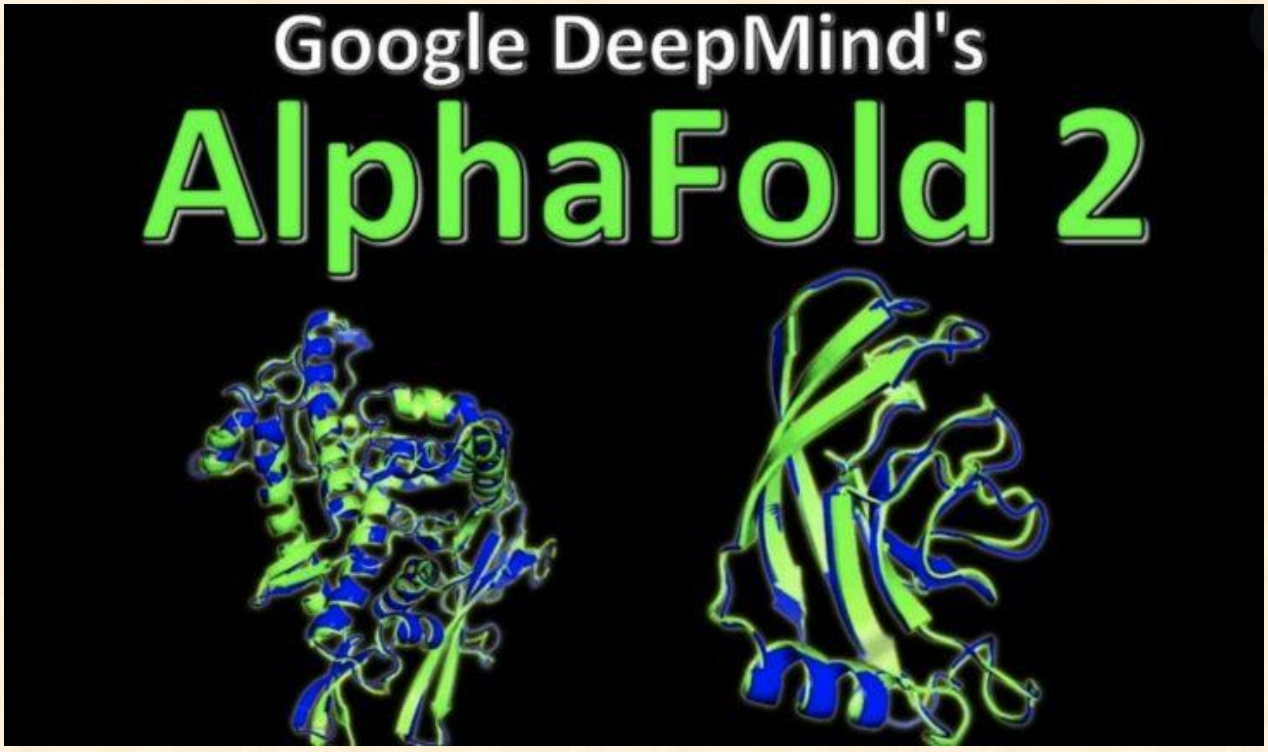
Artificial Intelligence Solving Problems with Huge Numbers of Possibilities

by Brian Wang / NextBigFuture.com / December 6, 2020



'Deep Mind' has been able to make significant progress toward solving the protein folding problem. Protein folding is not solved yet.

Here I will describe :

- the complexity the protein folding problem. 10^300 folding possibilities for the average protein based upon certain assumptions
- I will compare the other large problems to protein folding and some discussion about comprehending large numbers

Complexity of Protein Folding

The scale of the protein folding problem was fairly well defined and understood in 1969. Levinthal explained it:

==========

Proteins are macromolecules which possess several unique properties. They are very large (containing 2,000-or-more atoms) and complex. Their structures show no obvious regularity but a very subtle regularity is apparent upon close examination.

