By `Riddhiman Biswas`

Nobel Prize in Chemistry 2024

The Nobel Prize in Chemistry 2024 was divided, one half awarded to David Baker "for computational protein design", the other half jointly to Demis Hassabis and John Jumper "for protein structure prediction".

Introduction:

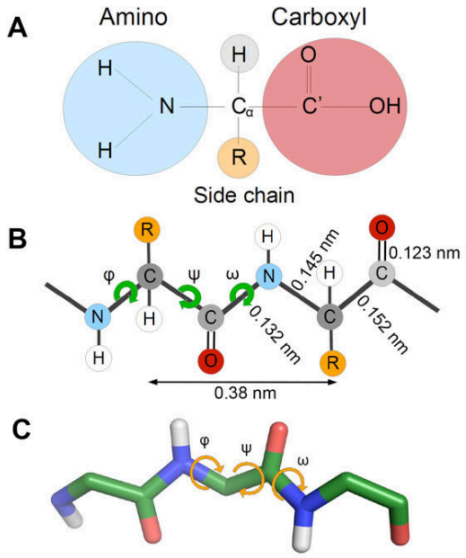
Sunsets with their dark red hue isn't just beautiful - it's biochemical. You effortlessly read this line, while noticing the colour at the same time. This process, called visual phototransduction, is made possible by several tiny molecules - the most prevalent being proteins, like rhodopsin. They convert the electromagnetic wave into electrical signals for your brain, which you then interpret as beautiful; or maybe not?

Proteins have a ubiquitous presence in our body, dealing with several different functions like signal transduction, DNA replication and repair, transportation - literally almost any process you could name. Crucial for the normal functioning of the protein is their 3D structure (crucial example being sickle-cell anaemia). Yet, for much of the last century, the process of how a protein finds its shape (and if we could predict the pathways) remained a mystery.

Detour to quick history of protein folding problem:

- 1949 : Frederick Sanger deciphers the first complete sequence of a protein - insulin. Turns out it's a linear chain of amino acids and not a random or branched chain [doi:10.1042/bj0450563].
- 1951 : Linus Pauling proposes the alpha-helix and beta sheet, held together by hydrogen bonds [doi:10.1073/pnas.37.5.235]. For the first time, protein folding has a geometrical backing.
- 1958 : John Kendrew reveals the 3D structure of myoglobin using X-ray diffraction [doi:10.1038/181662a0]. 5 years later, Max Perutz solved the structure of haemoglobin-IV [doi:10.1038/199633a0].
- 1961 : Christian Anfinsen takes a ribonuclease, denatures it, and watches it refold back to function perfectly - proving that sequence determines structure - the thermodynamic hypothesis [doi:10.1073/pnas.47.9.1309]. Nobel Prize in Chemistry 1972.
- 1969 : Meanwhile, Cyrus Levinthal poses a paradox [doi:10.1051/jcp/1968650044]: if a protein explored every possible shape randomly, it would take longer than the age of the universe to fold. And yet, it folds in milliseconds.

Recap of protein structure:



Protein structure

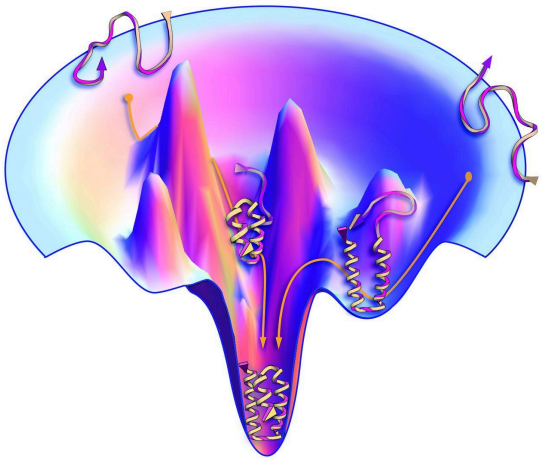
Here's a quick recap into the basic structure of a protein (you could skip if you are familiar)

- Proteins are made up of linear chains of amino acids, each connected by peptide bonds.
- Linear chain folds into the 3D structures (while going through secondary structures).
- Protein backbone is the structure formed by N (amide nitrogen) → Cα (alpha carbon) → C (carbonyl carbon) → next N.
- The 3D description of protein structure could be described using torsion angles, specifically, phi (ϕ) and psi (ψ) (remember Ramachandran plots?) - which describe the rotation around the bonds on either side of the central carbon, known as the alpha carbon (Cα).
- The amino acids also have side chains which extend out from this backbone, affecting the chemical nature of the amino acids and often interacting with the environment.

