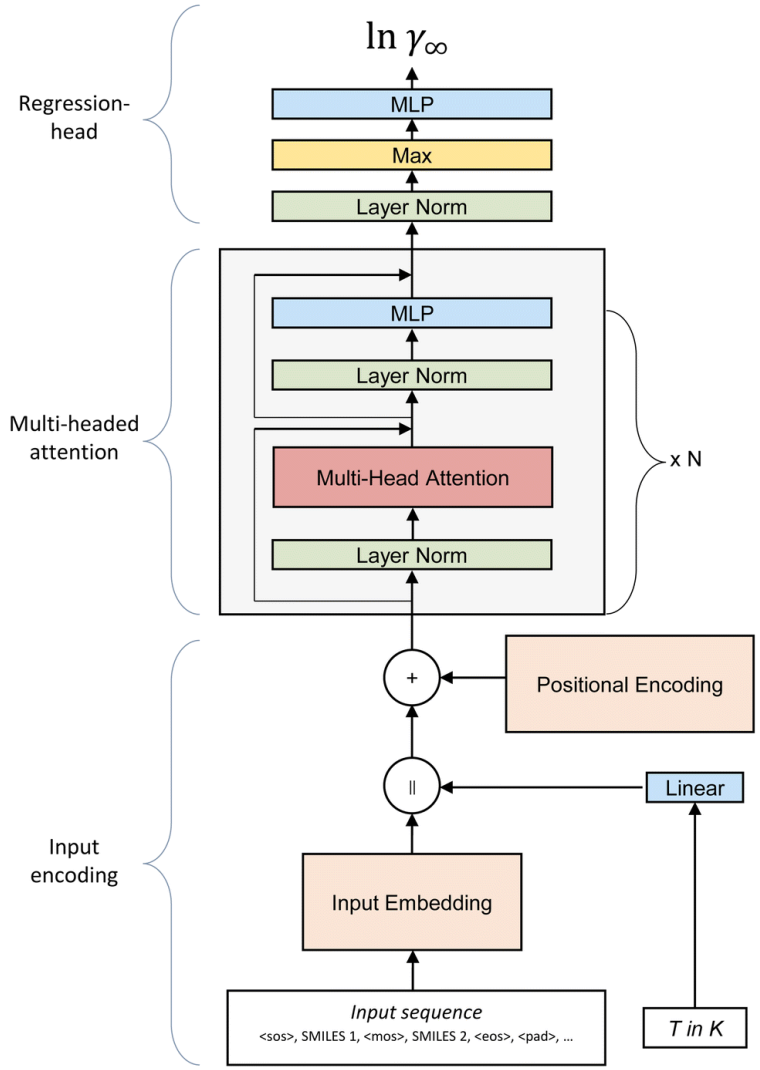


Figure: Input sequences of SMILES are tokenized into an embedding matrix and supplemented with temperature information. The resulting embeddings are passed through a transformer block with a subsequent regression head to reduce the transformer output to a single value, representing the property of interest, i.e. the limiting activity coefficient.

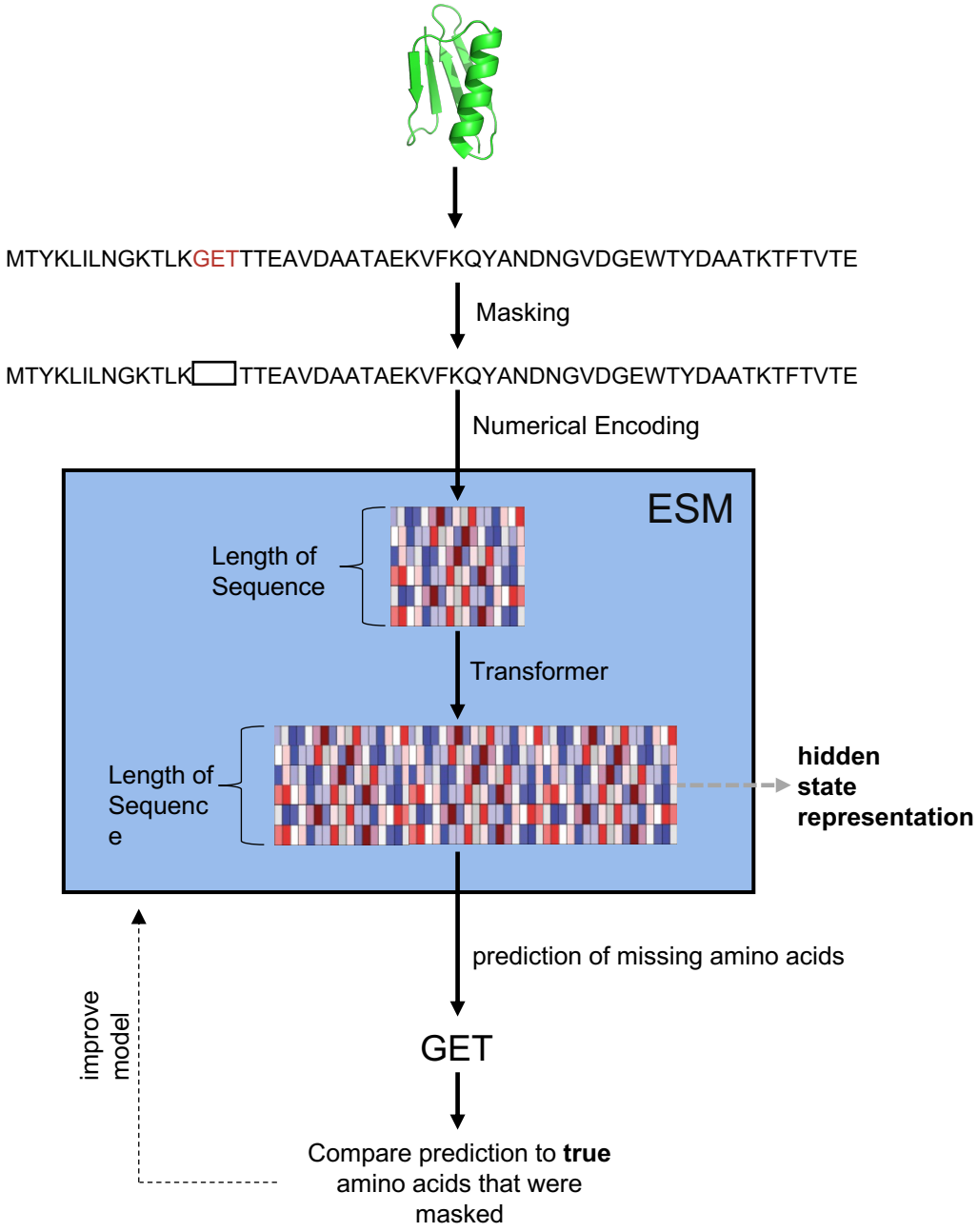
Winter, B., Winter, C., Schilling, J. & Bardow, A. A smile is all you need: predicting limiting activity coefficients from SMILES with natural language processing. *Digit. Discov.* **1**, 859–869 (2022).

Evolutionary Scale Modelling (ESM)

Self-supervised transformer language model

- Self-supervised learning with unlabelled datasets, far larger amounts of data available
- Trained with the masked language modelling objective on 250 million protein sequences
- Each input sequence is corrupted by replacing a fraction of the amino acids with a mask
- ESM transformer generates prediction for the masked amino acids from the sequence context

→ **Knowledge of intrinsic biological properties of the proteins emerges without supervision, encoded in the hidden representations of the language model**



Rives, A. *et al.* Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118**, e2016239118 (2021).

Evolutionary Scale Modelling (ESM)

Self-supervised transformer language model

- Self-supervised learning with unlabelled datasets, far larger amounts of data available
- Trained with the masked language modelling objective on 250 million protein sequences
- Each input sequence is corrupted by replacing a fraction of the amino acids with a mask
- ESM transformer generates prediction for the masked amino acids from the sequence context

→ **Knowledge of intrinsic biological properties of the proteins emerges without supervision, encoded in the hidden representations of the language model**

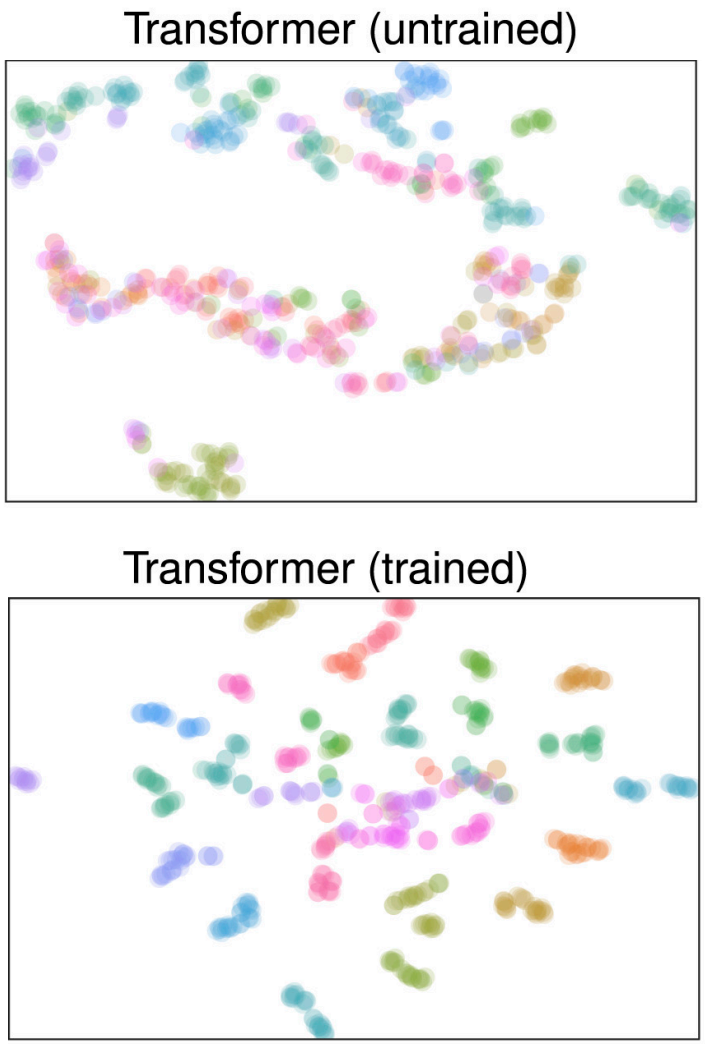


Figure: Projection of hidden representations of trained and untrained transformer model to 2D with t-distributed stochastic neighbor embedding (t-SNE). Each point represents a gene and is colored by the orthologous group it belongs to. Orthologous groups of genes are densely clustered in the trained representation space. By contrast, the untrained representation space does not reflect strong organization by evolutionary relationships.

Rives, A. *et al.* Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118**, e2016239118 (2021).

Downstream Applications & Transfer Learning

Transfer Learning – Applying a model trained on one task to a related but slightly different task can improve performance, especially when annotated data for the second task are scarce.

Fine-tuning protein language models on specific downstream tasks
Using supervised learning with (potentially small) labelled datasets to fine-tune the language model for particular applications

- Protein function prediction
- Prediction of mutational effects
- Prediction of protein stability
- Prediction of protein 3D structure

→ **The potential of self-supervised transformer representations is currently starting to be exploited**

Protein Structure Prediction

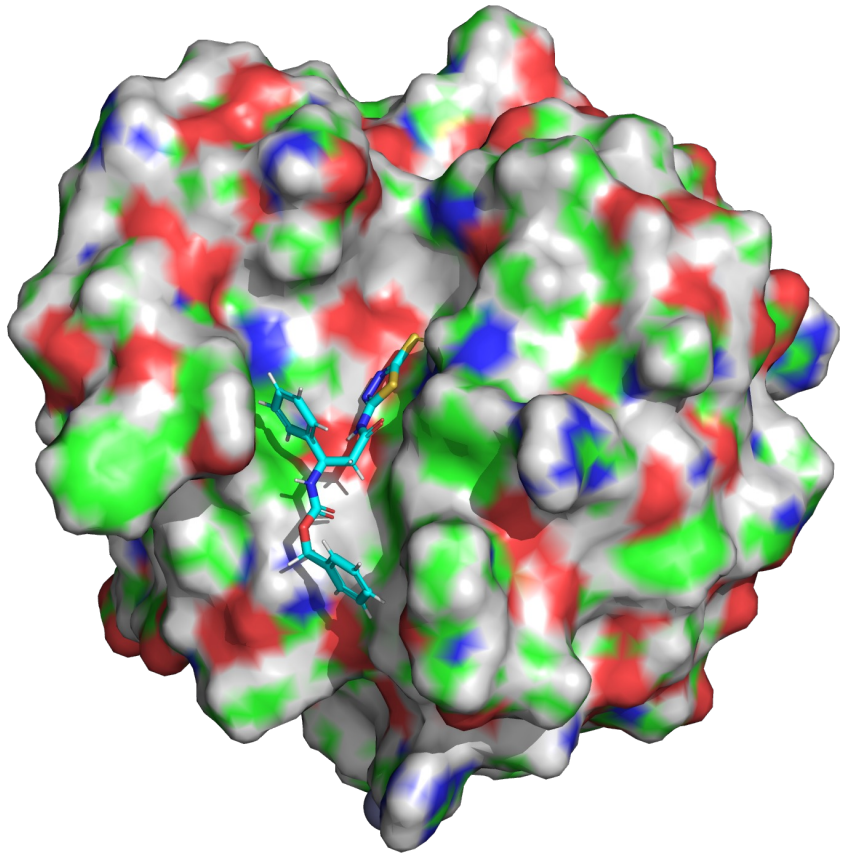
→ AlphaFold2
→ Evolutionary Scale Modelling (ESM)
→ Performance Comparison

Protein Structure

→ 3D coordinates of atoms of the protein in folded state

Why Protein Structure is important:

- Fundamental understanding:
  The structure of a protein defines its function. Knowing the structure of a protein helps to find its biological role
- Drug discovery:
  Knowing protein structure allows for the identification of active sites and interaction points for potential drug molecules



How to get a protein structure:
For many years, researchers have relied on very time-consuming experimental methods, such as X-ray Crystallography and Nuclear Magnetic Resonance Spectroscopy (NMR) to find the structure of a protein with a given sequence.

→ **Recent advancements use Deep Learning to predict protein structures**

The protein folding task

Predicting the 3D fold of a protein from its amino acid sequence is a complex and challenging task

- Amino acid residues can adopt many conformations due to the rotation around its backbone bonds and side chains
- Astronomical number of possible structures even for small protein
- Folding is driven by various interactions between amino acids which are difficult to model (e.g. hydrogen bonds, hydrophobic interactions, van der Waals forces etc.)
- Long-range dependencies: Folding is influenced by interactions between amino acids that are far apart in the sequence but come close in the 3D structure
- Training data is relatively scarce

→ **Critical Assessment of Protein Structure Prediction (CASP)**

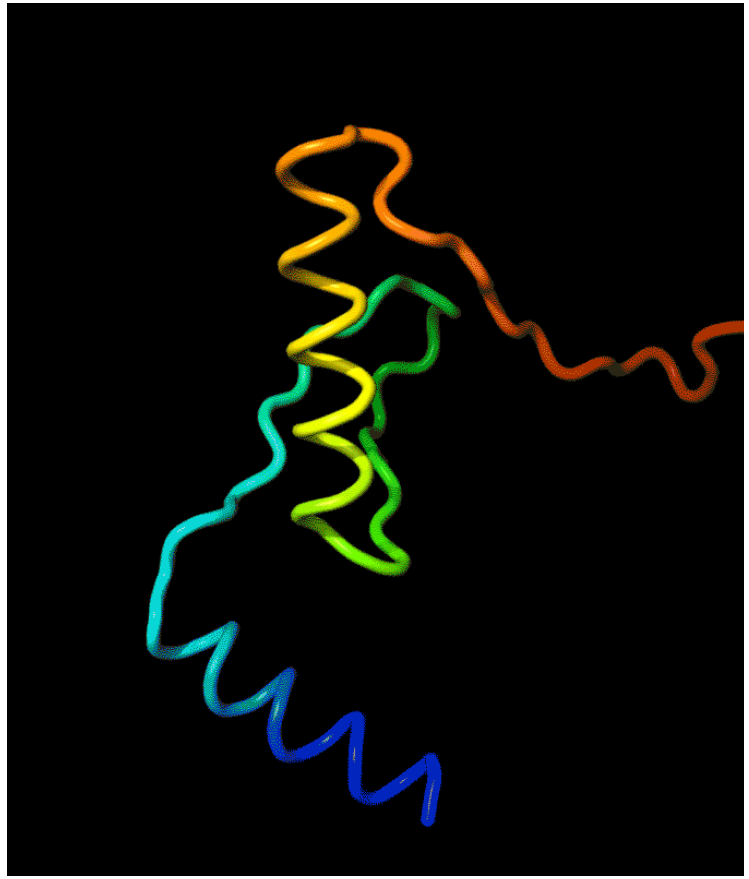


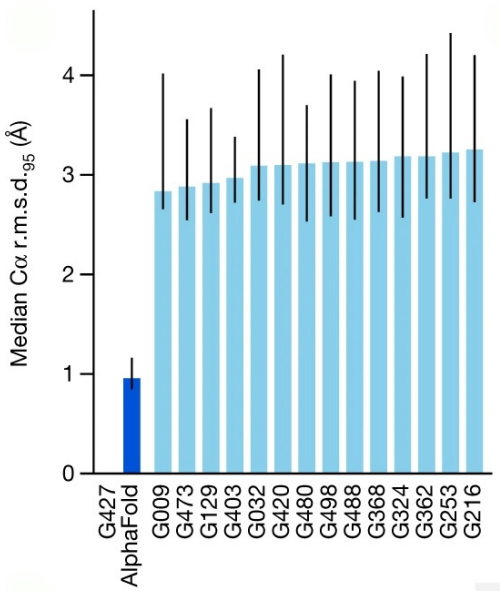
Image Credits: Deepmind

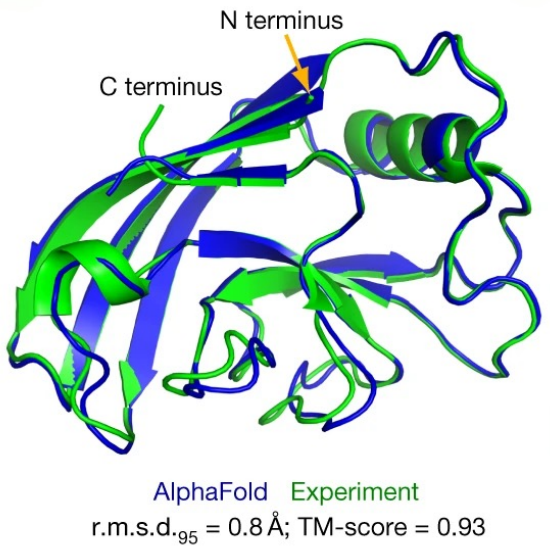
AlphaFold2 for protein structure prediction

First model demonstrating near-experimental accuracy in the task of protein structure prediction

- Winner of the 14th Critical Assessment of Protein Structure Prediction (CASP) challenge
- Predictions showed root-mean-square deviation of Cα-atoms of 0.96Å compared to experimental structures
- To compare: width of carbon atom = 1.4Å
- Combines multiple sequence alignments (MSAs) to databases of existing structures with an equivariant transformer architecture to predict 3D atomic coordinates