We know from the fact that proteins may be crystallized and further from x-ray crystallography that each atom occupies a unique place in the relative 3-dimensional space of the molecule. If we consider a protein containing 2,000 atoms with no structural restrictions, such a macromolecule would possess 6,000-degrees of-freedom. We know, however, from x-ray studies and other techniques as well, that there are indeed certain structural restrictions in a polypeptide structure.

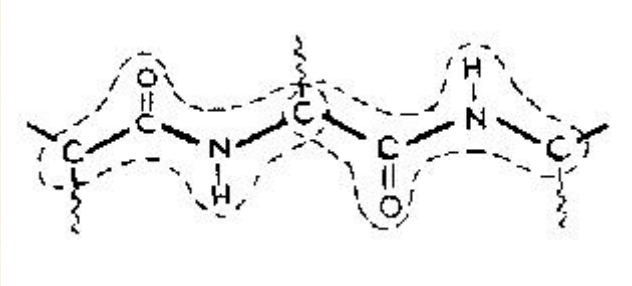


Figure 1

For example, if we schematically indicate a polypeptide chain as in Figure 1, we find that the 6 atoms in each unit indicated by the dotted lines lie in a common plane. Considerations of such factors allow us to predict only 450 degrees-of-freedom in a protein structure containing 150 amino acids, for example. Of these 450 degrees-of-freedom, 300 would be due to rotations and 150 would be due to relative bond angles of the side chains.

There was earlier (1960s work) attempting to predict the 3-dimensional structure of some polypeptides from primary sequence information.

If we begin with a set of bond angles and bond lengths and go to 3-dimensional coordinates (via vector matrix multiplications), we can build a 3-dimensional image and display it on a computer-controlled oscilloscope. If we know the coordinates of any two atoms and their interaction energy functions, could we extend this treatment to sum the total energy of a given polypeptide or protein structure?

How accurately must we know the bond angles to be able to estimate these energies? Even if we knew these angles to better than a tenth of a radian, there would be 10^{300} possible configurations in our theoretical protein.

In nature, proteins apparently do not sample all of these possible configurations since they fold in a few seconds. Even postulating a minimum time for going from one

conformation to another, the proteins would have time to try on the order of 10^8 different conformations at most before reaching their final state.

We feel that protein folding is speeded and guided by the rapid formation of local interactions which then determine the further folding of the peptide. This suggests local amino acid sequences which form stable interactions and serve as nucleation points in the folding process.

Then, is the final conformation necessarily the one of lowest free energy? We do not feel that it has to be. It obviously must be a metastable state which is in a sufficiently deep energy well to survive possible perturbations in a biological system. If it is the lowest energy state, we feel it must be the result of biological evolution (i.e., the first deep metastable trough reached during evolution happened to be the lowest energy state).

You may then ask the question *"Is a unique folding necessary for any random 150-amino acid sequence?"* I would answer "Probably not." Some experimental support for this statement comes from the difficulty many of us are all too aware of in attempting to crystallize peptides.

I would like to illustrate some of these points by telling you about some work we have done on an alkaline phosphatase enzyme. This enzyme has a molecular weight of 40,000 and consists of two similar or identical subunits. We have unfolded this enzyme and then followed the rate of refolding or renaturation under appropriate conditions as a function of temperature. As can be seen in the figure below, the optimum rate of renaturation occurs at 37°C and falls rapidly at higher and lower temperatures.

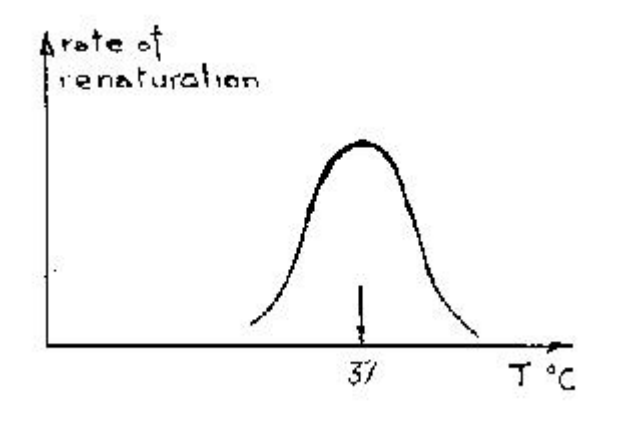


Figure 2

The organism which produces this enzyme grows optimally at 37°C also. Although the renaturation rate drops off above 37°C, the native intact enzyme (or the refolded enzyme) is stable up to 90°C. Thus, once the folding is complete, the resulting structure is quite stable.