Background:

One of the first models for protein folding, the nucleation condensation theory, tried to explain the folding process by suggesting that some of the residues need to form contacts with each other for the folding process to proceed. This was very similar to the nucleation and growth mechanism in first order phase transitions - leading to borrowing terms from the field of condensed matter like the free energy landscape.

This eventually gave rise to the concept of a funneled energy landscape, where protein folding is understood not as a random search, but as a guided process through a landscape of decreasing energy - passing through multiple intermediate conformations (or structures) before reaching its stable, functional structure (there are certain proteins though, like serpins, whose biologically functional form is a kinetically trapped structure in one of these intermediate steps). There are also other hypotheses, like the zipping and assembly (ZA) where small fragments of the protein can search their conformations more completely than the larger parts.



Funnel Energy Landscape

Experimentally, tertiary protein structures are resolved by X-ray crystallography, nuclear magnetic resonance (NMR) and more recently, by cryo-electron microscopy (cryo-EM). The Protein Data Bank (PDB) database was started in 1971 to store the growing three-dimensional structure data of proteins determined by X-ray diffraction (primarily). However, these experimental approaches were very time consuming, often taking a PhD's worth of time.

As a result, computational approaches began to gain momentum. Around the 1980s, atomic force fields and Monte Carlo (MC) sampling were used to study the structure of proteins. Molecular Dynamics (MD), with increasingly sophisticated algorithms to sample the free energy surface more accurately, were being developed.

In the 1990s, researchers began using the similarity between sequences from proteins which carry out similar tasks in different organisms, called homology modeling, to predict unknown structures based on known ones. This allowed for the inference of structure when related proteins had already been experimentally resolved.

In 1994, John Moult started CASP (Critical Assessment of Techniques for Protein Structure Prediction). CASP is a biennial, communitywide blind test to predict the unknown structures of protein. This was important to objectively assess the quality of predictions made by various computational approaches. The prediction methods are grouped into three categories - ab-initio approaches based on the thermodynamics hypothesis, threading/fold recognition relying on the idea that structures are more conserved than sequences, and finally homology methods.

David Baker's contributions:

By the 1990s, many groups were experimenting with various force fields and homology modeling. However, de novo structure prediction or predicting the protein structure solely from the sequence became highly successful after Baker lab introduced the Rosetta algorithm. While it used the known structure database for understanding the structural patterns, Rosetta did not rely on direct similarity to any specific sequence or template, making it a true ab-initio approach.

Rosetta tries to predict how a protein folds by building it piece by piece. Short stretches of amino acid sequences tend to form specific local structures - however these local structures can exist in various conformations for different proteins. These local structures then interact nonlocally to form the final conformation. Rosetta forms a fragment library of structures from the protein structure database that these short sequences are known to form.

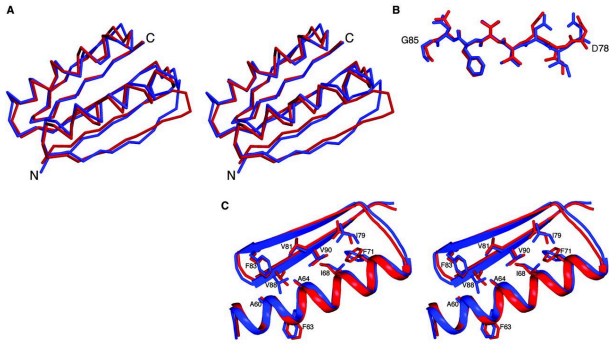
In the fragment assembly approach that it employs, Rosetta randomly selects a 9-residue (a residue is formed from an amino acid) fragment window. From the fragment library, it randomly chooses from the top 25 fragments ranked for this window. The program then changes the torsion angles in the protein chain, checks whether the change made it (i.e, lower energy or more stable), and keeps it if it did. If not, it accepts the change with a certain probability, to avoid getting stuck in a bad conformation (the Metropolis scheme as astute readers might have recognised). However, if no change is accepted for 150 moves, then it increases the probability till a move is accepted, after which the probability is reduced back to initial values.

Interestingly, even though Rosetta keeps track of angles in the torsion angle space, when calculating how good the structure is, it switches back to real 3D space (Cartesian space). In order to decide how good a structure is, Rosetta uses a scoring function based on physical assumptions. The function has several terms, like solvation, electrostatic terms, penalization of steric overlap of backbone and side chains, van der Waals interactions among others. In essence, the scoring function has two main parts - one looking into whether the structure makes sense physically, the other to check the likelihood of the particular structure based on what we know from real proteins.

Rosetta emerged as the leading de novo method, with foundations being laid for many extensions like loop modeling, protein-protein docking and ultimately de novo protein design.

And that's exactly what Baker did next.