→ **Breakthrough in the field of structural bioinformatics**





Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 model architecture

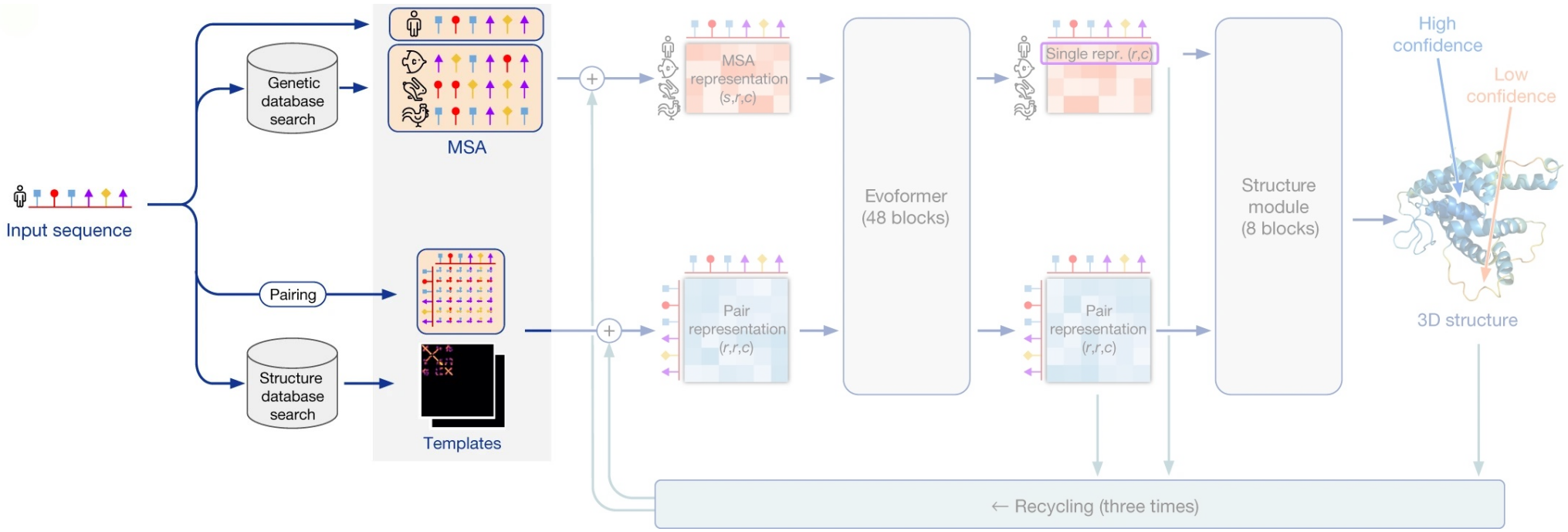


Figure: AlphaFold model architecture. Arrows show the information flow among the various components. Array shapes are shown in parentheses with s, number of sequences, r, number of residues and c, number of channels

Input to the model

- Input amino acid sequence
- Sequences from evolutionarily related proteins in the form of a multiple sequence alignment
- 3D atom coordinates of homologous structures (templates)

Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 model architecture

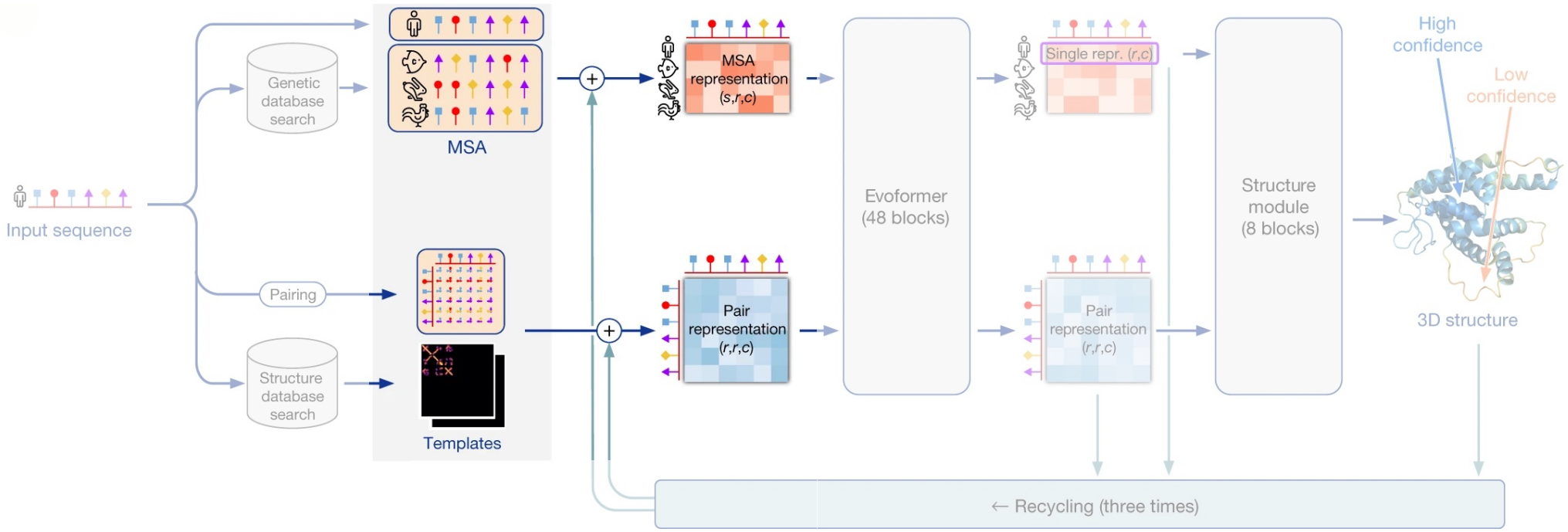


Figure: AlphaFold model architecture. Arrows show the information flow among the various components. Array shapes are shown in parentheses with s, number of sequences, r, number of residues and c, number of channels

Initial representations

- **MSA representation:** The input sequence and the MSA are translated into a MSA representation containing sequence conservation information and sequence entropy
- **Pair representation:** Based on analysis of known protein structures (templates), an initial prediction of pairwise amino acid distances is generated

Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 model architecture

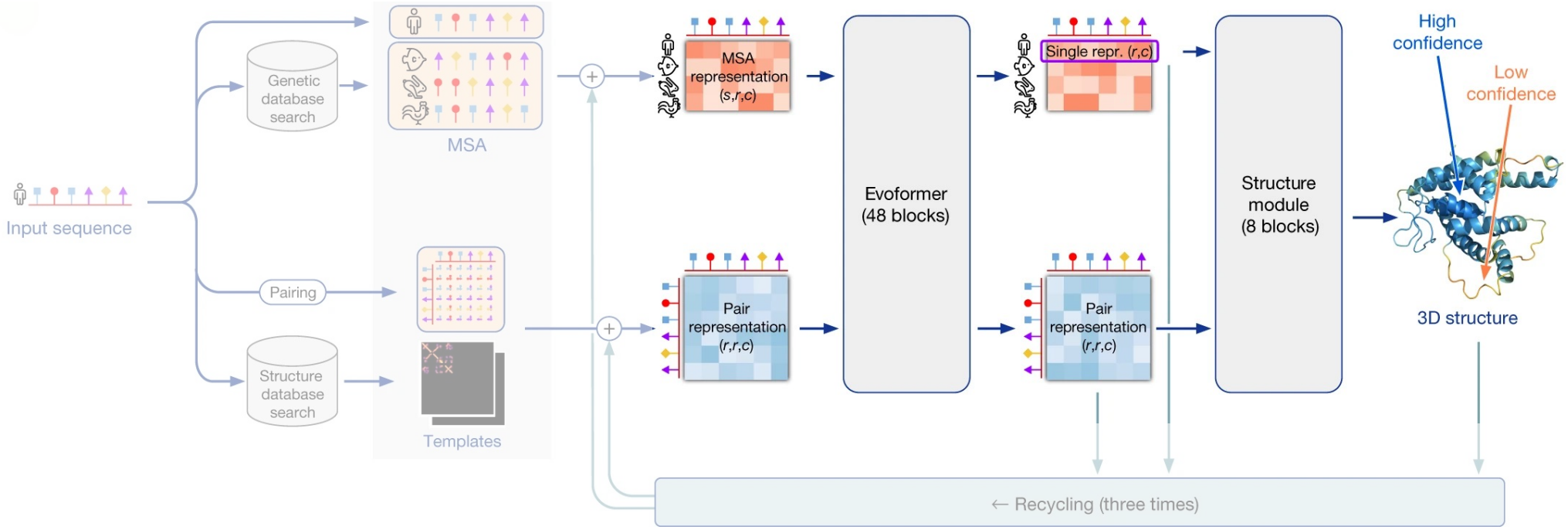


Figure: AlphaFold model architecture. Arrows show the information flow among the various components. Array shapes are shown in parentheses with s, number of sequences, r, number of residues and c, number of channels

Evoformer:

- Core of the architecture
- Updates the two representations through attention mechanisms and convolution-like operations

Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 model architecture

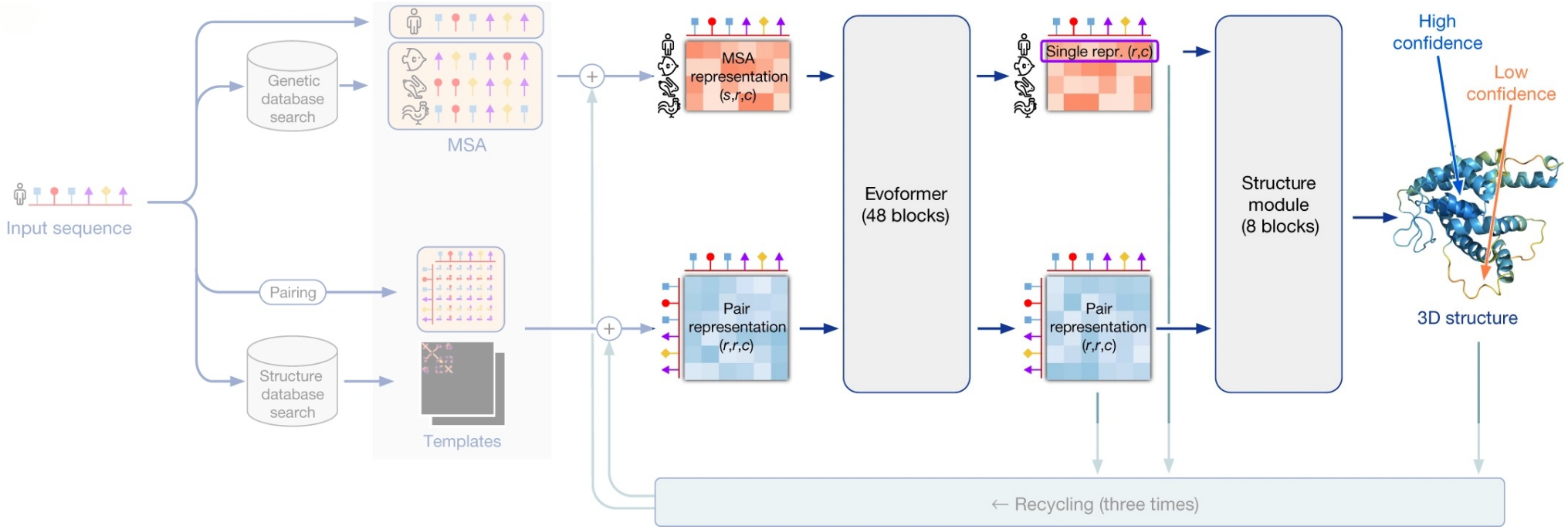


Figure: AlphaFold model architecture. Arrows show the information flow among the various components. Array shapes are shown in parentheses with s, number of sequences, r, number of residues and c, number of channels

Structure Module:

- Converts the processed representations from the Evoformer into spatial coordinates.
- Predicts the distances between residue pairs and the angles of bonds within the protein.
- Constructs a 3D model of the protein.

Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 model architecture

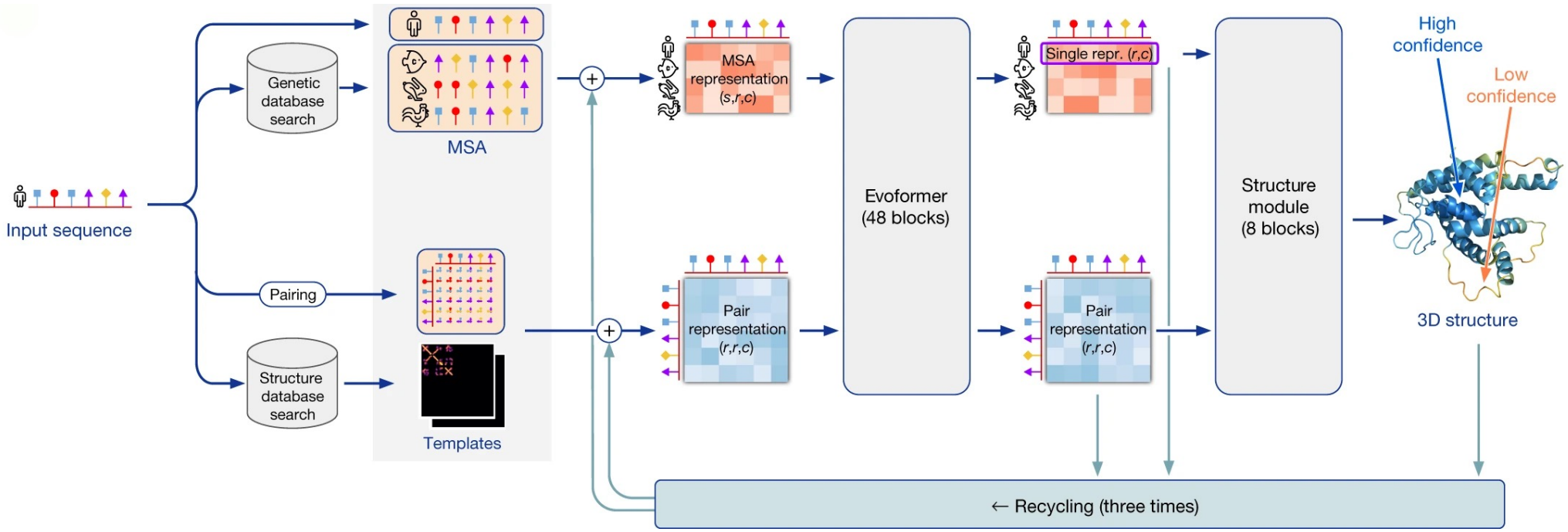


Figure: AlphaFold model architecture. Arrows show the information flow among the various components. Array shapes are shown in parentheses with s, number of sequences, r, number of residues and c, number of channels

Recycling:

- Iterative refinement process where the output of the structure module is fed back multiple times
- Helps refine the predictions by allowing the network to reconsider its earlier outputs
- Benefits of additional contextual information gleaned from subsequent layers.

1. Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

AlphaFold2 training

Structure objectives: Deviation from true structure is penalized

BERT objective: Random masking is applied on the input MSAs and the network is required to reconstruct the masked regions from the output MSA representation

Training with self-distillation:

1. **Initial supervised training:** Training on a dataset of known protein structures, using conventional supervised learning techniques on all structures in the Protein Data Bank (PDB).
2. **Iterative refinement with self-distillation:** Refine the model's predictions using its own outputs
   - Use the model trained in supervised learning to predict the structure of 350'000 additional sequences
   - Retrain the model from scratch using mix of PDB data and predicted data

→ **Makes effective use of unlabelled sequence and improves the accuracy of the model**

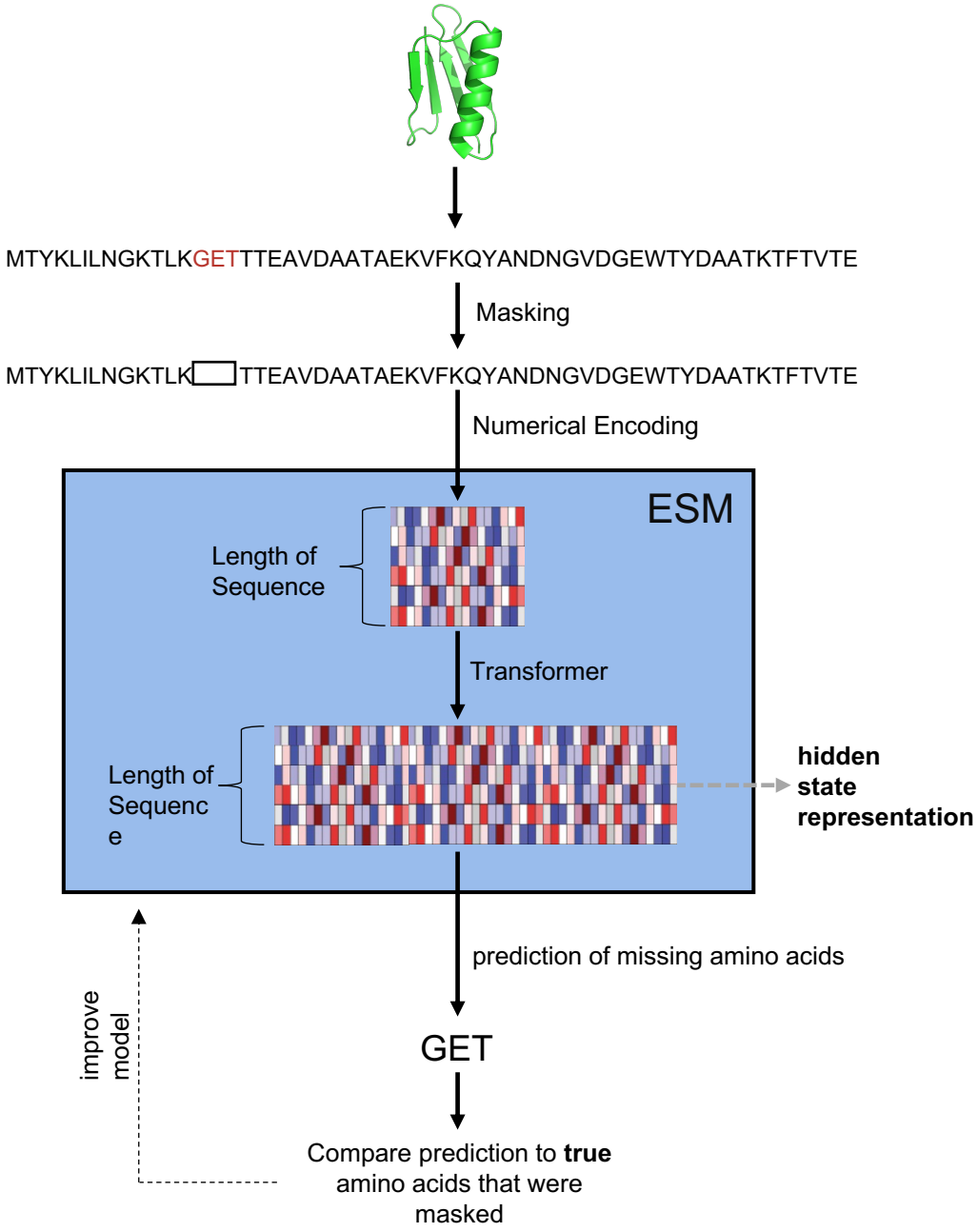
Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

Evolutionary Scale Modelling (ESM)

Self-supervised transformer language model

- Self-supervised learning with unlabelled datasets, far larger amounts of data available
- Trained with the masked language modelling objective on 250 million protein sequences
- Each input sequence is corrupted by replacing a fraction of the amino acids with a mask
- ESM transformer generates prediction for the masked amino acids from the sequence context

→ **Knowledge of intrinsic biological properties of the proteins emerges without supervision, encoded in the hidden representations of the language model**



Rives, A. *et al.* Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118**, e2016239118 (2021).