We have isolated mutants of this organism which produce active enzyme only when grown at temperatures below 37°C and we have found that the protein renatures only at temperatures below 37°C as shown in the figure below. Once this enzyme is formed, however, it again is stable to 90°C. This behavior is obviously not expected in an equilibrium situation.

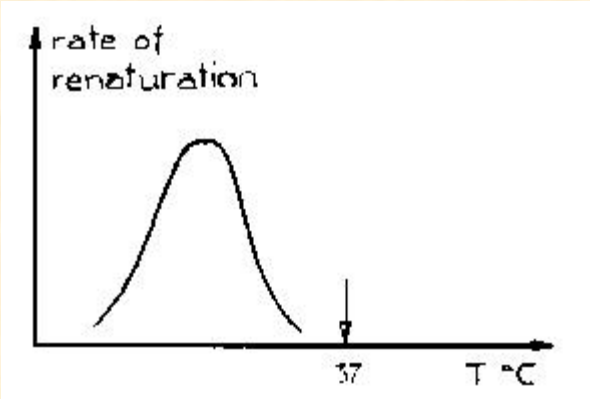


Figure 3

As it turns out upon closer study, the limiting rate in the formation of active enzyme is the formation of the dimeric species of the enzyme. We can, however, say that at least in the assembly of protein subunits, it matters in which order what events occur.

We may be helped ultimately by sufficient data from x-ray crystallographic work to find clues as to the kinds of local interactions which are most important in protein folding.

What then can we derive from computer calculations? We know very accurately:

1. bond lengths in polypeptides
2. planar groupings in the polypeptide structure.

For small molecules, it is possible to analyze x-ray diffraction data by means of the direct methods. For large molecules, this is generally beyond our ability at present and we must obtain phase information in order to reconstruct the reflected intensities. We hope to look for reflections from certain postulated substructures by having our computer search in Fourier space for such groupings and then refine these data by means of the tangent formula and then relate other intensities to these.

Professor Levinthal then showed a short motion picture which illustrated the synthesis of a polypeptide structure and the process of then forming a desired interaction via the most favored energy path as displayed on the computer-controlled oscilloscope. The relevance of these studies to Mossbauer spectroscopy may be in the understanding of small perturbations of polypeptide structures and their effect on the Mossbauer nucleus.

Discussion:

Q: Is a protein really ever truly unfolded (i.e., devoid of secondary and tertiary structure)?

A: Both physical measurements and synthetic polypeptide work suggest the answer is yes.

Q: The tangent formula requires phase information first in order to refine the data. Are you implying this is not the case?

A: Since we are looking for known substructures within the patterns, we can use the tangent formula.

Q: Have you used your method to produce a known structure and looked for the most likely thermal perturbation of the structure?

A: No, we haven't done calculations of that sort.

==========

Other Large Problems

Game Complexity has been analyzed for state space and game tree complexity.

The state-space complexity of a game is the number of legal game positions reachable from the initial position of the game. The game tree size is the total number of possible games that can be played (the number of leaf nodes in the game tree rooted at the game's initial position).

You would have to get past the third nested universe of atoms to get to 10^300.