In 2003, Baker's group designed a brand-new protein for the first time - one that never existed in nature, and yet folded reliably in the lab - using similar approaches to Rosetta. The result was Top7, a 93-residue α/β protein with a novel topology and an experimental structure that deviated by just 1.2 Å RMSD from the computational model.



Top7 structure - computational vs experimentally determined

Baker launched Rosetta@home, a distributed computing project that lets volunteers donate spare computing power from their personal devices to run protein folding simulations. In 2004, Baker's team launched Robetta, an online server that automated the process of structure prediction based on Rosetta. He also launched Foldit in 2008 - a multiplayer online game where non-experts fold proteins via an interactive interface. Players discovered novel folding strategies that sometimes outperformed the algorithm itself. They argued that human reasoning could help identify strategies to pursue suboptimal conformations instead of using stochastic (or random) search.

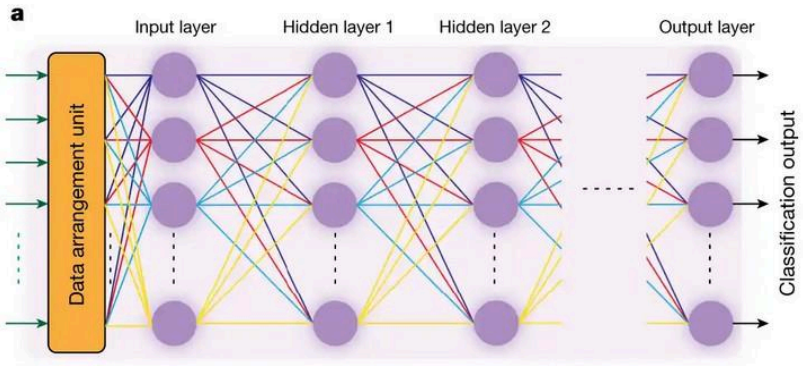
Some intermediate developments:

Baker's Rosetta had proved that physics-based modeling and statistics could help understand how proteins fold. However, ab-initio methods struggled with the enormous complexity of the protein folding process.

By the 2010s, with the growing database of protein sequences and structures - new methods were being actively developed. One of the critical aspects was the use of MSAs (multiple sequence alignments) to understand the evolutionary constraints that shape protein structure. MSAs are essentially the result of aligning the sequences of various biological molecules (including DNA, RNA or as is relevant here, proteins). These alignments capture evolutionary relationships by highlighting residue positions that co-evolve - mutations occurring at one site are correlated with mutations at another site which may not be close in the linear chain, suggesting spatial proximity in 3D structure plays an important role. MSAs could be traced back to 1994 paper by Göbel, where they show how correlated mutations derived from MSAs could be used to predict residue contacts, hinting at the role of evolutionary conservation in folding (co-evolutionary approach). Marks et al in 2011 used direct coupling analysis (a maximum entropy model), a statistical method enabling accurate residue contact prediction separating the direct correlations in mutations from the indirect correlations and noise in the MSA data.

In CASP XII (2016), there was an increase in precision for long-range contacts (more accurately, those residues separated by 24 or more positions) from 27% to 47% - a two-fold improvement. This was primarily driven by the MSA and new co-evolutionary based approaches. However, template based or homology modelling still remained the most accurate. Interestingly, there were some deep learning approaches as well (RaptorX-Contact by Xu et al - using deep convolutional residual neural network or ResNet).

Introduction to neural networks:



Representative Neural Network

Before delving into the complex architecture of AlphaFold2, first we dive a bit deeper into the world of deep learning. At the core of deep learning lies the neural network - a computational model loosely inspired by the connectivity seen in mammalian brains (including humans of course).

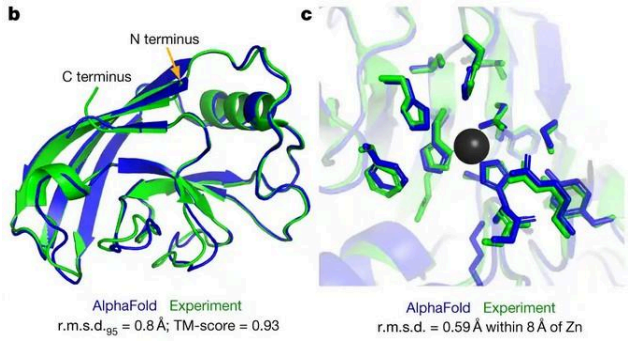
A neural network consists of layers of nodes or neurons, each of which performs a simple mathematical operation. It takes input values, multiplies them by adjustable parameter called weights, adds a bias (works similar to an intercept in a straight line equation) and then passes the result through a non-linear activation function (could be as simple as ReLU or Rectified Linear Unit - ReLU(x) = max(0,x)).