Evolutionary Scale Modelling (ESM)

Unsupervised pretraining encodes secondary structure into representations

- The model cannot observe protein structure directly, it observes patterns in the sequences that are determined by the structure (a hidden variable)

Dataset of proteins with known structure, each amino acid is part of a helix, a strand or a coil

- Derive a transformer hidden representation for each amino acid of the protein
- Fit logistic regression classifier to predict secondary structure membership for each amino from its hidden representation

→ **High accuracy - self-supervised training generates structural knowledge**

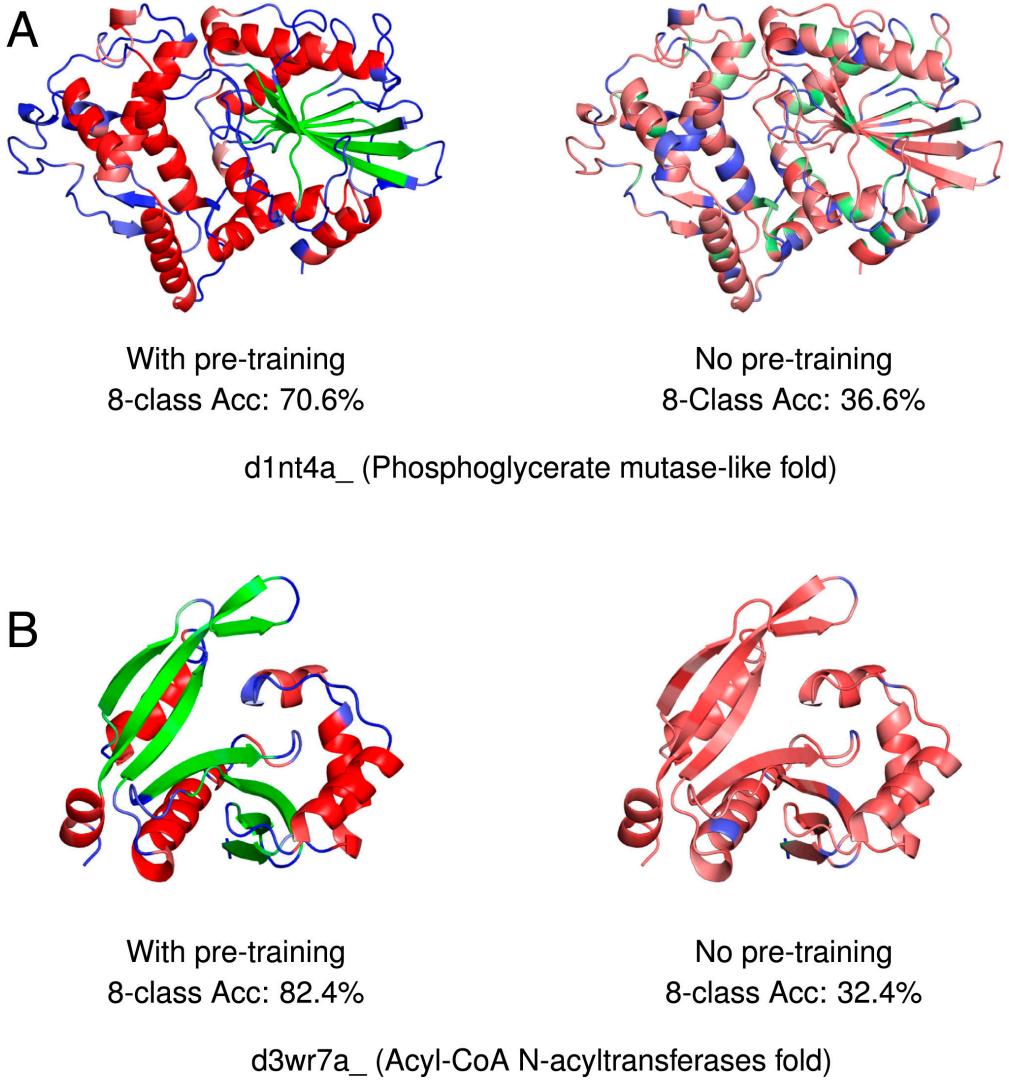


Figure: Unsupervised training encodes secondary structure into representations. Following pretraining, linear projections recover secondary structure (left column). Without pretraining, little information is recovered (right column). Colors indicate secondary structure class identified by the projection: **helix (red), strand (green), and coil (blue).**

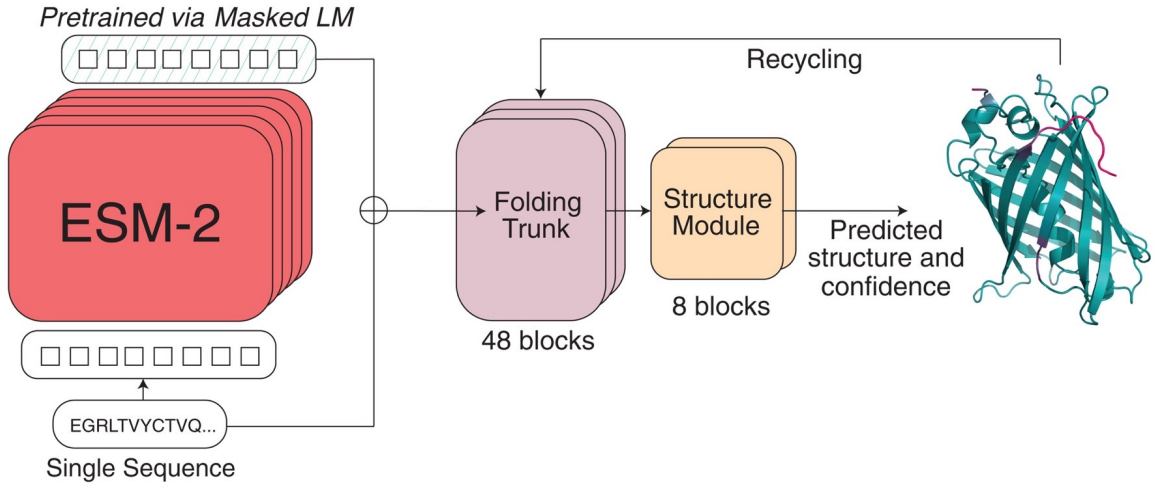
Rives, A. *et al.* Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118**, e2016239118 (2021).

Evolutionary Scale Modelling **2** (ESM**2**)

ESM language model is extended with a **folding head** to predict atomic coordinates

- Atomic coordinates are extracted from the hidden representations of ESM with an equivariant transformer
- Trained end-to-end on 325K experimentally determined structures from the Protein Data Bank (PDB)

→ **Structure prediction accuracy similar to AlphaFold2**



Lin, Z. [illegible] prediction of atomic level protein structure with a language model. (2022) doi:10 [illegible] 02.

AlphaFold and ESM – Prediction Accuracy

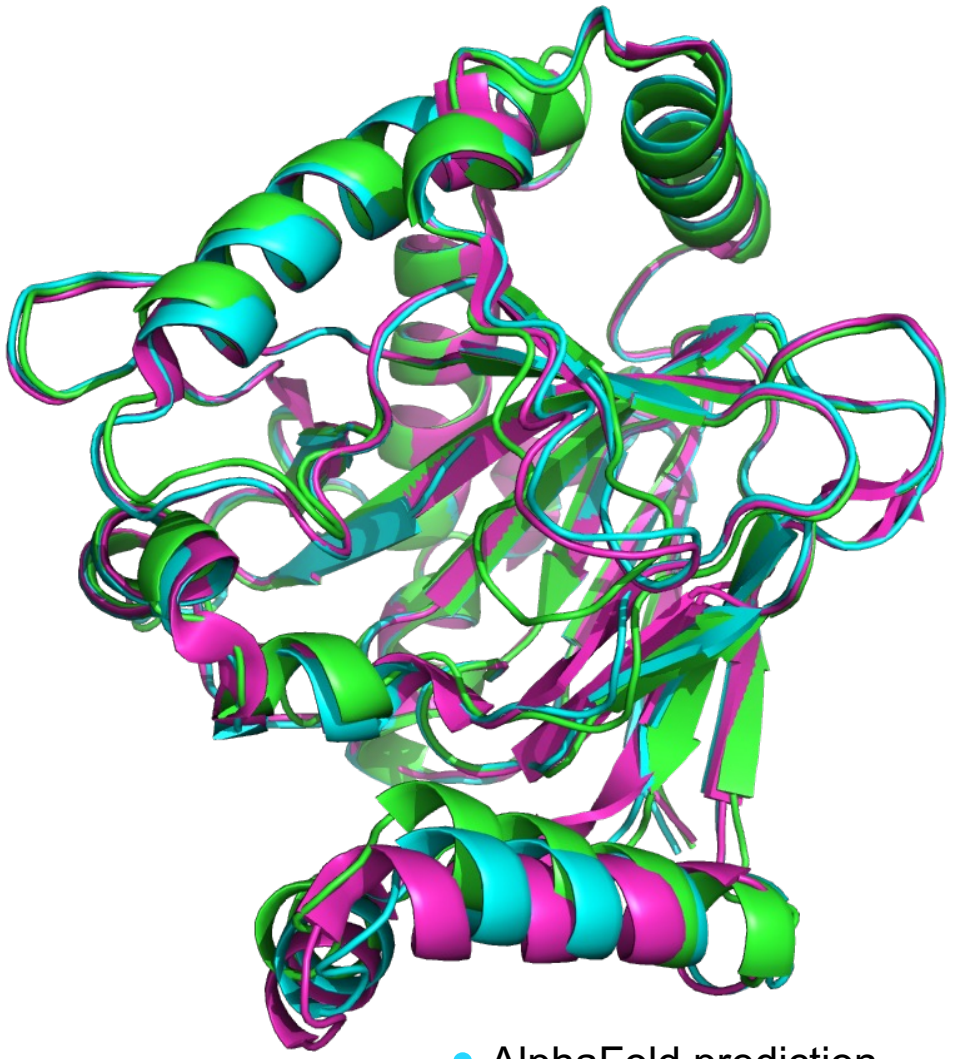
ESM language model is extended with a **folding head** to predict atomic coordinates

- Atomic coordinates are extracted from the hidden representations of ESM with an equivariant transformer
- Trained end-to-end on 325K experimentally determined structures from the Protein Data Bank (PDB)

→ **Structure prediction accuracy similar to AlphaFold2**

→ **Removes costly aspects AlphaFold (multiple sequence alignment), leading to 60x speed-up**

→ **Completely standalone, the only input needed for inference is the protein sequence**



Lin, Z. *et al.* Evolutionary-scale prediction of atomic level protein structure with a language model. (2022) doi:10.1101/2022.07.20.500902.

AlphaFold and ESM – Performance Comparison

Prediction of backbone atoms:
Both models perform well, with AlphaFold often leading in direct comparisons

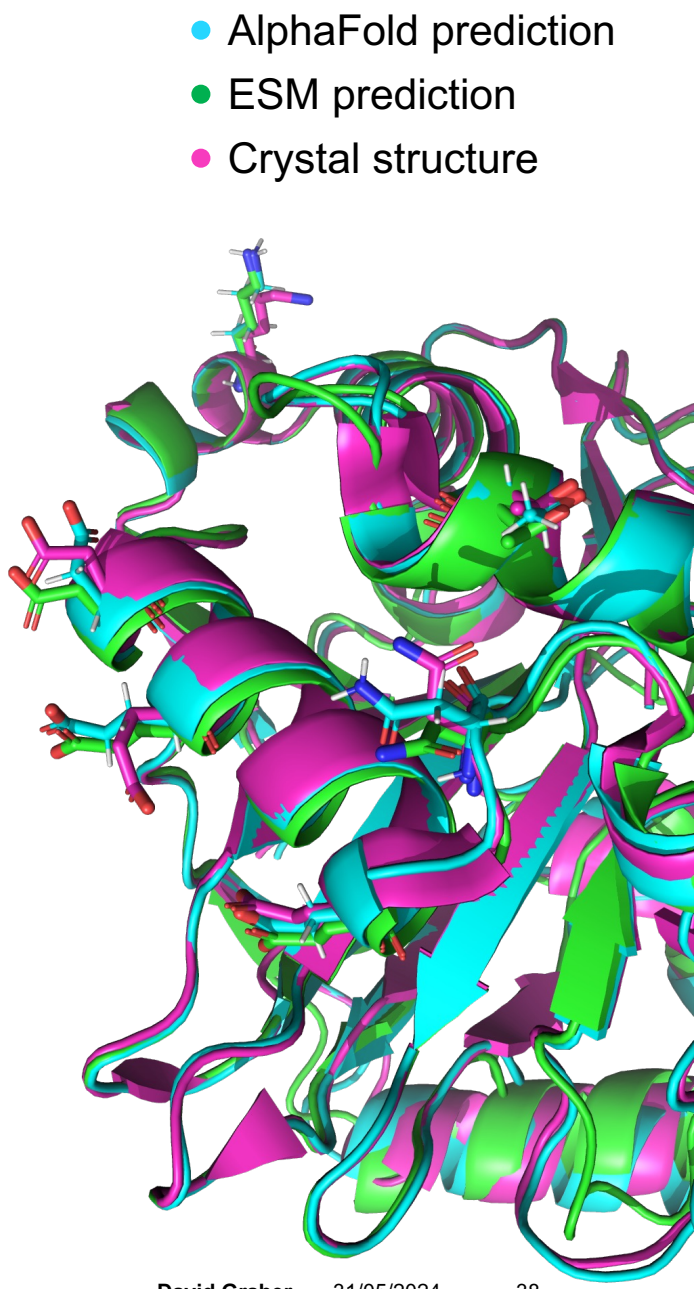
Prediction of side chain atoms:
The precision of both models drops slightly compared to backbone atoms, as side chains are more flexible and harder to predict. AlphaFold generally performs better in modelling side chains.

Which model to use:

→ AlphaFold predictions are often preferred when high accuracy in both backbone and side chain details is crucial.

→ ESMFold provides a highly valuable tool, especially when quick predictions are needed across a vast number of proteins

Protein Structures are now much easier and faster to obtain!



Structure-based AI models

→ 3D-Convolutional Neural Networks
→ Introduction to Graphs and Graph Neural Networks
→ Graph-based models

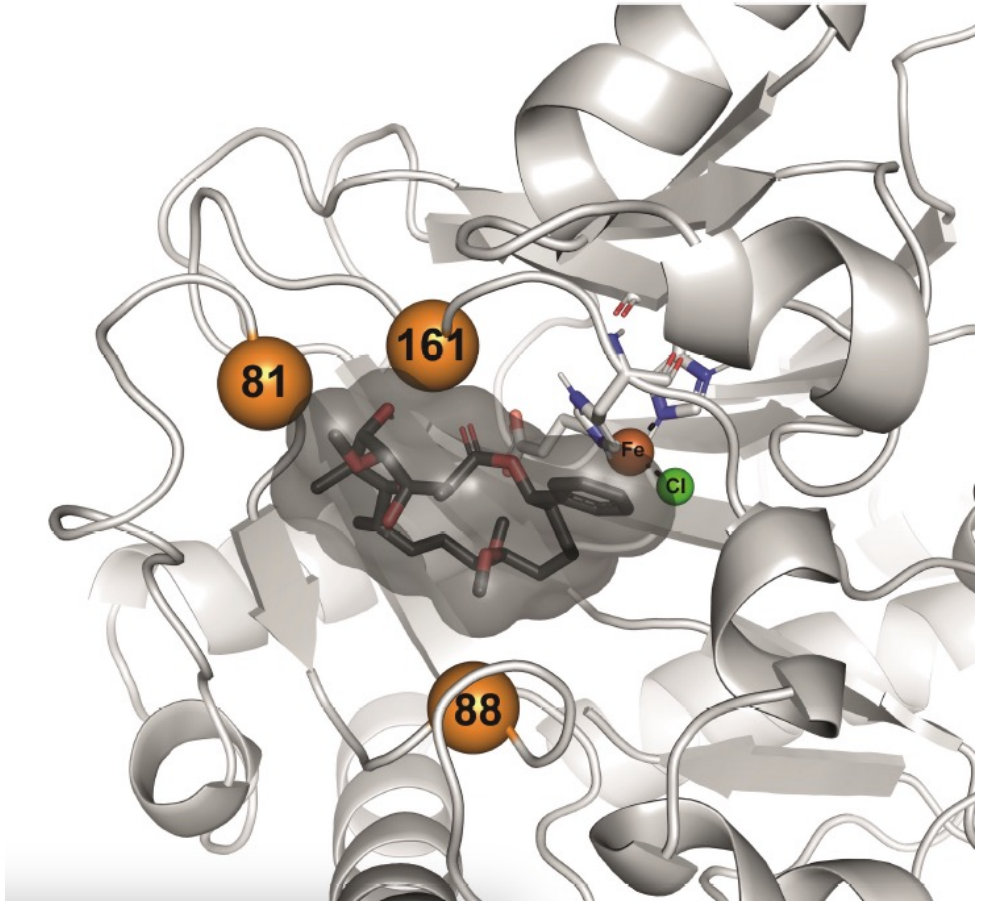
Using structural data as input for AI models

3D structure determines function: Proteins fold into complex shapes that allow them to interact specifically with other with other molecules

- A protein sequence alone provides no information about the actual spatial arrangement of atoms in the 3D space
- Models that incorporate this structural information **should** be superior at prediction tasks that require an understanding of the physical placement of residues (e.g. interaction prediction, enzyme activity prediction…)

Challenges in the use of structural data:

- Limited availability of structural data
- Sequence-based methods are generally simpler and more computationally efficient



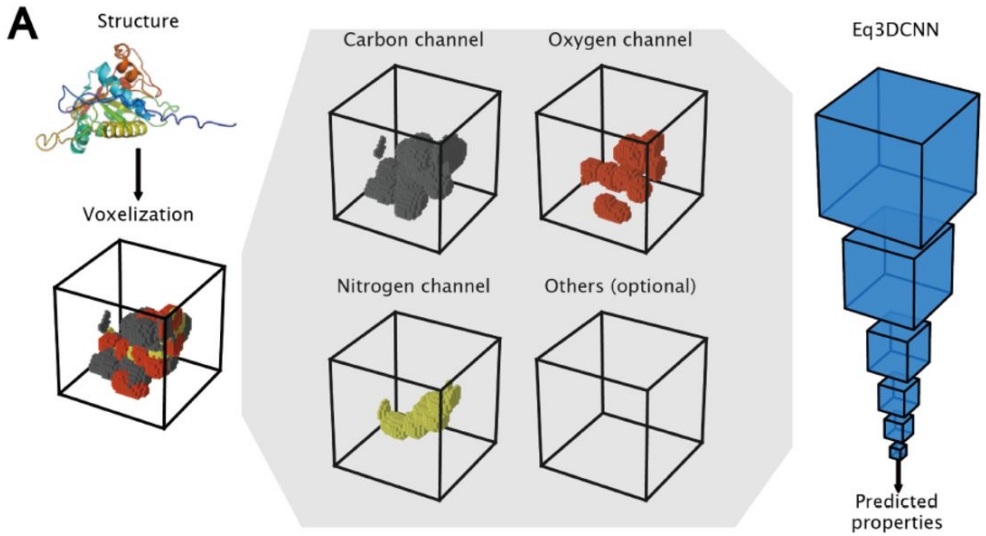
Structure defines function: In the 3-dimensional (3D) fold of this enzyme, different parts of the amino acid chain come together in 3D space to form an active site that accepts a specific substrate. The function of the enzyme is largely defined by the spatial arrangement of the involved residues.

Image credits: Büchler, J. *et al.* Algorithm-aided engineering of aliphatic halogenase WelO5* for the asymmetric late-stage functionalization of soraphens. *Nat Commun* **13**, 371 (2022).

3D-Convolutional Neural Networks (3D-CNNs)

3D-CNNs – general procedure

- To convert the atomic coordinates into a format suitable for 3D-CNNs, the space around the protein is divided into a grid of voxels
- Different types of atomic properties can be used to define multiple channels in the input data (e.g. atom types, charges…)
- Convolution and pooling layers to extract relevant features and reduce the spatial dimensions of the input volume
- Fully connected layer derives global prediction



➔ 3D-CNNs have been shown to perform well in tasks such as protein thermostability prediction

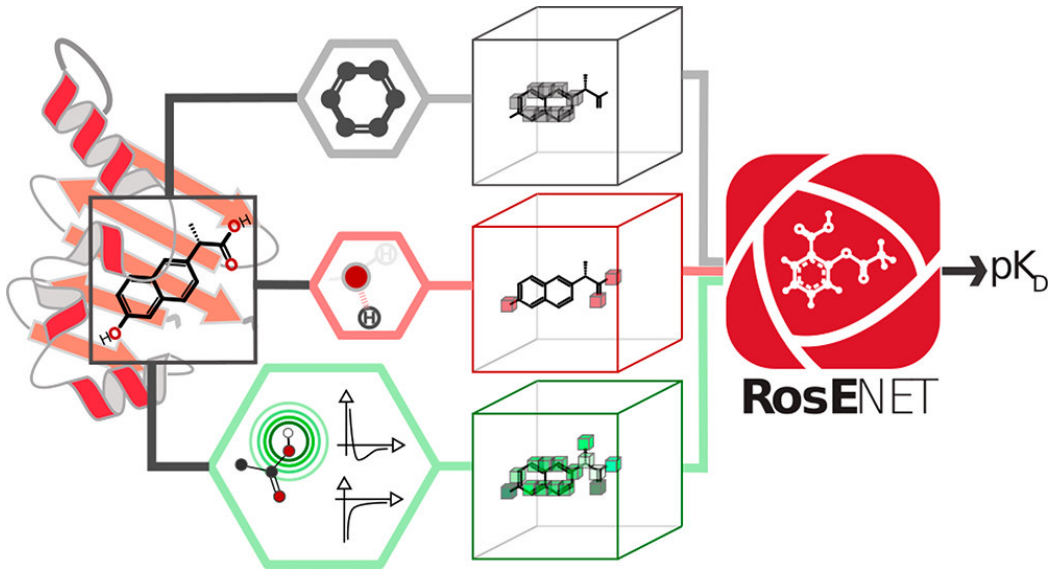
Chen, H. *et al.* Protein property prediction based on local environment by 3D equivariant convolutional neural networks. *bioRxiv* 2024.02.07.579261 (2024) doi:10.1101/2024.02.07.579261.