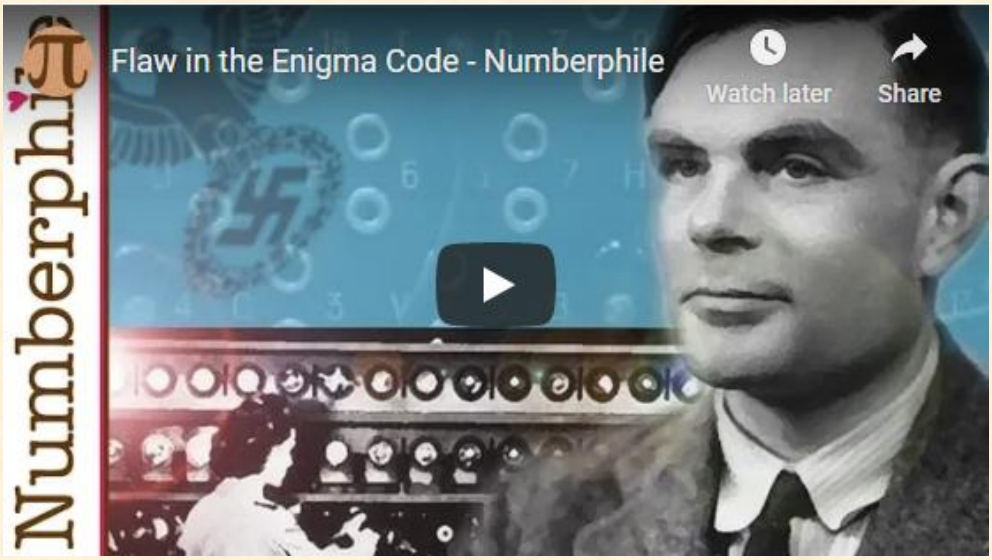
10^82*10^82*10^82*10^54 = 10^300

Game	Board size (positions)	State-space complexity (as log to base 10)	Game-tree complexity (as log to base 10)	Average game length (plies)	Branching factor	Ref	Complexity class of suitable generalized game
Tic-tac-toe	9	3	5	9	4		PSPACE-complete[5]
Sim	15	3	8	14	3.7		PSPACE-complete[6]
Pentominoes	64	12	18	10	75	[7][8]	?, but in PSPACE
Kalah [9]	14	13	18			[7]	Generalization is unclear
Connect Four	42	13	21	36	4	[1][10]	?, but in PSPACE
Domineering (8 × 8)	64	15	27	30	8	[7]	?, but in PSPACE; in P for certain dimensions[11]
Congkak	14	15	33			[7]	
English draughts (8x8) (checkers)	32	20 or 18	31	70	2.8	[1][12]	EXPTIME-complete[13]
Awari[14]	12	12	32	60	3.5	[1]	Generalization is unclear
Qubic	64	30	34	20	54.2	[1]	PSPACE-complete[5]
Double dummy bridge[nb 1]	(52)	<17	<40	52	5.6		PSPACE-complete[15]
Fanorona	45	21	46	44	11	[16]	?, but in EXPTIME
Nine men's morris	24	10	50	50	10	[1]	?, but in EXPTIME
Tablut	81	27				[17]	
International draughts (10x10)	50	30	54	90	4	[1]	EXPTIME-complete[13]
Chinese checkers (2 sets)	121	23				[18]	EXPTIME-complete [19]
Chinese checkers (6 sets)	121	78				[18]	EXPTIME-complete [19]
Reversi (Othello)	64	28	58	58	10	[1]	PSPACE-complete[20]
OnTop (2p base game)	72	88	62	31	23.77	[21]	
Lines of Action	64	23	64	44	29	[22]	?, but in EXPTIME
Gomoku (15x15, freestyle)	225	105	70	30	210	[1]	PSPACE-complete[5]
Hex (11x11)	121	57	98	50	96	[7]	PSPACE-complete[5]
Chess	64	47	123	70	35	[23]	EXPTIME-complete (without 50-move drawing rule)[24]
Bejeweled and Candy Crush (8x8)	64	<50				[25]	NP-hard
GIPF	37	25	132	90	29.3	[26]	
Connect6	361	172	140	30	46000	[27]	PSPACE-complete[28]
Backgammon	28	20	144	55	250	[29]	Generalization is unclear
Xiangqi	90	40	150	95	38	[1][30][31]	?, believed to be EXPTIME-complete
Abalone	61	25	154	87	60	[32][33]	PSPACE-hard, and in EXPTIME
Havannah	271	127	157	66	240	[7][34]	PSPACE-complete[35]
Twixt	572	140	159	60	452	[36]	
Janggi	90	44	160	100	40	[31]	?, believed to be EXPTIME-complete
Quoridor	81	42	162	91	60	[37]	?, but in PSPACE
Carcassonne (2p base game)	72	>40	195	71	55	[38]	Generalization is unclear
Amazons (10x10)	100	40	212	84	374 or 299[39]	[40][41]	PSPACE-complete[42]
Shogi	81	71	226	115	92	[30][43]	EXPTIME-complete[44]
Go (19x19)	361	170	360	150	250	[1][45][46]	EXPTIME-complete[47]
Arimaa	64	43	402	92	17281	[48][49][50]	?, but in EXPTIME
Stratego	92	115	535	381	21.739	[51]	
Infinite chess[nb 2]	infinite	infinite	infinite	infinite	infinite	[54]	Unknown, but mate-in-n is decidable[55]