These layers are stacked on top of each other, with the output of one serving as the input for the next layer. During the training process, the network is shown many examples (in our case, protein sequences with known structures from the PDB) and it adjusts its weights (using backpropagation - using basically simple partial derivatives and chain rule to calculate the error propagation).

As the depth of these networks increased, they came to be known as deep neural networks, capable of learning increasingly abstract features - from simple patterns in early layers to highly complex relationships in deeper ones. However the deep neural networks suffered problems related to vanishing gradients (esp when the input becomes too weak to update the earlier layers) - leading to the introduction of ResNets (Residual Networks, the same as you might have noticed above). They introduced shortcut/residual connections - something which allows the output of one layer to skip one or more next layers, and get directly added to a later output - essentially skipping the layers which don't improve the result, increasing speed. They are great for image recognition and other related problems.

However, sequential data esp. like language, DNA and in our case protein sequences, we need a different architecture, instead of traditional models which process sequences one step at a time. In a landmark paper, 'Attention is all you need', Transformer architecture was introduced. This used a new mechanism called self-attention, where all parts of the input sequence are looked at simultaneously - essentially help model the complex, non-local interactions - something very crucial in protein folding (where two residues way many positions apart could interact - prominent example being a disulphide linkage). Additionally, they use positional encoding to keep track of the order of elements (will pop up later).

Deep Learning and AlphaFold2:



AlphaFold2 vs Experiment

DeepMind was founded by Demis Hassabis, Shane Legg and Mustafa Suleyman in 2010, using deep learning to initially teach artificial intelligence how to play old games from the seventies and eighties. In 2016, AlphaGo (an artificial neural network from DeepMind) beat Lee Sedol in a game of Go, becoming the first computer Go program to have beaten a 9-dan professional without handicap. After this, DeepMind turned its resources to the problem of protein folding.

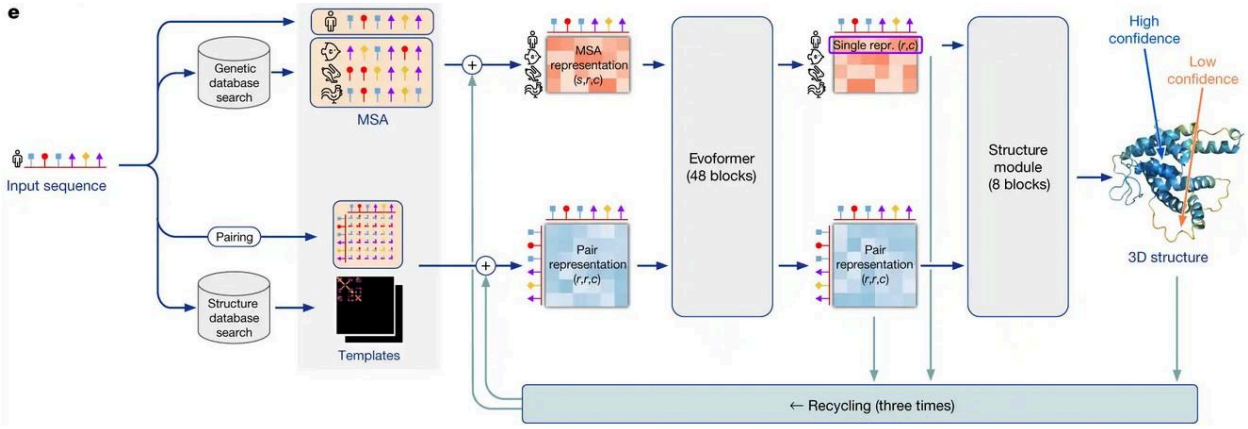
In CASP XIII (2018), DeepMind submitted results from AlphaFold - a deep learning neural network (more accurately again a ResNet; for interested readers, RaptorX-Contact just used mean and variance of the residue distribution, while AlphaFold used the entire distribution, allowing better potentials) - topping the free modelling (FM) category (one that requires ab-initio methods, no close templates available). It achieved a TM-score (a measure of similarity between protein structures) of ~0.70 - way higher than competing methods which hovered in the 0.40-0.60 range. However, it still didn't quite outperform in the template-based modelling (or one that used homology primarily), where Baker's RosettaCM still held quite strong. However, CASP XIII reported general use of deep learning techniques which had

significantly improved the residue contact prediction between pairs of residues, improving almost every structure prediction.