3D-Convolutional Neural Networks (3D-CNNs)

3D-CNNs for Protein-Ligand Affinity Prediction

- Centering a 25×25×25 Å grid with a spacing of 1Å around the geometric center of the ligand
- The position of each atom within the grid was mapped to a voxel
- Additional chemical properties of the atoms are incorporated as additional channels
- Convolutional Neural Network reduces 3D grid to a single pK_D value representing an affinity prediction

➔ 3D-CNNs were the first structure-based models to outperform sequence-based affinity prediction



RosENet 3D-CNN for binding affinity prediction: A 25x25x25 Å grid around the center of the ligand molecule is defined and featurized with different properties of the underlying protein and ligand atoms (here aromatic carbon atoms, hydrogen bond donors and electrostatic energies). The different property channels are then reduced to a single value with a 3D-convolutional neural network.

Hassan-Harrirou, H., Zhang, C. & Lemmin, T. RosENet: Improving Binding Affinity Prediction by Leveraging Molecular Mechanics Energies with an Ensemble of 3D Convolutional Neural Networks. *J. Chem. Inf. Model.* **60**, 2791–2802 (2020).

3D-Convolutional Neural Networks (3D-CNNs)

Drawbacks

- **Model size:** Significantly larger compared to 2D-CNNs, model parameters grow cubically with the resolution of the grid
- **Sparsity:** Grids often contain a significant portion of voxels representing empty or homogeneous regions that do not contribute meaningful information.
- **Rotational Invariance:** To achieve invariant predictions regardless of the spatial orientation of the objects in the input data, it is necessary to present the objects in multiple orientation

➔ 3D-CNNs usually require large training datasets to perform well and avoid overfitting.

➔ Training and inference require significant memory and processing power

➔ Application to large datasets of structural data is limited (e.g. molecular dynamics simulations)

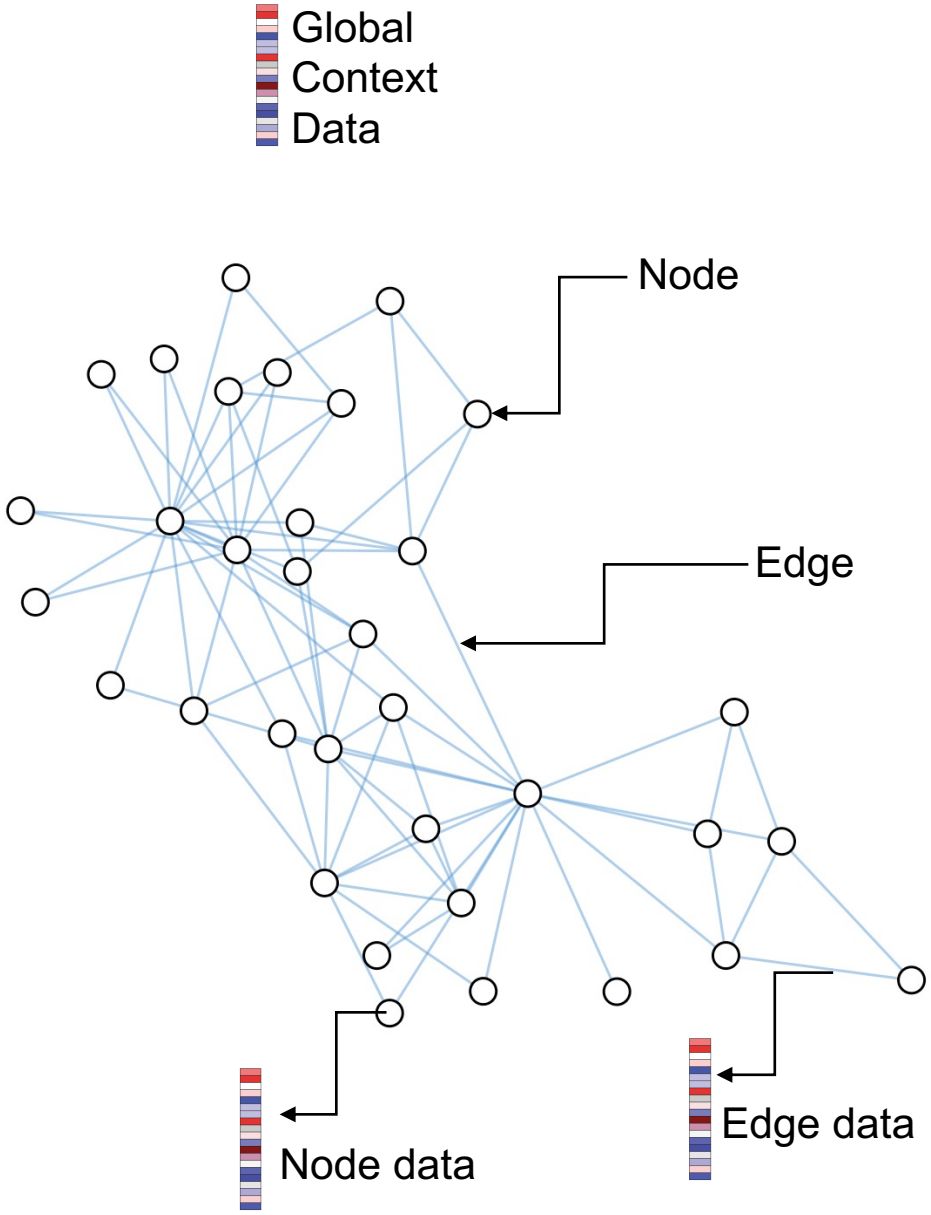
Structure-based AI models

→ 3D-Convolutional Neural Networks
→ Introduction to Graphs and Graph Neural Networks
→ Graph-based models

Introduction to Graphs and GNNs

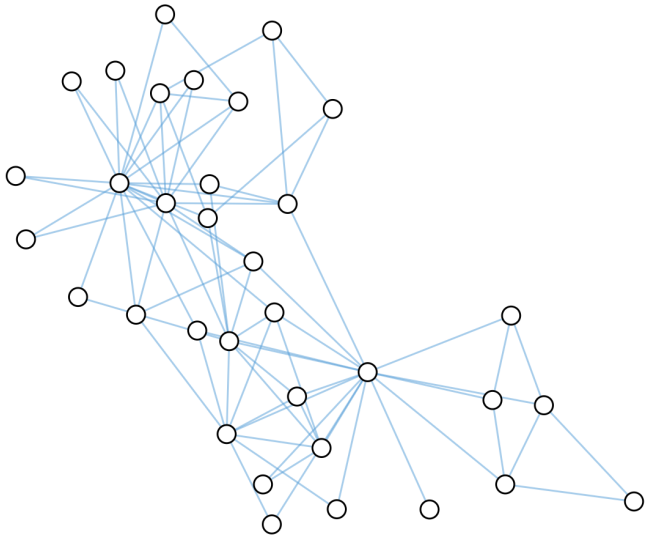
- Graphs describe a set of objects (nodes) and the connections between them (edges).
- Information can be stored in each node and each edge of a graph.
- Powerful representation of data, especially for non-Euclidean and unordered data
  - Social network with friendships
  - Citation networks
  - Atoms in molecules

➔ **Graph Neural Networks are Neural Networks developed to operate on graphs**

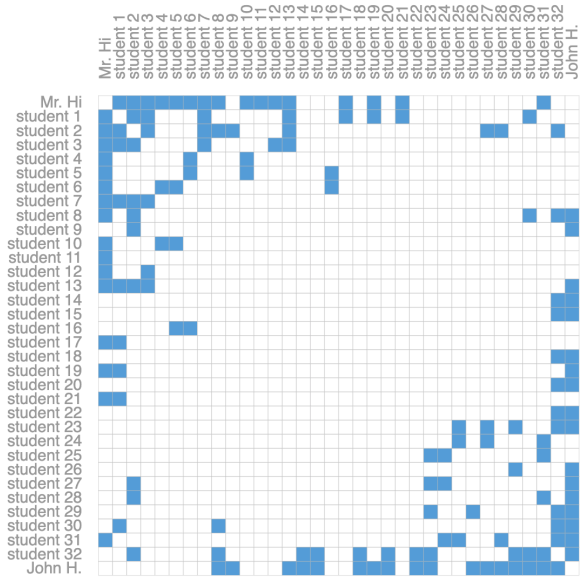


Introduction to Graphs and GNNs

The connectivity of a graph is saved in a square matrix called adjacency matrix $(n_{nodes} * n_{nodes})$. The elements of the matrix indicate whether pairs of nodes are connected or not



Social interactions as a graph

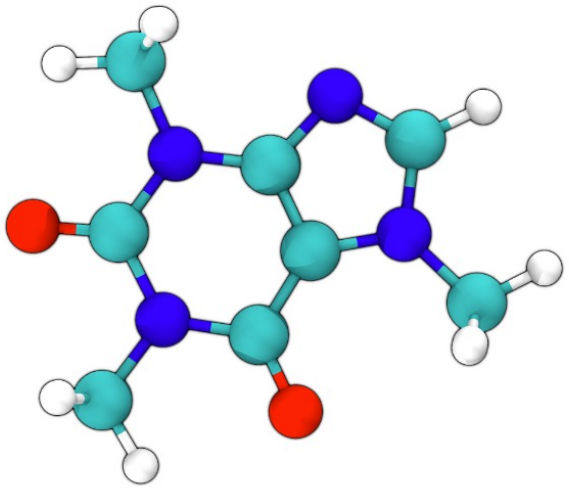


Adjacency matrix

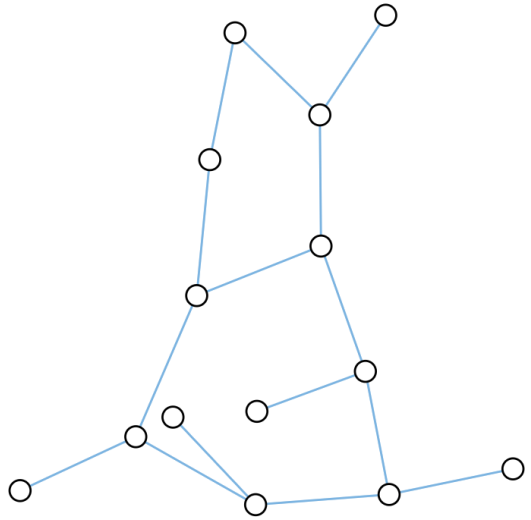
Social Networks: We can build a graph representing groups of people by modelling individuals as nodes, and their relationships as edges.

Introduction to Graphs and GNNs

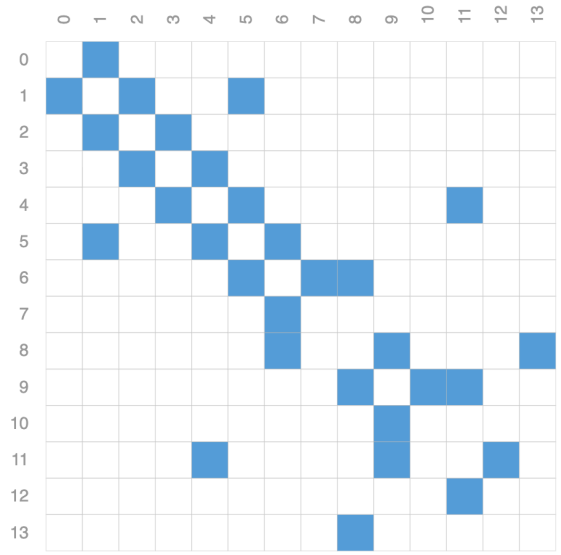
The connectivity of a graph is saved in a square matrix called adjacency matrix $(n_{nodes} * n_{nodes})$. The elements of the matrix indicate whether pairs of nodes are connected or not



Molecule



Molecule as a graph



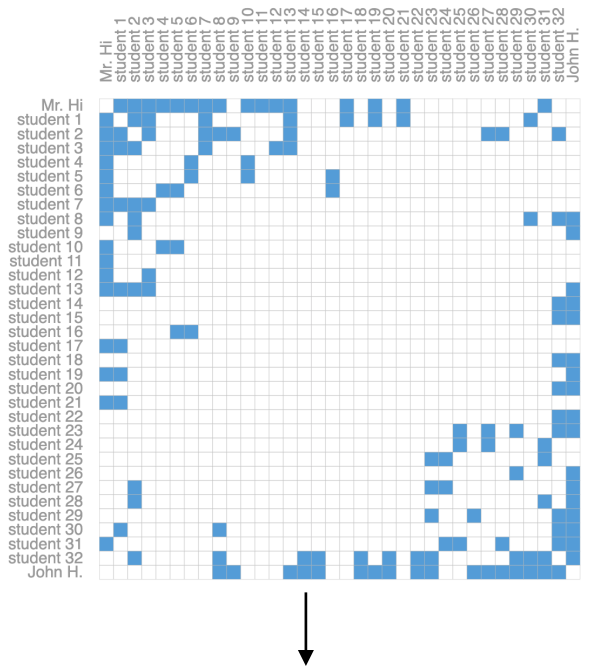
Adjacency matrix

Molecules: It's a very convenient and natural abstraction to describe molecules as graphs, where nodes are atoms and edges are covalent bonds.

Graph Description in Matrix Format

Graph have up to four types of information that we want to use to make predictions:
Node features, edge features, connectivity and global context

1. Assign each node an index i
2. Node features: Store the features of node i in row i of a node feature matrix
3. Connectivity: Adjacency matrix or edge index
4. Edge feature matrix: Store the features of the edges in the same order as in the edge index
5. Global context: A vector of graph-level data



Adjacency Matrix:
These matrices are easily tensorisable, but they are often very large and sparse. The number of nodes in a graph can be on the order of millions, and the number of edges per node can be highly variable. Often, this leads to very sparse adjacency matrices, which are space-inefficient.

```
[2 1]
[3 1] [3 2]
[4 1] [4 2] [4 3]
[5 1]
[6 1]
[7 1] [7 5] [7 6]
[8 1] [8 2] [8 3] [8 4]
[9 1] [9 3]
[10 3]
[11 1] [11 5] [11 6]
[12 1]
[13 1] [13 4]
[14 1] [14 2] [14 3] [14 4]
[17 6] [17 7]
[18 1] [18 2]
[20 1] [20 2]
[22 1] [22 2]
[26 24] [26 25]
[28 3] [28 24] [28 25]
[29 3]
[30 24] [30 27]
[31 2] [31 9]
[32 1] [32 25] [32 26] [32 29]
[33 3] [33 9] [33 15] [33 16] [33 19] [33 21] [33 23] [33 24] [33 30] [33 31] [33 32]
[34 9] [34 10] [34 14] [34 15] [34 16] [34 19] [34 20] [34 21] [34 23] [34 24] [34 27] [34 28] [34 29] [34
30] [34 31] [34 32] [34 33]
```

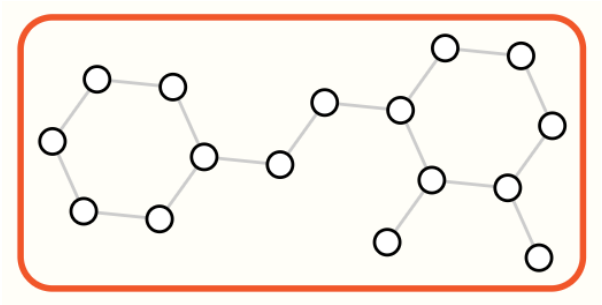
Edge Index:
One elegant and memory-efficient way of representing sparse matrices is as adjacency lists. These describe the connectivity of edge e_k between nodes n_i and n_j as a tuple (i,j) in the k[th] entry of an adjacency list.

Since we expect the number of edges to be much lower than the number of entries for an adjacency matrix (n^2_{nodes}), we avoid computation and storage on the unconnected parts of the graph.

What tasks do we want to perform on this data?

Graph Level Tasks:
We predict a single property for an entire graph

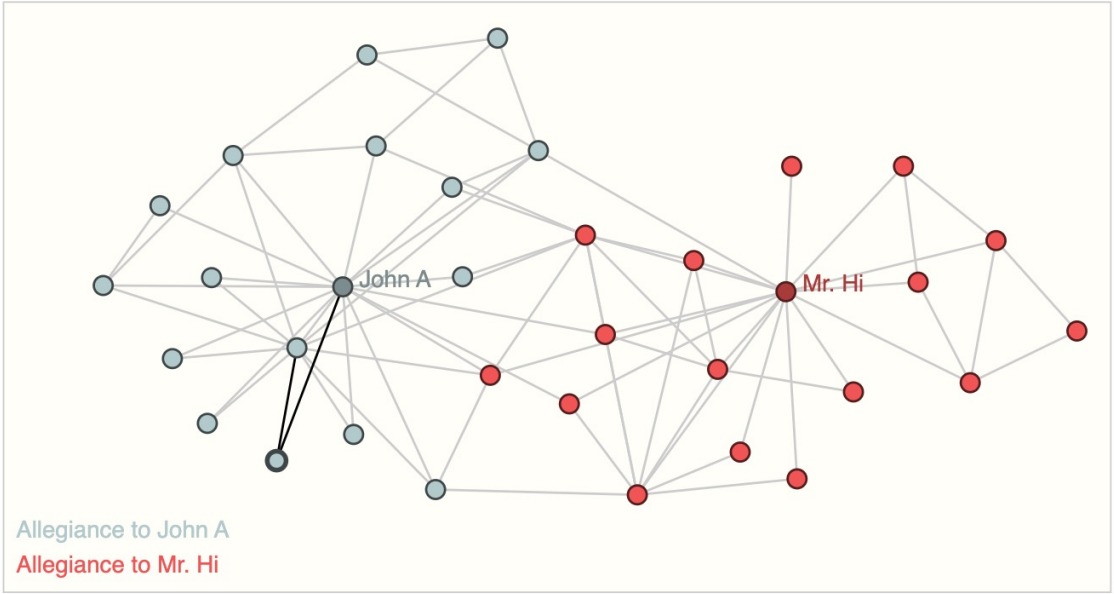
Example: Is this molecule an antibiotic?