In CASP XIV (2020), DeepMind competed with a newer version, AlphaFold2, achieving a median GDT_TS score of 92.4 (another important measure - computed over the alpha carbon atoms of the protein) across all targets - outperforming almost every other method by a huge margin. It achieved a median backbone accuracy of 0.96 Å RMSD (for comparison, the width of a carbon atom is 1.4 Å). Directly quoting the CASP report - "...the results represent a solution to the classical protein-folding problem, at least for single proteins". However, it's important to realize that many groups had already reached the level of AlphaFold1 by then, even developing web servers and extending to a wider range of tasks like docking.

Before proceeding deeper into the network architecture of AlphaFold2, as the authors of CASP XIV have also mentioned, AlphaFold2 doesn't completely solve the classical problem of protein folding. Essentially, the prediction should be possible without any evolutionary information feeded. There is also an uncomfortable point - we don't really know the physics of the process since we don't really know what the machine is doing.

Architecture of AlphaFold2:



AlphaFold2 architecture

AlphaFold2 is trained on a hybrid dataset - 25% from the already existing structures in PDB and the rest (75%, yeah easy to calculate) from self-distilled data. Self-distillation is when it uses its own predicted structures as training data. AlphaFold2 uses the undistilled model (which has been just trained on PDB data only), and predicts the 3D structures of a large set of protein sequences for which no experimental structures are available. It then creates a distribution of pairwise distance between residues (or amino acids), compares it with a reference distribution (using KL divergence, for the interested) and gets a confidence score on how good the structure is. The self-distilled data is then ready after having ignored the residues with low confidence.

This hybrid data set is looped over while applying certain operations every time to promote diversity. The operations include quality filtering, MSA block deletion, MSA clustering and residual cropping. Let's dive deep into these operations (or you could skip it for the first read)

- Quality filtering - Only high quality 3D protein structures with resolution better than 9 Å are used. Chains are then filtered with probability (1/512) * max(min(N_res,512),256), essentially making the network train on longer chains often and sample fewer protein from a cluster of similar proteins
- MSA block deletion - The MSA is a N_sequence × N_residue matrix (setting terminology for the sake of convenience). Contiguous blocks of the MSA sequences (groups of residues next to each other) are randomly deleted (why so? Similar sequences appearing next to each other tend to be conserved)
- MSA clustering - The computational cost increases as N_sequence^2 × N_residue (accurately, for the Evoformer - more on that later). Thus, a random subset of sequences are chosen (and without replacement, probability people) from the original input MSA, with the first sequence set to the protein AlphaFold is trying to predict. Mask is then generated (basically some residues are hidden on purpose to allow the model to predict) with a probability of 15% for each position in the MSA cluster center (basically the now randomly chosen subset). The elements in the mask are then replaced (or not replaced) according to fixed probabilities. Astute readers should however claim that choosing a special subset biases the result. To tackle this problem, the remaining sequences are assigned to the closest cluster center (the sequence in the now randomly chosen subset) (using Hamming distance, essentially a measure of how similar the sequences are). Then, summary statistics are calculated for each cluster (frequency of each amino acid at each position and other such info). N_extra_sequence such sequences are also chosen for extra MSA features (later used in input embeddings)
- Residual cropping - The full sequence is not used in the training as well (N_residue is essentially reduced). The sequence is cropped to a fixed size in two modes - one with fixed size, the other with a degree of randomness on top of fixed size.

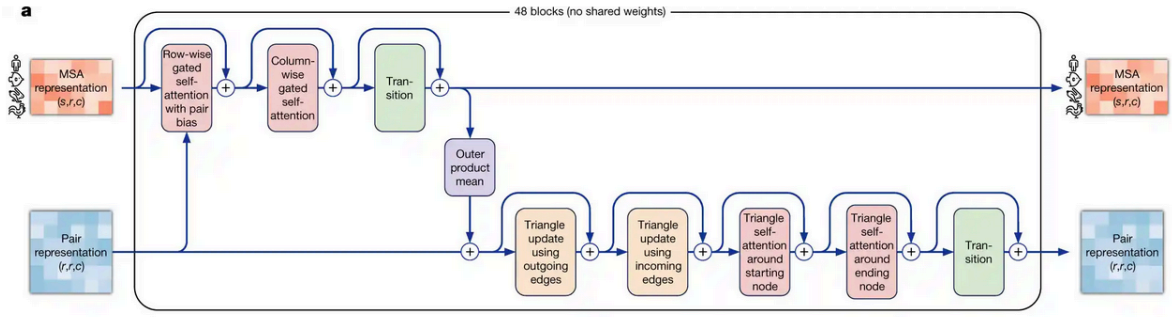
Now, we come to the next part of the architecture - input embeddings. Raw input data is now converted into numerical representations which contain information about the amino acid sequence, MSA features, and the extra MSA features as well. This is taken in to output MSA representation and pair representation of residues (size N_residue X N_residue)

Tha MSA representation is created by linearly transforming the MSA sequence features at each residue position and summing it with the linear transformation of the target sequence's features. The pair representation creation is a bit more involved - describing features and positions as vectors and then some linear transformation on them. However, here two important tricks are used - one-hot encoding and relative positional encoding. One-hot encoding basically converts the data into a binary vector where only one element is 1 ('hot' element) and the rest are 0's. For example, you could have a vector with length 20 (each position representing amino acid) and could represent each amino acid by putting 1 in a specific position. On the other hand, relative positional encoding encodes relative positions of residues instead of absolute numbering with distance ranging from -32 to 32, preventing residues at large distances from dominating encoding.

Trunk of the architecture:

Next we come to main important parts of the architecture - the Evoformer (the Transformer part) and the Structure Module.

The Evoformer is the core neural network part. It takes in both the MSA and pair representations and iteratively updates both of them through the 48 blocks or layers. Each block takes the MSA and pair representations as input and outputs updated versions, which serves as input for next layers via the residual connections we spoke about before.

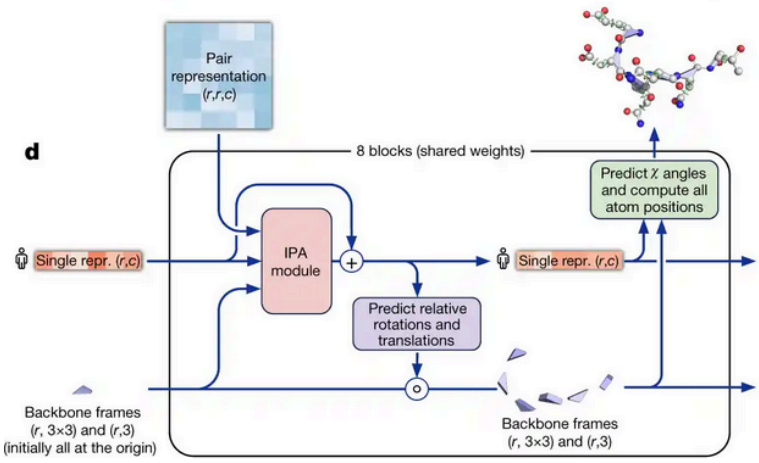


Evoformer architecture

Each Evoformer block has several components

- MSA row-wise gated self-attention with pair bias - Attention applied along each row of the MSA (across all sequences but at the same residue position). This allows MSA to talk across sequences while considering pairwise residue info, combining evolutionary information with geometric information.
- MSA column-wise gated self-attention - Attention applied column-wise (across residues but within the same sequence). This allows to infer about how the residues influence each other, while simultaneously capturing long range interactions as mentioned before
- MSA transition - 2-layer MLP (multi-layer perceptron, a type of deep neural network). This helps to better understand the non-linear influences.
- Outer product mean - Connects MSA information (evolutionary info) into pairwise residue information (spatial info). It's based on the fact that residue that co-evolve tend to be close in 3D (as mentioned earlier)
- Triangular multiplicative update - Updates pair representation based on triangles formed by i, j, and k residues. This helps capture complex residue interactions using some update algorithm, while encoding geometry of residue triplets like bond angles and distances (especially the triangle inequality of distances)
- Triangular self-attention - another pair representation update. This reinforces geometrical understanding
- Transition in the pair stack - similar 2-layer MLP as earlier

The Structure module then takes the learned MSA and pair representations and converts it into 3D atomic coordinates. The protein backbone is represented as a series of local coordinate frames, one for each residue. Here's an interesting part - all the frames start at the origin called the 'black hole initialization'. This means all residues overlap at the same point - a highly unphysical situation but this simplifies the learning process.



Structure Module

The Structure module has 8 layers with each layer doing the following

- Invariant Point Attention (IPA) - It's a specialized attention mechanism that is invariant to global rotations and translations. This is based on the fact that the structure should remain the same even if you rotate it or translate (move) it along some direction.
- Transition layer - a small MLP again
- Frame composition - From the pair representation, the network predicts a small frame update (rotation and translation per residue). These are added on top of existing frames to evolve the structure iteratively. Similar to Baker, they represent the atomic position via torsion angles, using a shallow ResNet to predict them. Interestingly, rotations use quaternions (vectors with 4 components along with certain special properties) with the first component being fixed to 1. The rest three are predicted by the network and are used to define the Euler axis of rotation (using quaternions than rotation matrices uses less components for rotations, has numerical stability and other advantages)

After all 8 layers, the module outputs the final backbone frames for each residue, predicted torsion angles for side chains and full atomic coordinates for all heavy atoms (everything other than H). It also predicts a confidence measure, pLDDT score which estimates the local accuracy.