Edge-Level Tasks:
We predict the property or presence of edges in a graph

Example: Friendship Suggestions on Facebook

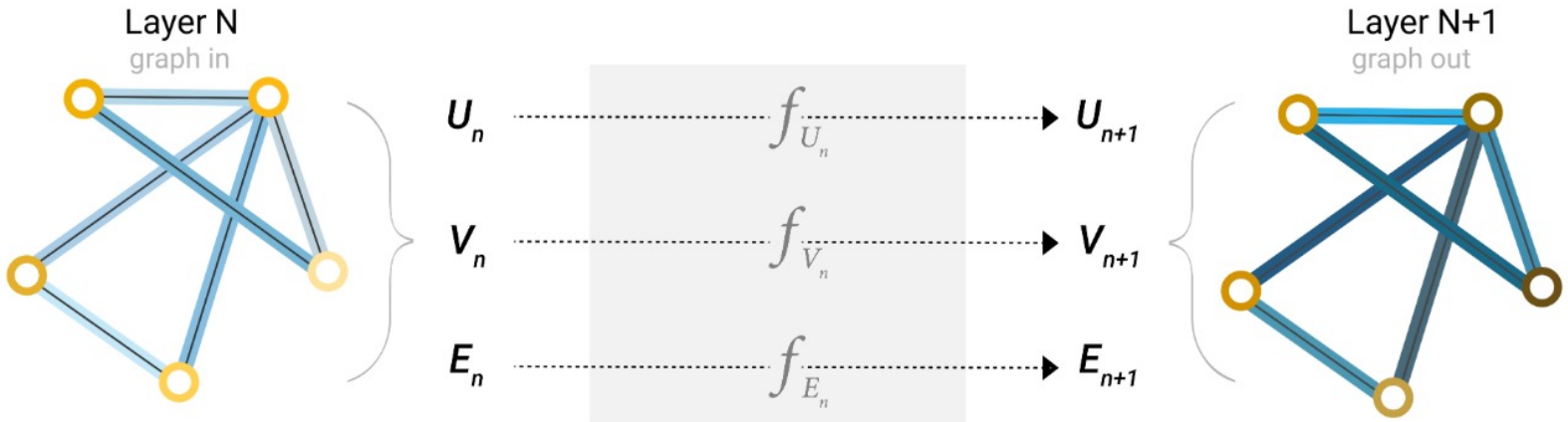
Node-Level Tasks:
We predict some property for each node in a graph



Zach's karate club dataset: A classic example of a node-level prediction problem. The dataset is a single social network graph made up of individuals that have sworn allegiance to one of two karate clubs after a political rift. As the story goes, a feud between Mr. Hi and John H creates a schism in the karate club. The nodes represent individual karate practitioners, and the edges represent interactions between these members outside of karate. The prediction problem is to classify whether a given member becomes loyal to either Mr. Hi or John H, after the feud.

Graph Neural Networks – Basic Principles

A GNN is an optimizable transformation on all attributes of the graph (nodes, edges, global-context) that preserves graph connectivity

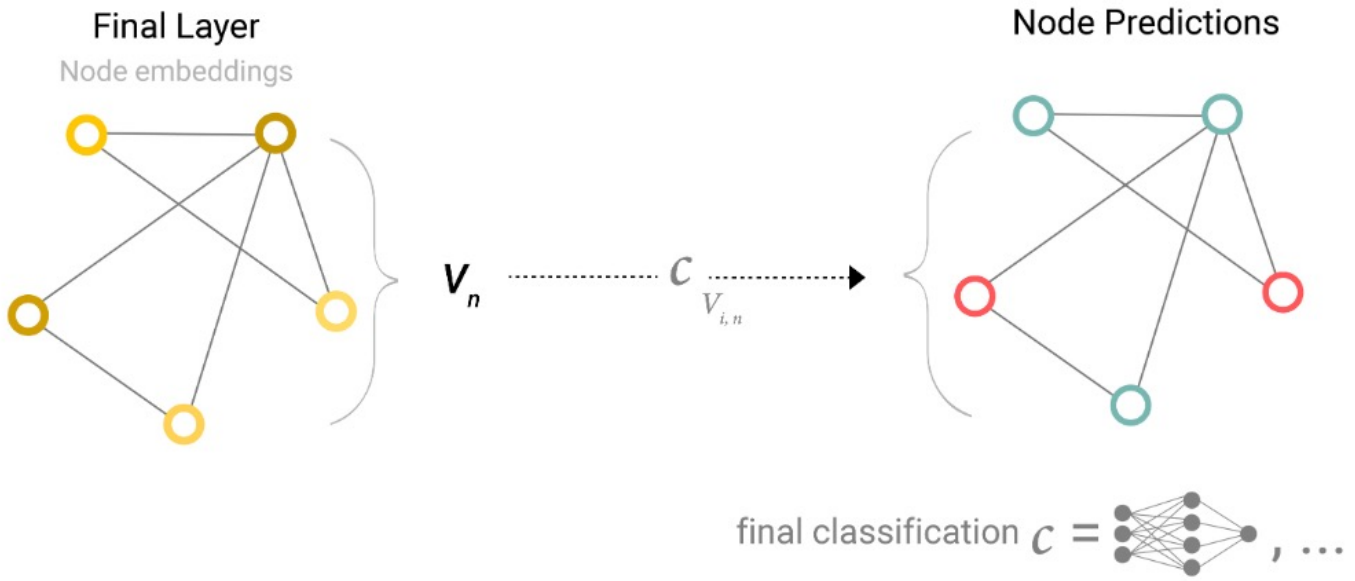


Transformation of graph attributes with update functions f_U, f_V and f_E: Each attribute (global context U, nodes V and edges E) is transformed and updated in a GNN layer. The output of the GNN is the same graph with updated node, edge and global context representations.

- GNNs adopt a “graph-in, graph-out” architecture
- Accept a graph as input, with information loaded into its nodes, edges and global-context, and progressively transform these embeddings, without changing the connectivity of the input graph.

Graph Neural Networks – Basic Principles

A GNN is an optimizable transformation on all attributes of the graph (nodes, edges, global-context) that preserves graph connectivity



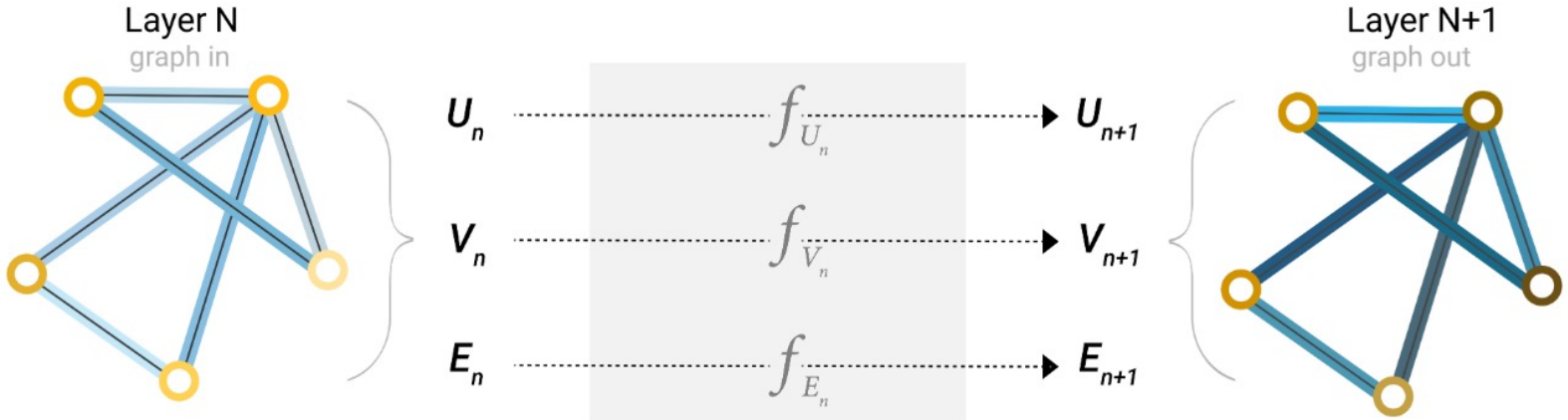
Node classification:
Given a graph with transformed node embeddings (V_n), apply a fully connected neural network C to derive a prediction for each node.

Binary predictions on nodes:

- To get a prediction for each node in the graph, apply a linear classifier to transformed node embeddings
- The same can be done for edge classification
- Edge features or global-context information can be included through pooling functions.

Graph Neural Networks – Basic Principles

A GNN is an optimizable transformation on all attributes of the graph (nodes, edges, global-context) that preserves graph connectivity



Transformation of graph attributes with update functions f_U, f_V and f_E: Each attribute (global context U, nodes V and edges E) is transformed and updated in a GNN layer. The output of the GNN is the same graph with updated node, edge and global context representations.

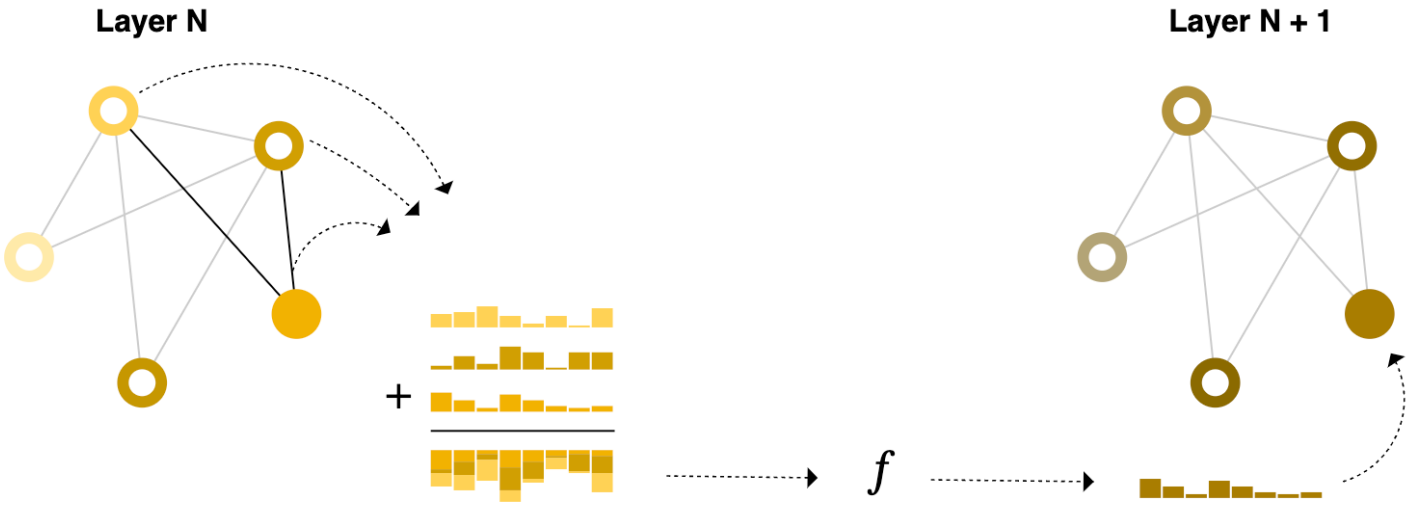
To update the graph features GNNs apply the "message passing neural network" framework:

➔ Neighboring nodes or edges exchange information and influence each other's updated embeddings

➔ Makes the learned embeddings aware of graph connectivity

Graph Neural Networks – Basic Principles

A GNN is an optimizable transformation on all attributes of the graph (nodes, edges, global-context) that preserves graph connectivity



The three steps of message-passing in graphs:

1. Aggregate information from adjacent nodes
2. Transform information
3. Update graph with new information

Message-passing works in three steps

1. For each node in the graph, gather all the neighboring node embeddings (or messages)
2. Aggregate all messages using an aggregation function (including the node's own embedding)
3. Pass the pooled messages through an update function, usually a learned neural network

Graph Neural Networks – Basic Principles

Message-passing vs. **Image convolution.**

- Both are operations to aggregate and process the information of an element's neighbors in order to update the element's value
- In graphs the elements are nodes, and in images the elements are pixels
- The number of neighboring nodes in a graph can be variable, unlike in an image where each pixel has a fixed number of neighboring elements

➔ **The message-passing process in graphs is also called graph convolution**

As in 2D images processed with CNNs, we can stack graph convolutional layers. After three layers, a node has information about the nodes three steps away from it

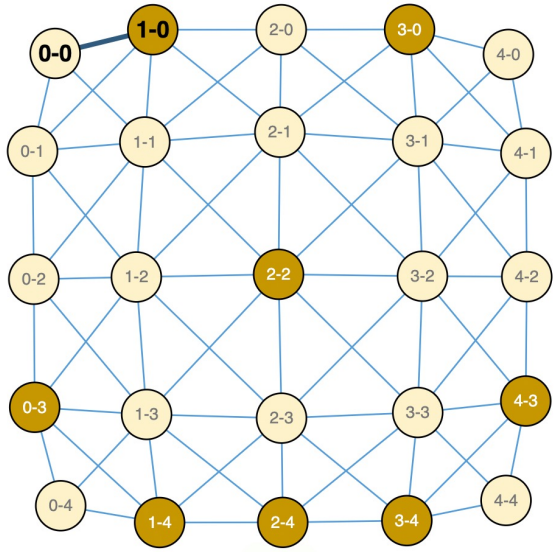


Image Convolution can be thought of as graph convolution on a graph with regular structure, where each node represents a pixel and is connected to adjacent pixels by and edge.

Graph Convolutional Layers

Several types of graph convolutional layers are commonly used, each with its own method of aggregating information from neighboring nodes.

Graph Convolutional Network Layer (GCNConv):

The updated node features $x_i^{(l+1)}$ of node i at layer $(l+1)$ are computed as

$$x_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{d_j d_i}} W^{(l)} x_j^{(l)}\right)$$

$\mathcal{N}(i)$	neighboring nodes of node i
$x_j^{(l)}$	node features of neighbor j layer l
d_i	the degree of node i
$W^{(l)}$	learnable weight matrix at layer l,
σ	activation function (e.g. ReLU)

Graph Attention Network Layer (GATConv)

The updated node features $x_i^{(l+1)}$ of node i at layer $(l+1)$ are computed as

$$x_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} W^{(l)} x_j^{(l)}\right)$$

$a_{ij}^{(l)}$	attention score between nodes i and j
$\mathcal{N}(i)$	neighboring nodes of node i
$x_j^{(l)}$	node features of neighbor j layer l
$W^{(l)}$	learnable weight matrix at layer l
σ	activation function (e.g. ReLU)

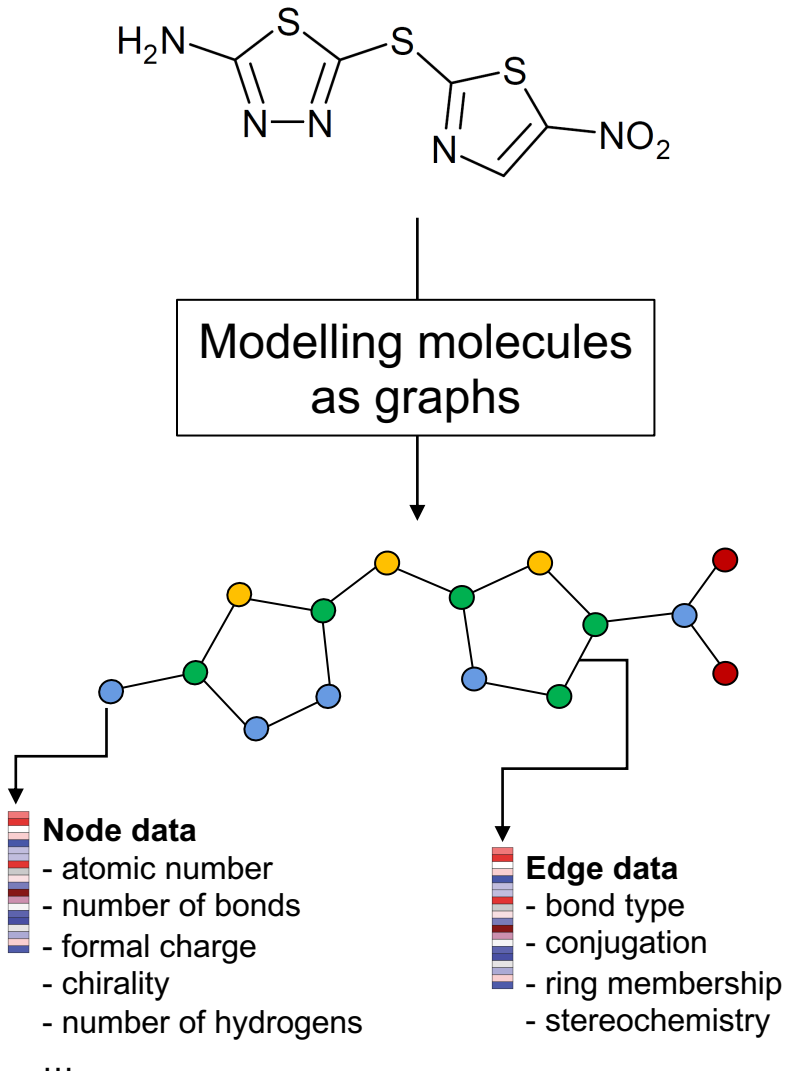
Structure-based AI models

→ 3D-Convolutional Neural Networks
→ Introduction to Graphs and Graph Neural Networks
→ Graph-based models

Antibiotic Discovery with GNNs (Stokes et al. 2020)

Graph Neural Network capable of predicting molecules with antibacterial activity

- **Graph-level binary classification problem** (antibacterial / not antibacterial)
- **Training set:** Library of 2560 drug molecules of diverse structure and function containing 120 molecules with known growth inhibition against E.Coli
- **Graph Modelling:** Generate graph representation of molecules, include chemical properties as edge and node features
- **Graph Convolutional Neural Network**
  - GCNConv convolution applied to edges and nodes, followed by global add pooling
  - Feed-forward neural network that outputs a predicted probability of growth-inhibition of E.Coli



Stokes, J. M. et al. A Deep Learning Approach to Antibiotic Discovery. Cell 180, 688-702.e13 (2020).

Antibiotic Discovery with GNNs

Make predictions on a large molecular library
(Drug Repurposing Hub molecule library n=6111)

- Inference with an ensemble of models trained on twenty random folds of the training data
- Among the **99** molecules with **highest** model prediction, **51** molecules displayed growth inhibition against E.Coli
- Among the **63** molecules with **lowest** model prediction, **two** molecules displayed growth inhibition against E.Coli

➔ **Discovery of Halicin, which displays bactericidal activity against a wide range of bacteria**

➔ **Very low structural similarity to its nearest neighbor antibiotic shows that the model was capable of generalization, accessing new antibiotic chemistry**

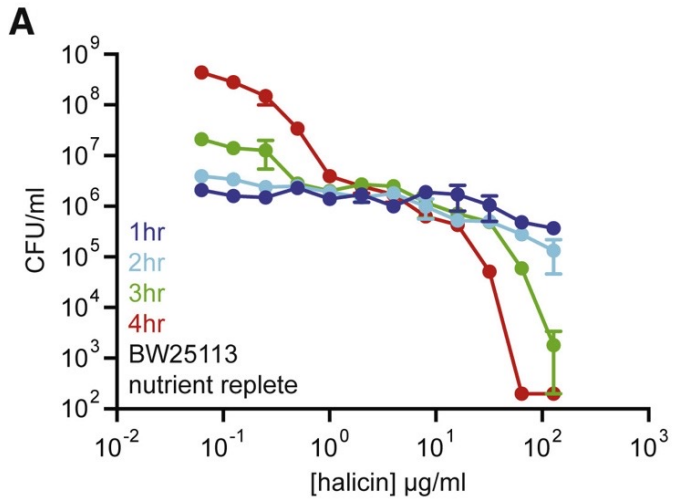


Figure A: Killing of E. coli in LB media in the presence of varying concentrations of halicin after 1 h (blue), 2 h (cyan), 3 h (green), and 4 h (red).

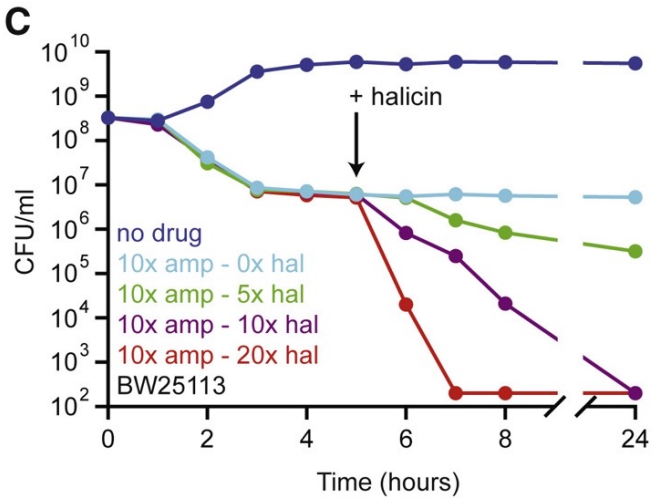
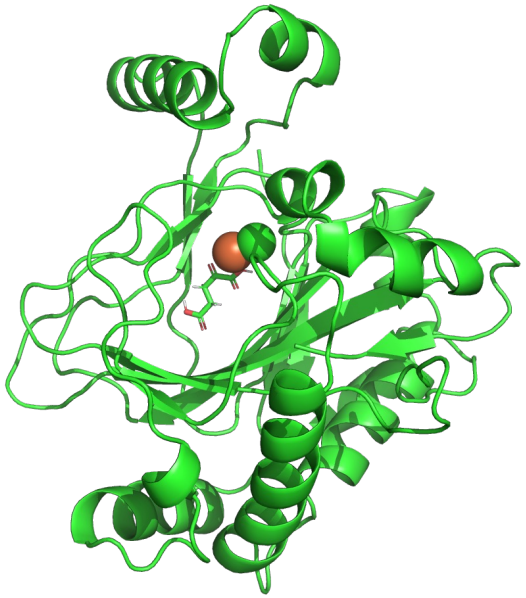


Figure B: Killing of E. coli persisters by halicin after treatment with 10 µg/mL of ampicillin. Light blue is no halicin. Green is 5× MIC halicin. Purple is 10× MIC halicin. Red is 20× MIC halicin.

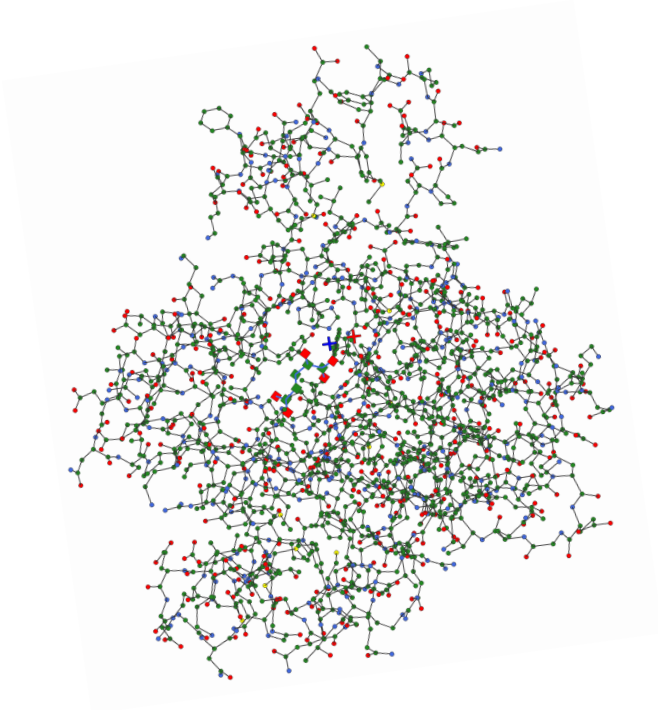
Stokes, J. M. et al. A Deep Learning Approach to Antibiotic Discovery. Cell 180, 688-702.e13 (2020).

Graph Neural Networks for Proteins

Molecules: Graph modelling can also be extended to proteins, as proteins are large molecules (with a fixed vocabulary of building blocks)



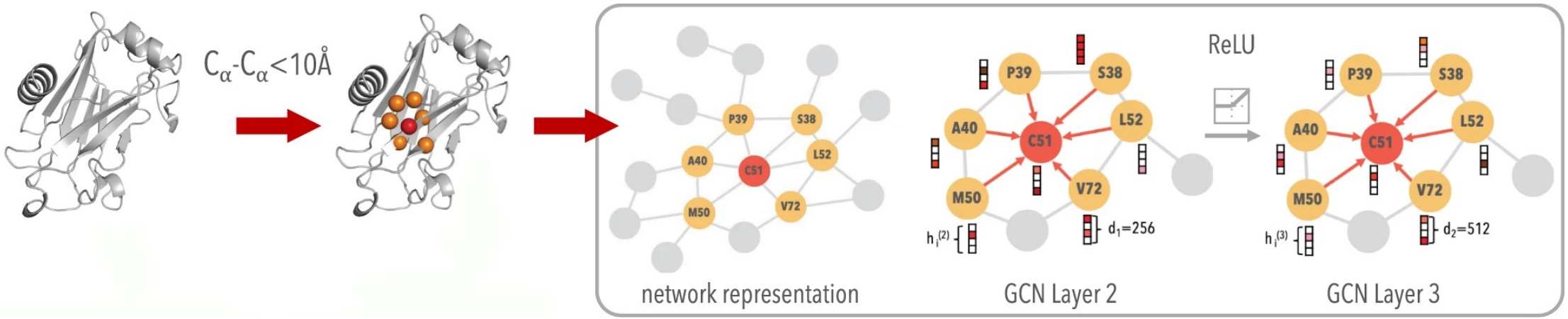
Modelling proteins as graphs



Graph Neural Networks for Proteins - DeepFRI

DeepFRI - Graph Convolutional Network for predicting protein functions

- Protein is modelled as a point cloud of amino acids (nodes = amino acids)
- Amino acids which are less than 10Å apart are connected through an edge
- The resulting contact map is the input to the convolutional neural network



Gligorijević, V. et al. Structure-based protein function prediction using graph convolutional networks. Nat. Commun. 12, 3168 (2021).

Graph Neural Networks for Proteins - DeepFRI

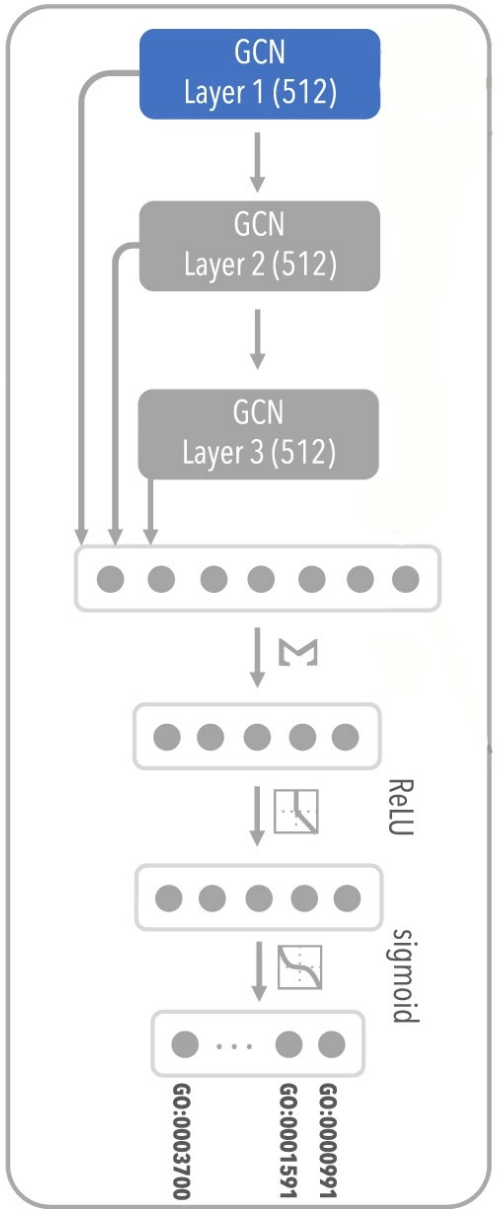
Featurization: Amino acid embeddings derived from a protein language model serve as initial node features.

Model Architecture:

- Three layers of graph convolution (GATConv) with skip connections
- global add pooling (to reduce the representation into a vector)
- Fully connected neural network to derive protein classification from the vector representation of the protein.

➔ **Convolution sends messages between residues that are distant in the primary sequence, but close to each other in the 3D space, leading to better protein representations.**

➔ **Simple GCN outperforms other ML tools for protein function classification**

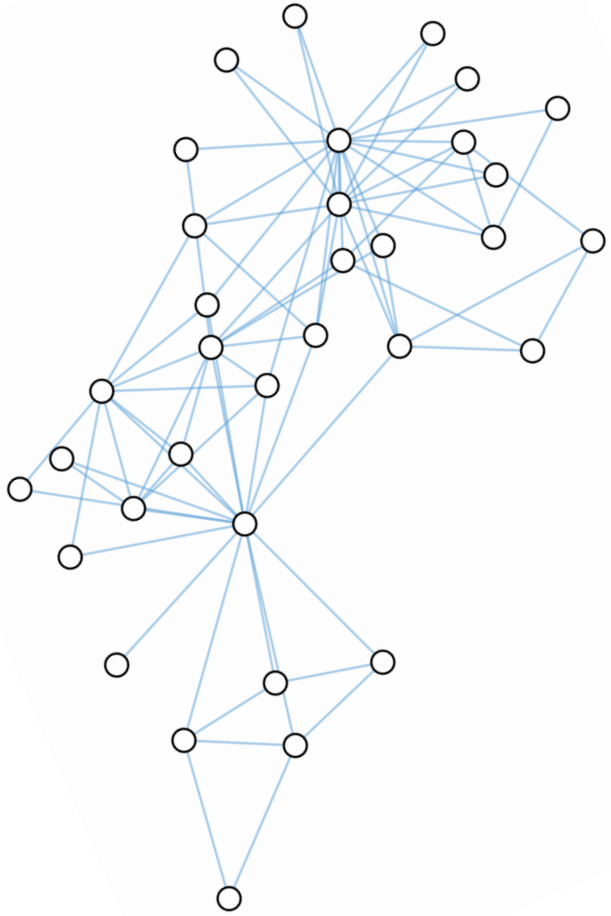


Gligorijević, V. et al. Structure-based protein function prediction using graph convolutional networks. Nat. Commun. 12, 3168 (2021).

Graph Neural Networks - Conclusions

Graph Neural Networks offer many advantages for modelling and processing 3D objects, such as molecules

- Can handle data defined in irregular domains, without mapping onto a regular grid
- Graph representations are invariant to translation and rotation
- Much sparser representation of 3D objects than voxel grids
- Many possibilities for data integration
- Efficient update mechanisms with graph convolution and weight sharing



➔ **GCNs perform well in tasks where an understanding of the structure and the interconnectivity of data is important**

➔ **Models usually perform very well with relatively few parameters**

Summary

Text-based AI models

- Proteins and small molecules can be represented in text and handled efficiently with methods of Natural Language Processing
- Language models trained on these text representations can generate informative latent-space representation and are even able to predict the structure of protein

Structure-based AI models

- Perform better in many tasks where information on the spatial placement of the elements is crucial
- Graphs are ideally suited to model and process non-Euclidean and unordered data such as molecules and social networks
- Graph Neural Networks are efficient models for learning tasks on graph-like data

Conclusion

- The engineering of proteins and small molecules is challenging due to the vast size of the search space
- AI models can (and will) help to navigate these search spaces and accelerate development processes

Generative AI for *de novo* proteins design

→ RF diffusion for backbone design
→ ProteinMPNN for sequence design

Protein engineering vs. *de novo* design

Protein engineering

- Explore the protein space by modifying the sequence of a natural proteins to optimize its properties
- Resulting proteins share most of their sequence with the protein that served as starting point
- Resulting proteins are structurally similar to a natural proteins

***De novo* protein design**

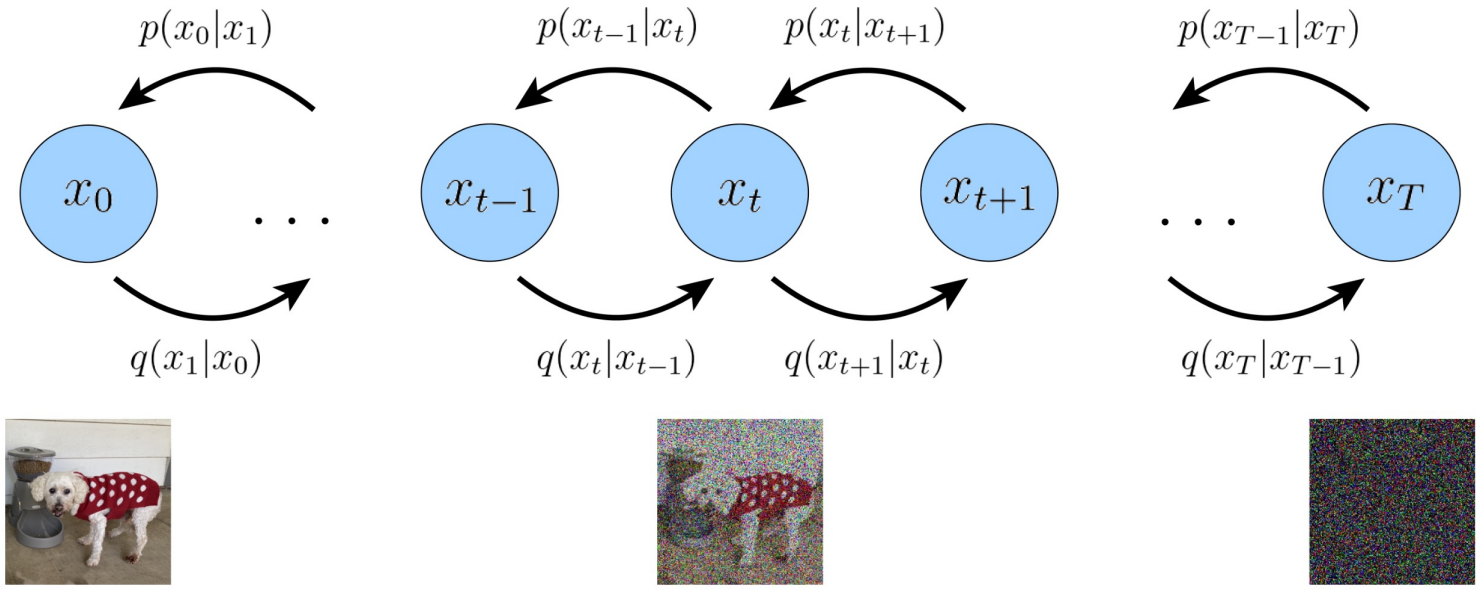
- Designing a novel sequence from scratch
- Does not rely on natural protein sequences, but uses computational methods to design sequences that are potentially very different from all existing sequences

“Evolution has only explored a tiny subset of all proteins that could exist”

Diffusion Models

Probabilistic Diffusion Models achieve state of the art performance in image generation

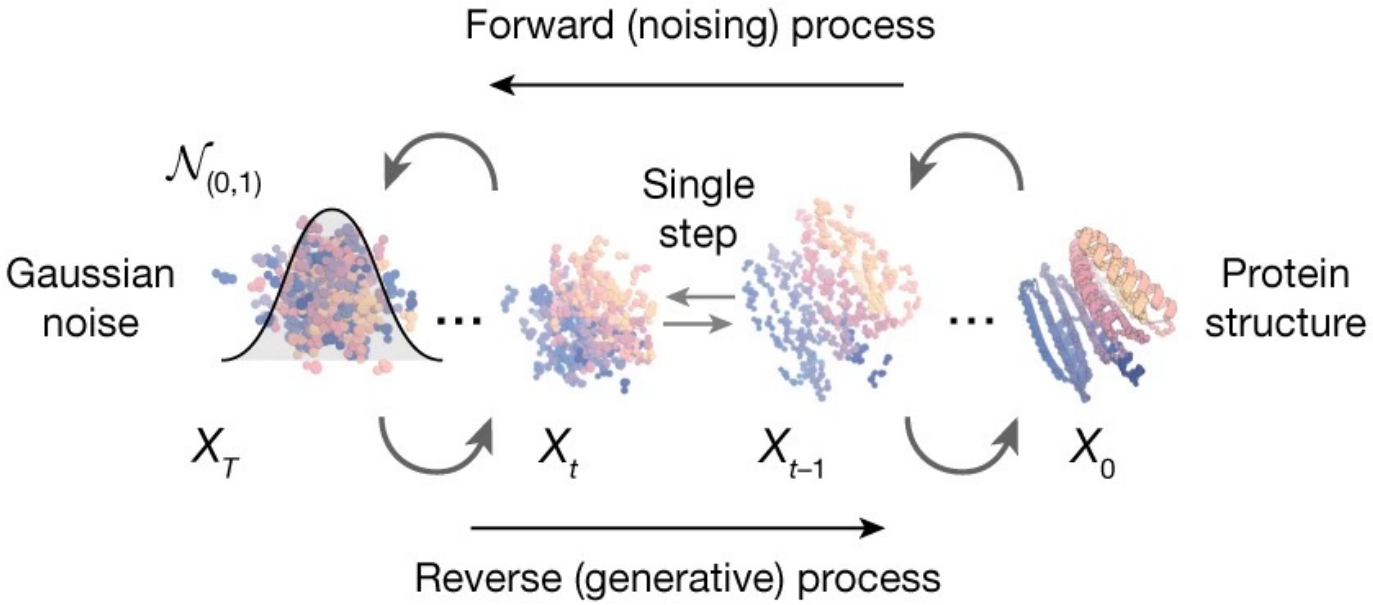
- Add increasing amounts of gaussian noise over a certain number of timesteps
- Train a neural network to undo the steps of noise
- Sample from your known gaussian distribution and feed this pure noise to your models
- Realistic image from the distribution you trained on



Luo, C. Understanding Diffusion Models: A Unified Perspective. *arXiv* (2022) doi:10.48550/arxiv.2208.11970.