However, in spite of all these, there still are physical violations like steric clashes. In order to remove those energy minimization is done using AMBER99SB force field (yes, the force field from the beginning) with restraints applied to heavy atoms to allow the structure to relax near the input structure. If the residues still have violations, restraints are removed and the entire procedure is repeated till there are no physical violations.

And voila! You have the structure ready!

Future Directions:

While clearly AlphaFold2 has achieved a historic feat in structural biology - a lot remains to be done.

In 2023, AlphaFold3 was introduced by DeepMind and Isomorphic Labs, expanding beyond just protein structures to predict the joint structure of complexes including proteins, nucleic acids, small molecules, ions and modified residues.. Unlike AlphaFold2, AlphaFold3 uses a diffusion-based generative model.

Meanwhile, RosettaFold, developed by the Baker Lab, takes a different route. Its newer versions like RosettaFoldNA and RFdiffusion explore multi-chain complexes, nucleic acid interactions, and even de novo protein design. This allows us not just to predict structures, but to come up with new ones that nature hasn't made yet.

These models however still face challenges - dynamics, post-translational modifications, membrane interactions, and the physics of folding remain imperfectly captured.

About Laureates:

David Baker:

Born in 1962, David Baker is a professor of biochemistry at the University of Washington and director of the Institute for Protein Design. He is also an investigator with the Howard Hughes Medical Institute (HHMI). He primarily works on the developments of new computational methods for protein structure prediction and even designing them, focusing on the functional aspects. He also has an experimental biochemistry lab and has authored several highly cited papers.

Primarily known for developing the Rosetta algorithm for ab-initio protein structure prediction, his lab has also developed a computing project Rosetta@home, computer game Foldit and RosettaFold. His lab was the first to design artificial protein with novel fold, Top7. He has been awarded numerous prizes including Breakthrough Prize (2021), Feynman Prize in Nanotechnology (2004), Newcomb Cleveland Prize (2004).

He has been awarded one half of the 2024 Nobel Prize in Chemistry for his work on computational protein design.

Demis Hassabis:

Born in 1976, Demis Hassabis is an AI researcher, and entrepreneur, currently serving as CEO and co-founder of DeepMind, an AI company acquired by Google in 2015. He is also the founder of Isomorphic Labs under Alphabet Inc. (parent company of Google) and is UK government AI advisor.

Hassabis led the development of AlphaGo, the first AI to defeat a human Go champion, and later AlphaFold, which made significant progress in the field of protein folding. For his contributions, he has been elected as a fellow to the Royal Society and won many prestigious awards including the Breakthrough Prize (2023), Lasker Award among others. In 2024, he was knighted and has been in the Time 100 list twice (2017,2025) featuring on cover in 2025 as well.

John Jumper:

Born in 1985, John Jumper is a computer scientist and computational chemist. He currently works at Google DeepMind as their director. He has a PhD in theoretical chemistry from the University of Chicago in 2017 with thesis being ‘New methods using rigorous machine learning for coarse-grained protein folding and dynamics’.

He made important contributions to the development of AlphaFold and AlphaFold2. In 2021, he was included in the Nature’s 10 list. For his contributions, he was jointly awarded the Breakthrough Prize (2023), Lasker Award, along with Demis Hassabis, among others awards.

Demis Hassabis and John Jumper have been jointly awarded the other half of the 2024 Nobel Prize in Chemistry for his work on protein structure prediction.

References:

On Protein Folding History and other reviews -

- Chen, Y. et al. (2007) 'Protein folding: Then and now,' Archives of Biochemistry and Biophysics, 469(1), pp. 4–19. https://doi.org/10.1016/j.abb.2007.05.014.
- Dill, K.A. et al. (2008) 'The protein folding problem,' Annual Review of Biophysics, 37(1), pp. 289–316. https://doi.org/10.1146/annurev.biophys.37.092707.153558.
- Zhang, Y. (2008) 'Progress and challenges in protein structure prediction,' Current Opinion in Structural Biology, 18(3), pp. 342–348. https://doi.org/10.1016/j.sbi.2008.02.004.
- Dill, K.A. and MacCallum, J.L. (2012) 'The Protein-Folding Problem, 50 years on,' Science, 338(6110), pp. 1042–1046. https://doi.org/10.1126/science.1219021.
- Englander, S.W. and Mayne, L. (2014) 'The nature of protein folding pathways,' Proceedings of the National Academy of Sciences, 111(45), pp. 15873–15880. https://doi.org/10.1073/pnas.1411798111.
- Nassar, R. et al. (2021) 'The protein folding Problem: The role of Theory,' Journal of Molecular Biology, 433(20), p. 167126. https://doi.org/10.1016/j.jmb.2021.167126.
- Brini, E., Simmerling, C. and Dill, K. (2020) 'Protein storytelling through physics,' Science, 370(6520). https://doi.org/10.1126/science.aaz3041.

CASP papers (from relevant years) **-**

- Moult, J. et al. (2017b) 'Critical assessment of methods of protein structure prediction (CASP)—Round XII,' Proteins Structure Function and Bioinformatics, 86(S1), pp. 7–15. https://doi.org/10.1002/prot.25415.
- Kryshtafovych, A. et al. (2019b) 'Critical assessment of methods of protein structure prediction (CASP)—Round XIII,' Proteins Structure Function and Bioinformatics, 87(12), pp. 1011–1020. https://doi.org/10.1002/prot.25823.
- Kryshtafovych, A. et al. (2021) 'Critical assessment of methods of protein structure prediction (CASP)—Round XIV,' Proteins Structure Function and Bioinformatics, 89(12), pp. 1607–1617. https://doi.org/10.1002/prot.26237.

Baker Lab papers -

- Rohl, C.A. et al. (2004) 'Protein structure prediction using Rosetta,' Methods in Enzymology on CD-ROM/Methods in Enzymology, pp. 66–93. https://doi.org/10.1016/s0076-6879(04)83004-0.
- Kuhlman, B. et al. (2003) 'Design of a Novel Globular Protein Fold with Atomic-Level Accuracy,' Science, 302(5649), pp. 1364–1368. https://doi.org/10.1126/science.1089427.
- Kim, D.E., Chivian, D. and Baker, D. (2004) 'Protein structure prediction and analysis using the Robetta server,' Nucleic Acids Research, 32(Web Server), pp. W526–W531. https://doi.org/10.1093/nar/gkh468.
- Cooper, S. et al. (2010) 'Predicting protein structures with a multiplayer online game,' Nature, 466(7307), pp. 756–760. https://doi.org/10.1038/nature09304.
- Baek, M. et al. (2021) 'Accurate prediction of protein structures and interactions using a three-track neural network,' Science, 373(6557), pp. 871–876. https://doi.org/10.1126/science.abj8754.

AlphaFold -

- Jumper, J. et al. (2021) 'Highly accurate protein structure prediction with AlphaFold,' Nature, 596(7873), pp. 583–589. https://doi.org/10.1038/s41586-021-03819-2.
- Abramson, J. et al. (2024) 'Accurate structure prediction of biomolecular interactions with AlphaFold 3,' Nature, 630(8016), pp. 493–500. https://doi.org/10.1038/s41586-024-07487-w.

Future Directions -

- Perrakis, A. and Sixma, T.K. (2021) 'AI revolutions in biology,' EMBO Reports, 22(11). https://doi.org/10.15252/embr.202154046.
- Moore, P.B. et al. (2022) 'The protein-folding problem: Not yet solved,' Science, 375(6580), p. 507. https://doi.org/10.1126/science.abn9422.

Others -

- Marks, D.S. et al. (2011) 'Protein 3D Structure Computed from Evolutionary Sequence Variation,' PLoS ONE, 6(12), p. e28766. https://doi.org/10.1371/journal.pone.0028766.
- De Juan, D., Pazos, F. and Valencia, A. (2013) 'Emerging methods in protein co-evolution,' Nature Reviews Genetics, 14(4), pp. 249–261. https://doi.org/10.1038/nrg3414.
- Wang, S., Sun, S. and Xu, J. (2017) 'Analysis of deep learning methods for blind protein contact prediction in CASP12,' Proteins Structure Function and Bioinformatics, 86(S1), pp. 67–77. https://doi.org/10.1002/prot.25377.
- Göbel, U. et al. (1994) 'Correlated mutations and residue contacts in proteins,' Proteins Structure Function and Bioinformatics, 18(4), pp. 309–317. https://doi.org/10.1002/prot.340180402.
- W, A., Senior et al. (2020) 'Improved protein structure prediction using potentials from deep learning,' Nature, 577(7792), pp. 706–710. https://doi.org/10.1038/s41586-019-1923-7.
- Vaswani, A. et al. (2017) Attention is all you need. https://arxiv.org/abs/1706.03762.
- Castorina, L. (2025) How to solve the protein folding problem: AlphaFold2. https://towardsdatascience.com/how-to-solve-the-protein-folding-problem-alphafold2-6c81faba670d.
- How does DeepMind AlphaFold2 work? (2021). https://borisburkov.net/2021-12-25-1.
- Fuchs, F. (no date) Fabian Fuchs. https://fabianfuchsml.github.io/alphafold2.