Diffusion on protein structures

Diffusion models applied to proteins: Generate realistic protein backbones *de novo*



- Generated training data by noising structures from the Protein Data Bank (PDB)
- Model is trained to recover ground-truth structures from noised inputs
- Generate new structures X_0 through iterative denoising of random noise X_T

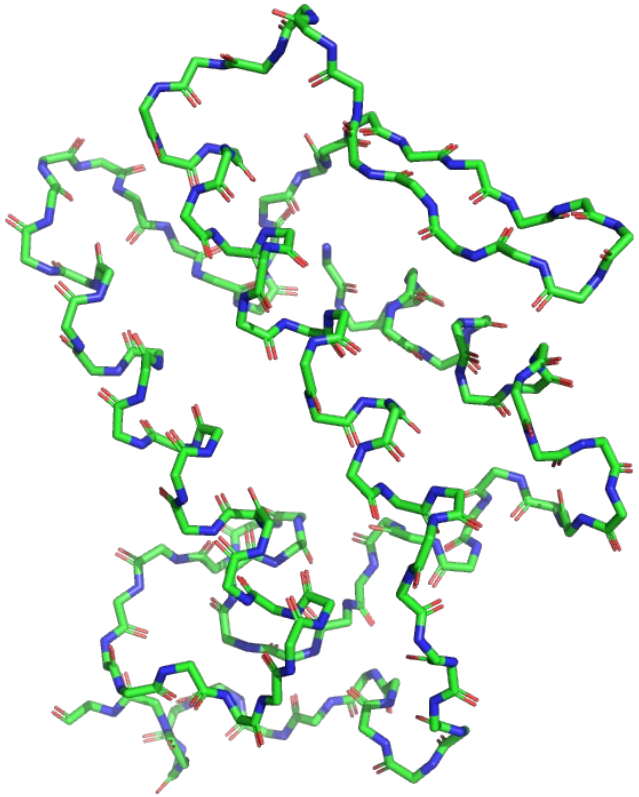
Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RFdiffusion

How to add gaussian noise to protein backbones

1. Frame-based representation of protein backbone with L residues
   - The atoms N, Cα, and C form a nearly planar, rigid triangle
   - Each triangle can be described by a translation and a rotation
   - **Protein Backbone Structure = Set of L translations and rotations**

2. Add 3D gaussian noise to translations (perturb Cα coordinates)
3. Use Brownian motion on the manifold of rotation matrices



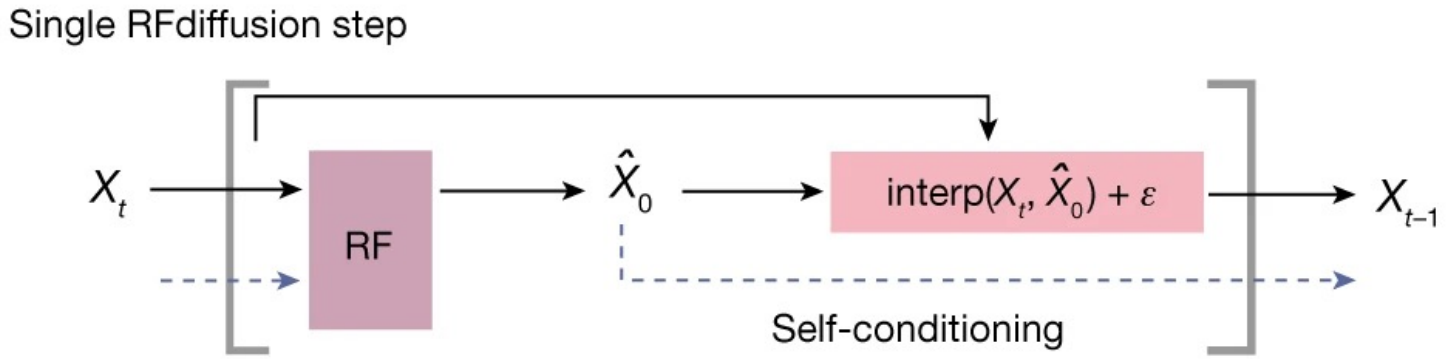
Noising process transforms set of triangles form neatly aligned along a protein backbone into randomly distributed and oriented

→ Train neural network to reverse this process

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RFdiffusion training

Training of RFdiffusion to recover original protein structure from noised protein structures



1. RFdiffusion takes the current coordinates X_t (initially random noise) and makes a prediction of the ground truth coordinates $\hat{X}_0$
2. The next coordinate input to the model (X_{t-1}) is generated by a noisy interpolation towards $\hat{X}_0$
3. Ground-truth prediction $\hat{X}_0$ is not discarded but handed to the next timestep (self-conditioning)

Self-conditioning: The model can use the positions of the coordinates at the current timestep, and the predicted ground truth coordinates of the previous timestep

→ **Markedly improves RFdiffusion performance**

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RoseTTAFold → RFdiffusion

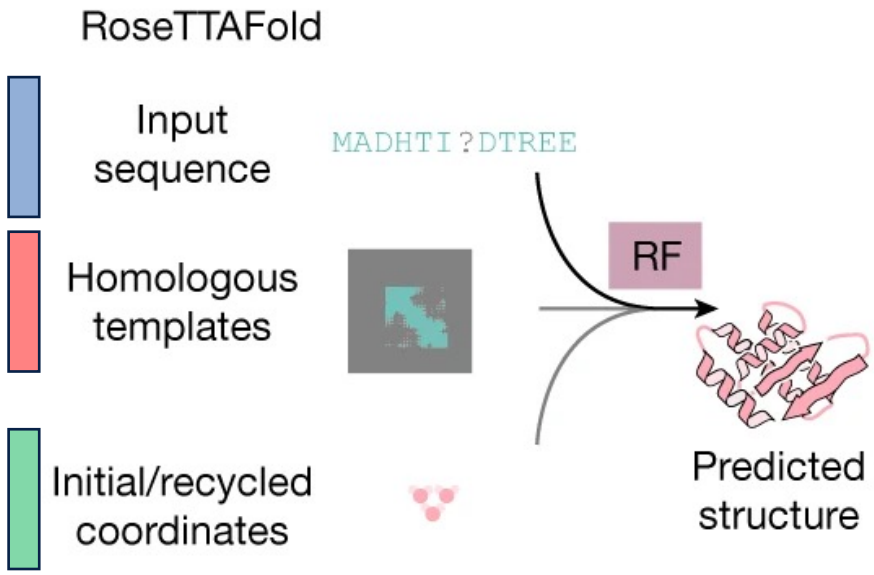
RoseTTAFold (similar to AlphaFold) accurately predicts protein structures

→ Improved diffusion models for protein design could be developed by taking advantage of the deep understanding of protein structure in structure prediction models

→ RFdiffusion was developed through fine-tuning of the RoseTTAFold structure prediction network

RoseTTAFold modelling pipeline

- Given input sequence
- Search for homologous proteins in a databases
- Creates a first structure prediction based on the homologous protein's structure
  - Create pairwise residue distance matrix
  - Create frame-based representation of protein backbone
- Output predicted structure



Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RoseTTAFold → RFdiffusion

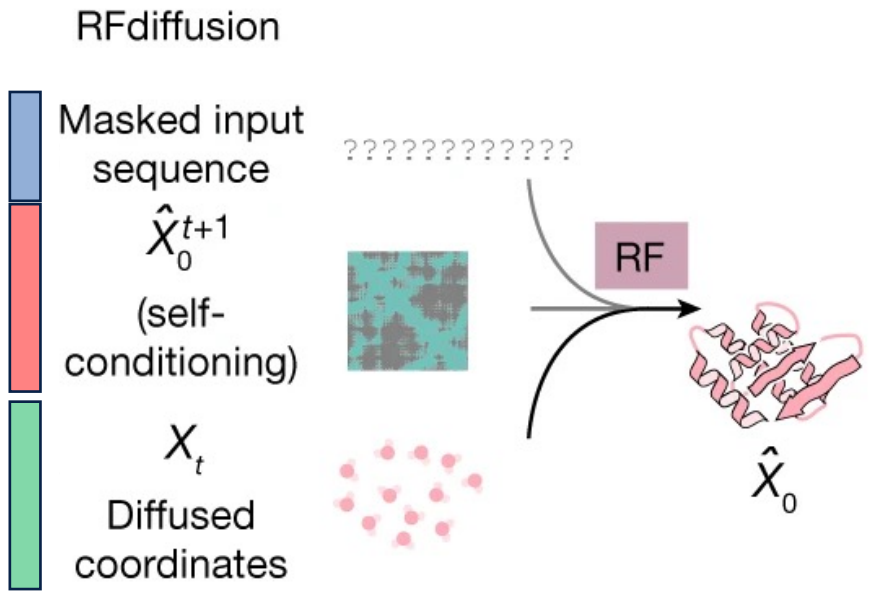
RoseTTAFold (similar to AlphaFold) accurately predicts protein structures

→ Improved diffusion models for protein design could be developed by taking advantage of the deep understanding of protein structure in structure prediction models

→ RFdiffusion was developed through fine-tuning of the RoseTTAFold structure prediction network

RFdiffusion modelling pipeline inputs:

- The sequence input in masked
- Current coordinate prediction at this timestep
- Prediction of the ground-truth coordinates $X_0^{(t+1)}$ of the previous timestep (self-conditioning)
- Output predicted structure



Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RFdiffusion - Summary

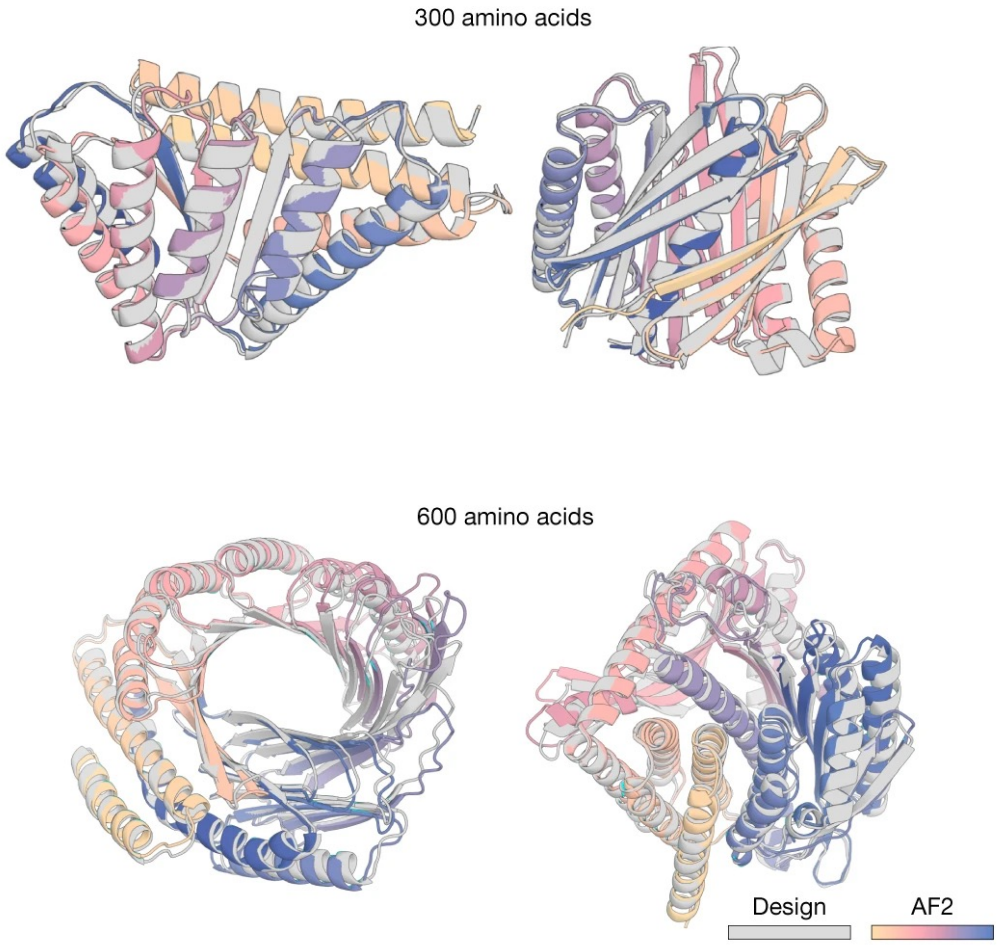
Reverse generative process for backbone design

→ The trained RFdiffusion model is a generative model for protein backbones

→ Starting from random noise, RFdiffusion can generate protein backbones that are highly diverse

→ Many unconstrained designs show little overall similarity to structures seen during training, indicating that the model can generalize beyond its training data

→ Experimental characterization showed that most generated proteins are stable and soluble



RFdiffusion can generate new monomeric proteins of different lengths with no conditioning information

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RFdiffusion - Summary

Conditioning of RFdiffusion for specific designs

→ RFdiffusion can be customized to specific design challenges by addition of external potentials and fine-tuning

→ Guiding at each step of the iterative denosing process towards specific design objectives

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

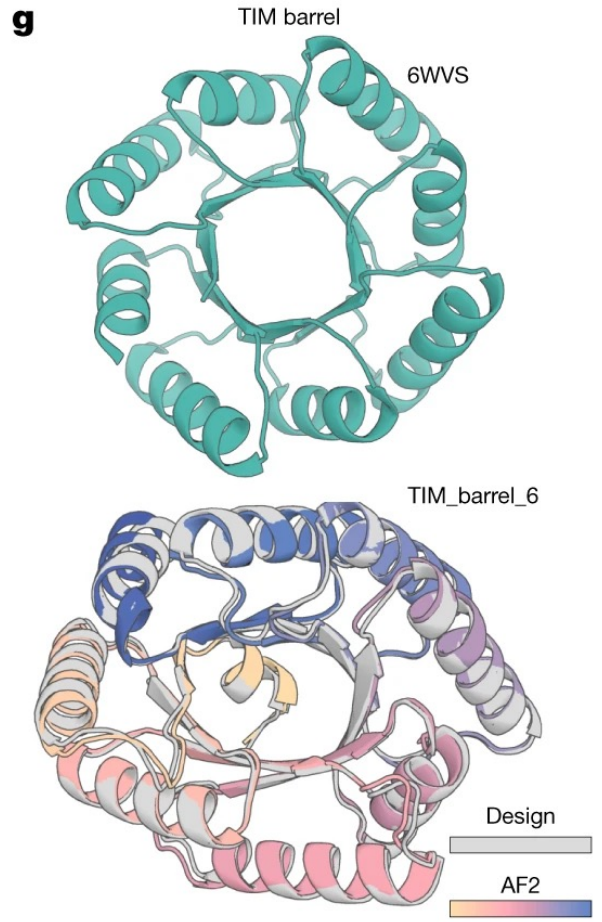
RFdiffusion - Summary

Conditioning of RFdiffusion for specific designs

→ RFdiffusion can be customized to specific design challenges by addition of external potentials and fine-tuning

→ Guiding at each step of the iterative denosing process towards specific design objectives

→ RFdiffusion conditioned with structure of TIM barrel protein 6WVS produced diverse designs with the desired topologies



RFdiffusion can condition on fold information. An example TIM barrel is shown (bottom left), conditioned on the secondary structure and block adjacency of a previously designed TIM barrel, PDB 6WVS. Designs have very similar circular dichroism spectra to PDB 6WVS

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

RFdiffusion - Summary

Conditioning of RFdiffusion for specific designs

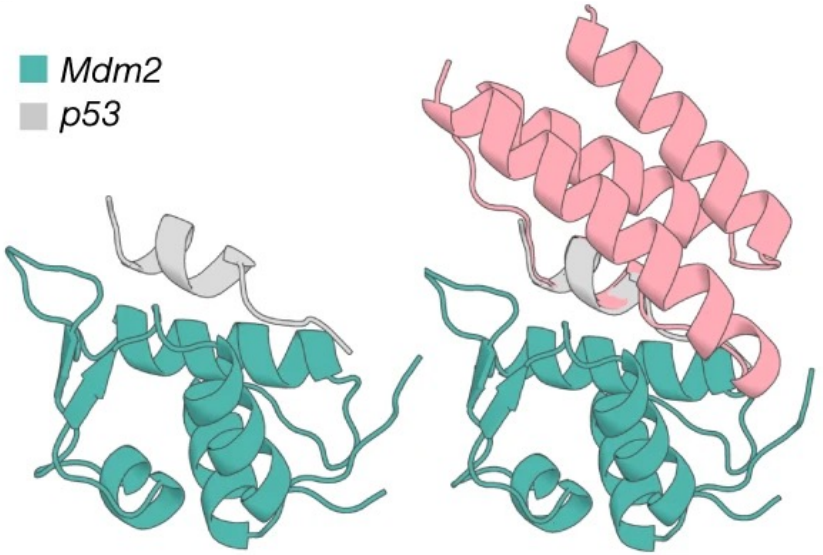
→ RFdiffusion can be customized to specific design challenges by addition of external potentials and fine-tuning

→ Guiding at each step of the iterative denosing process towards specific design objectives

→ RFdiffusion conditioned with structure of TIM barrel protein 6WVS produced diverse designs with the desired topologies

→ RFdiffusion can build scaffolds to hold a desired motif

- In tumours, MDM2 is often overexpressed and binds to p53, preventing p53-mediated cell death
- Designing a competitor protein to bind to MDM2 is attractive
- Take the MDM2-binding helix of the p53 protein into new designs
- RFdiffusion (fine-tuned on complexes) conditioned to include this helix generates designs that strongly bind to MDM2
- Potential cancer drug candidate



RFdiffusion has been used to design a MDM2-binding protein that can compete with p53 and thus prevent p53-MDM2 interaction, which is important for cell-death evasion in many tumours. To incorporate the MDM2-binding helix of p53 in a new design, RFdiffusion was conditioned to include this helix.

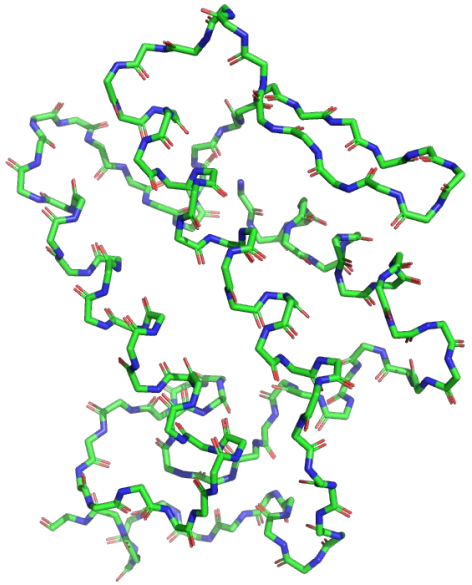
Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

From designed backbones to proteins

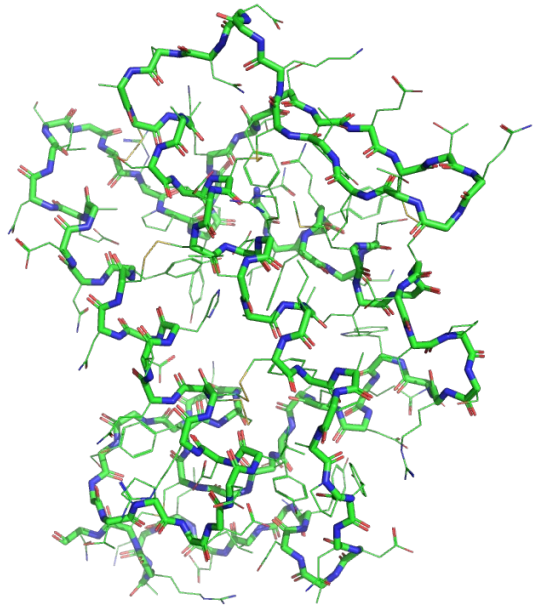
Designed backbones are need amino acids

- Protein backbone designs (e.g., using RFdiffusion) do not include specific amino acid sequences
- To complete the protein design, we need to identify an amino acid sequence that will correctly fold into the designed backbone

Backbone Design



Sequence Design



Folding with AlphaFold2
Check if the generated sequence folds into the correct backbone design

Watson, J. L. *et al.* De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).

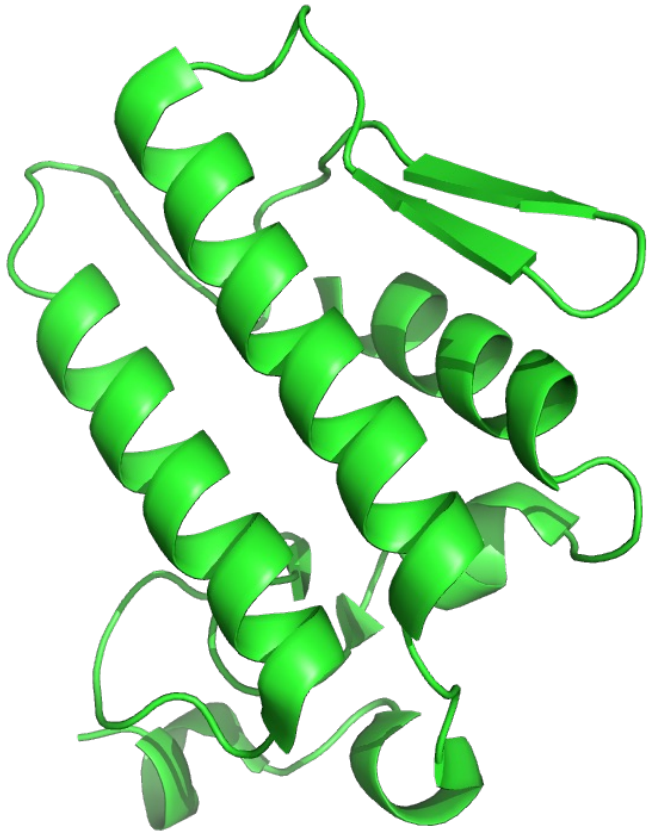
Generative AI for *de novo* proteins design

→ RF diffusion for backbone design
→ ProteinMPNN for sequence design

ProteinMPNN

Generative model for sequence design that integrates graph-based representations of protein structure

- Sequence-from-structure objective: Design a new sequence for a given fixed backbone
- Inverse Folding Problem
- Find a sequence that will fold into the given structure

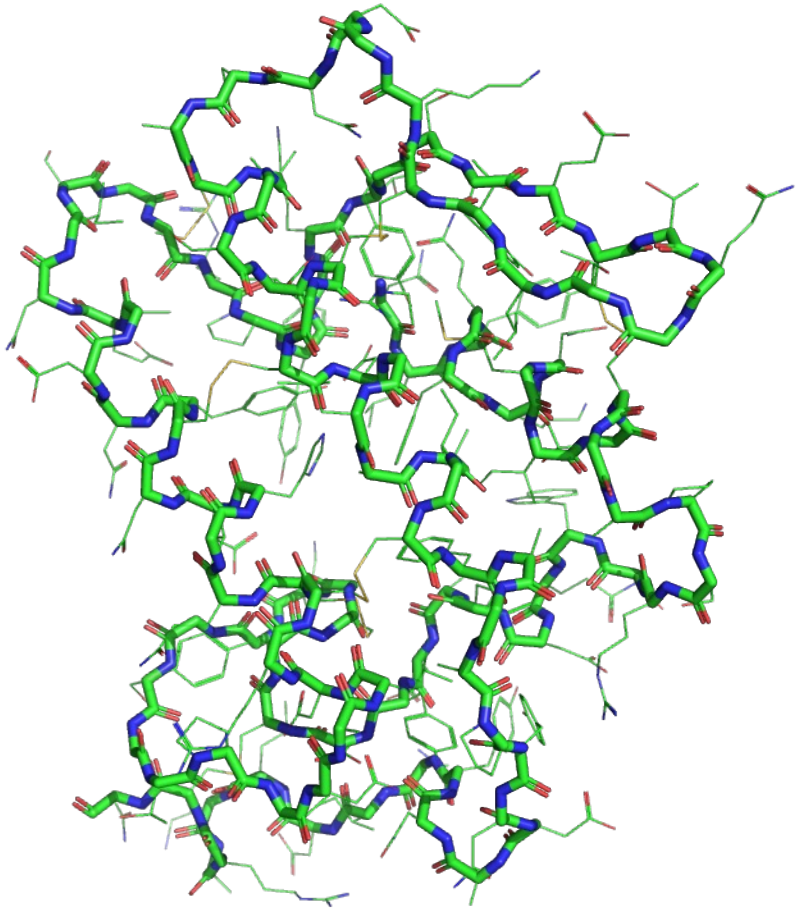


Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

ProteinMPNN

Generative model for sequence design that integrates graph-based representations of protein structure

- Sequence-from-structure objective: Design a new sequence for a given fixed backbone
- Inverse Folding Problem
- Find a sequence that will fold into the given structure

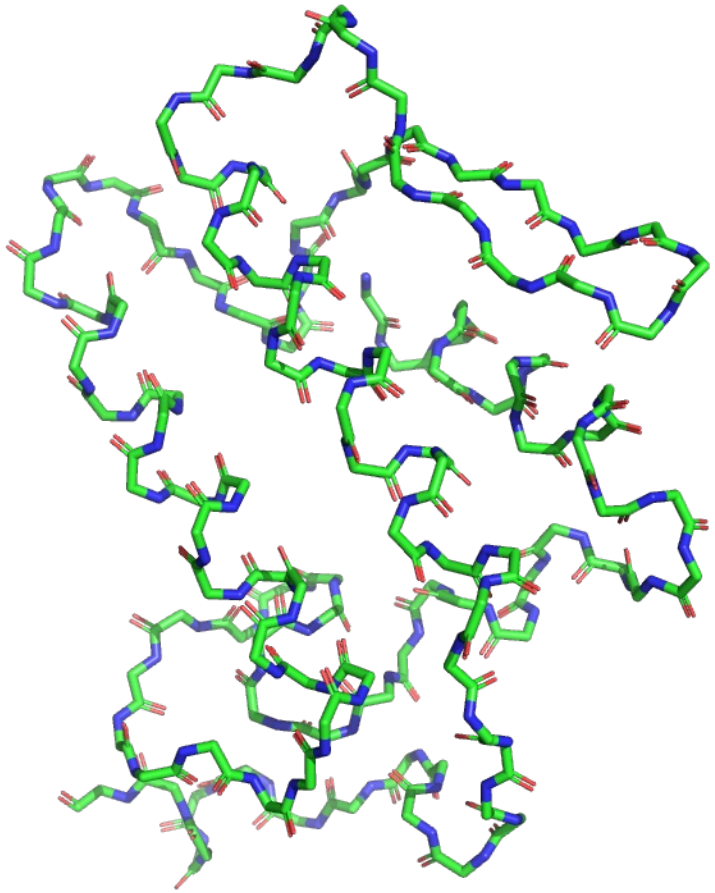


Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

ProteinMPNN

Generative model for sequence design that integrates graph-based representations of protein structure

- Sequence-from-structure objective: Design a new sequence for a given fixed backbone
- Inverse Folding Problem
- Find a sequence that will fold into the given structure



Fixed protein structure is modelled as a graph:

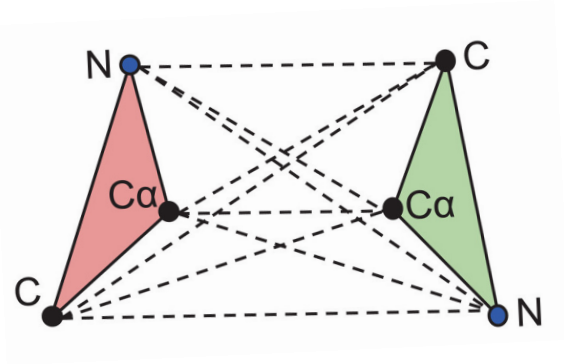
- **Nodes** = amino acids in the fixed structure (specifically the Cα-atom of the unknown amino acid)
- **Edges:** Each node is connected to its 32 nearest neighbor nodes by an edge (each amino acid is connected to its 32 neighboring amino acids)

Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

ProteinMPNN

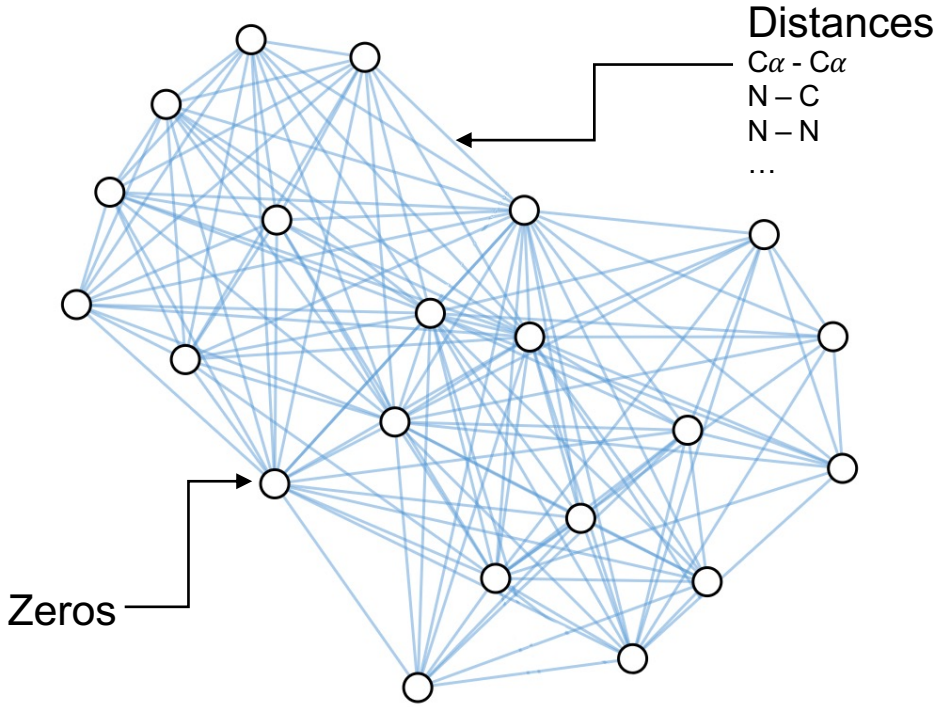
Fixed Protein Structure is modelled as a graph:

- **Node features** = Zeros
- **Edge features** = All distances between the backbone amino acids of both amino acids (to encode the spatial orientation of the amino acids)



Encoder: Message-Passing Neural Network

- Initial node features (zeros) and initial edge features (distances) are transformed with graph convolutions
- Information about the spatial environment of a node is encoded in its node features

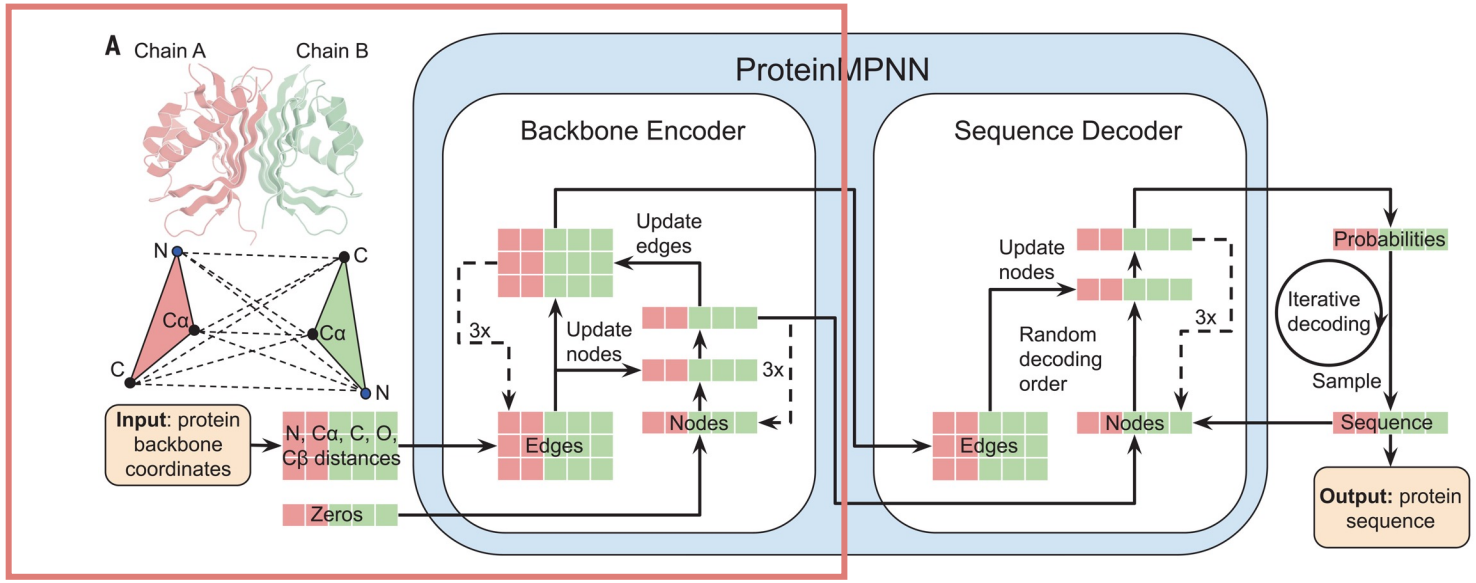


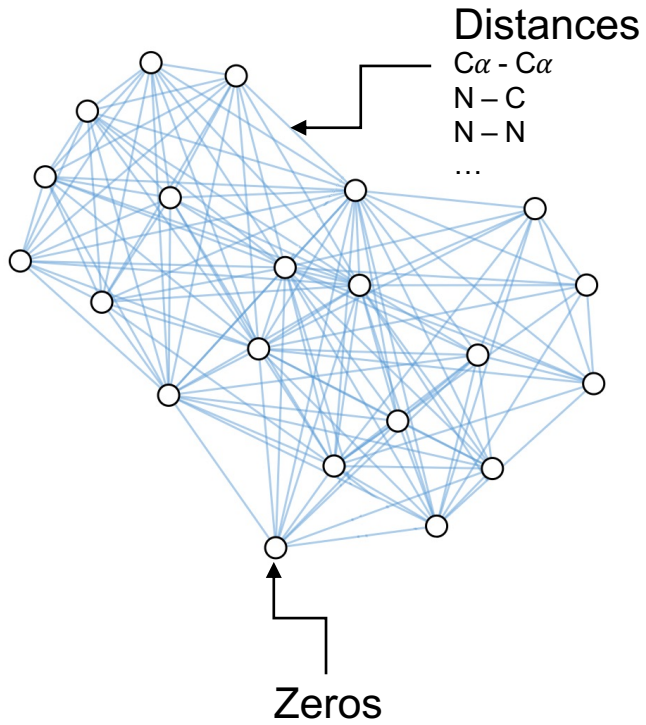
Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

ProteinMPNN

Encoder: Message-Passing Neural Network

- Updates the nodes
- Updates the edges
- Repeat 3x





→ Graph Convolution generates new edge and node embeddings

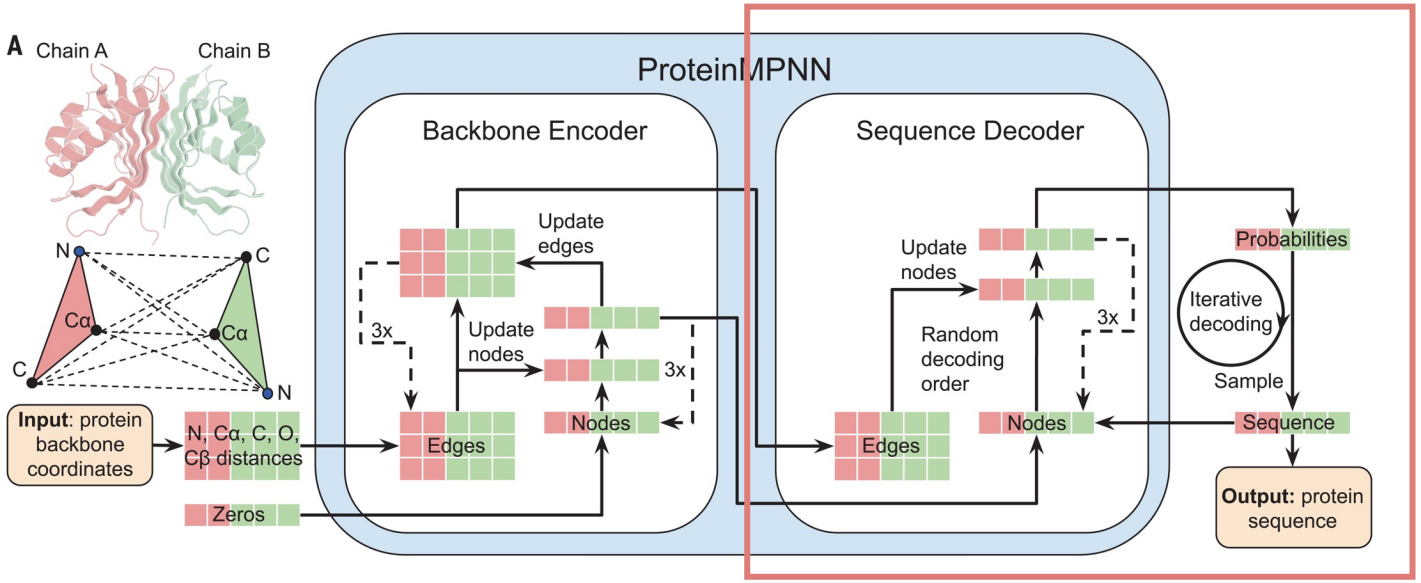
→ Encoding information of the spatial neighborhood of an amino acid

Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

ProteinMPNN

Decoder: Transformer Model

- Autoregressive decomposition
- One amino acid at a time – Run decoder as many times as there are amino acids



For each amino acid:

- Input: Updated **edge and node embeddings** and **previously predicted sequence**
- Returns a categorical probability distribution from which the next amino acid is sampled

Dauparas, J. et al. Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv 2022.06.03.494563 (2022) doi:10.1101/2022.06.03.494563.

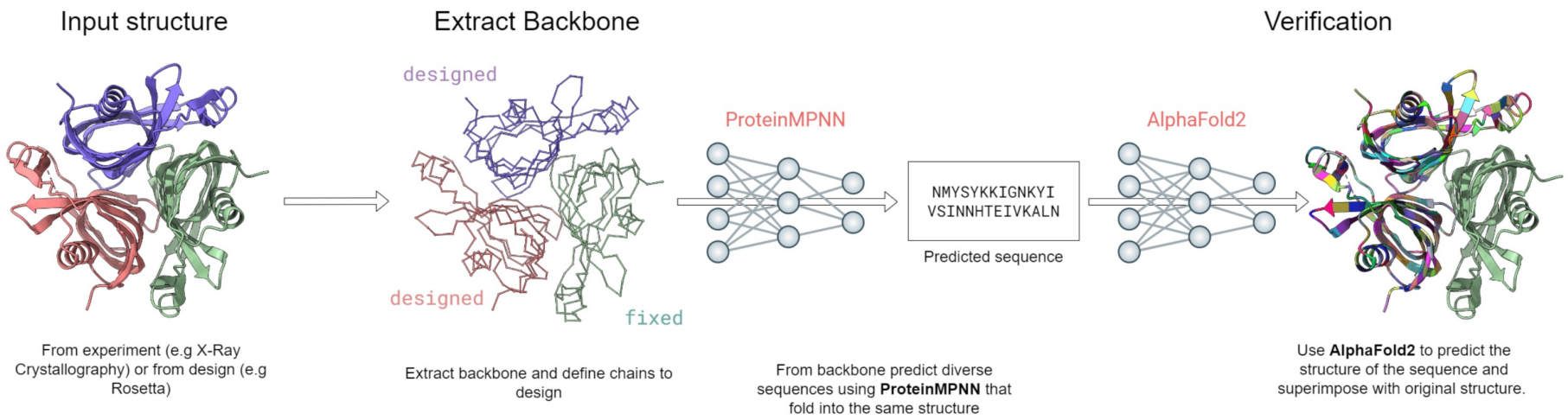
ProteinMPNN

Training Data: Proteins with known sequence and experimentally solved structure

- With removed amino acid side chains (only backbone, side chains to be rediscovered by MPNN)
- Loss = Categorical cross entropy per amino acid $CCE = -\sum_{i=1}^{20} p_i \log q_i$

Training Performance:

- Sequence recovery of ≈ 50% (Percentage of correct amino acids recovered from original sequence)
- Structure Validation – Do the predicted sequences fold into the right structure (same as input)?



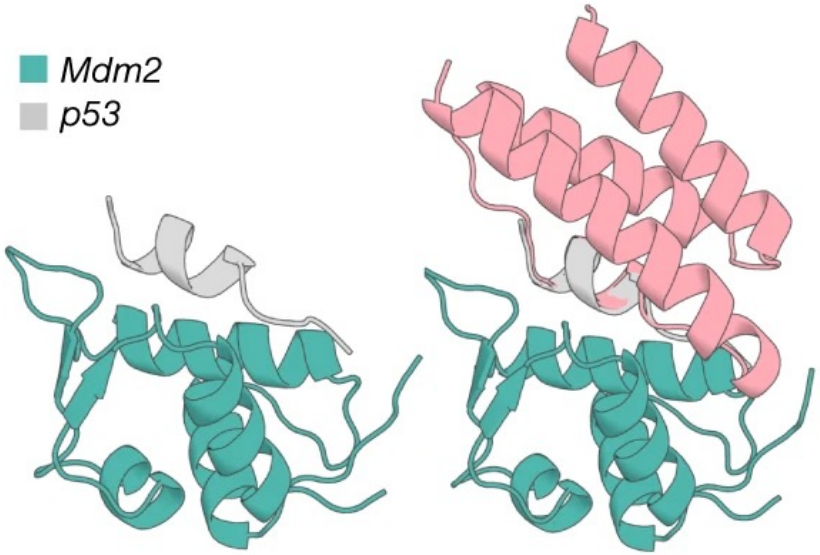
ProteinMPNN - Summary

ProteinMPNN solves sequence design problems, finding sequences that fold into specific backbone structures

- In silico and experimental validation show high accuracy
- Graph-based modelling of 3D backbone structures and MPNN-derived node features are crucial for model performance

Desired motifs can be integrated by keeping a part of the sequence fixed during sequence generation

- For design of MDM2-binding protein with the p53 helix, one could keep the residues of the helix fixed as they are in p53
- ProteinMPNN then designs the sequence around the fixed motif



Sequence design for backbones generated with RFdiffusion: Given backbone coordinates of a MDM2-binding protein generated by RFdiffusion, ProteinMPNN can find sequences that fold into the desired structures. Knowing that the original sequence of the p53-helix binds to MDM2, we can fix these residues during sequence generation with ProteinMPNN, and the model will design a sequence around the fixed amino acids.

Summary

→ **RFdiffusion** is a generative model for protein backbones that generates highly diverse protein backbones from random noise

→ **ProteinMPNN** is a generative model for sequence design, that finds sequences that fold into specific structures

The combination of RFdiffusion and ProteinMPNN is one potential approach for de novo protein structures design

Both models can be conditioned to include desired motifs/topologies, allowing to design proteins for specific applications.