Solving the Nazi Enigma Code

On July 9, 1941, Allied code breakers broke the Nazi enigma code. The Nazi enigma code machine had 159 quintillion settings (1.59 * 10^20).

Enigma could not encode the same letter back to itself.



Operator shortcomings of the use of Enigma.

Operating shortcomings greatly helped in breaking Engima broken.

- The production of an early Enigma training manual containing an example of plaintext and its genuine ciphertext together with the relevant message key. When Rejewski was given this in December 1932, it "made his reconstruction of the Enigma machine somewhat easier".

Repetition of the message key as described in Rejewski's characteristics method above. (This helped in Rejewski's solving Enigma's wiring in 1932 and was continued until May 1940.)

- Repeatedly using the same stereotypical expressions in messages, an early example of what Bletchley Park would later term cribs. Rejewski wrote that "… we relied on the fact that the greater number of messages began with the letters ANX -- German for "to" followed by X as a spacer".

- The use of easily guessed keys such as AAA or BBB or sequences that reflected the layout of the Enigma keyboard such as "three [typing] keys that stand next to each other or diagonally [from each other]…"[91] At Bletchley Park, such occurrences were called cillies. Cillies in the operation of the four-rotor Abwehr Enigma included four-letter names and German obscenities. Sometimes with multi-part messages, the operator would not enter a key for a subsequent part of a message, merely leaving the rotors as they were at the end of the previous part, to become the message key for the next part.

- Having only three different rotors for the three positions in the scrambler. (This continued until December 1938 when it was increased to five and then eight for naval traffic in 1940.)

- Using only six plugboard leads leaving 14 letters unsteckered. (This continued until January 1939 when the number of leads was increased, leaving only a small number of letters unsteckered.)

Other useful shortcomings that were discovered by the British and later the American cryptanalysts included the following, many of which depended on frequent solving of a particular network:

- The practice of re-transmitting a message in an identical or near-identical form on different cipher networks. If a message was transmitted using both a low-level cipher that Bletchley Park broke by hand and Enigma, the decrypt provided an excellent crib for Enigma decipherment.

- For machines where there was a choice of more rotors than there were slots for them, a rule on some networks stipulated that no rotor should be in the same slot in the scrambler as it had been for the immediately preceding configuration. This reduced the number of wheel orders that had to be tried.

- Not allowing a wheel order to be repeated on a monthly setting sheet. This meant that when the keys were being found on a regular basis, economies in excluding possible wheel orders could be made.

- The stipulation for Air Force operators that no letter should be connected on the plugboard to its neighbor in the alphabet. This reduced the problem of identifying the plugboard connections and was automated in some Bombes with a Consecutive Stecker Knock-Out (CSKO) device.

- The sloppy practice that John Herivel anticipated soon after his arrival at Bletchley Park in January 1940. He thought about the practical actions that an Enigma operator would have to make and the shortcuts that he might employ. He thought that after setting the alphabet rings to the prescribed setting and closing the lid, the operator might not turn the rotors by more than a few places in selecting the first part of the indicator. Initially this did not seem to be the case. But after the changes of May 1940, what became known as the Herivel tip proved to be most useful.

- The practice of re-using some of the columns of wheel orders, ring settings, or plugboard connections from previous months. The resulting analytical shortcut was christened at

Bletchley Park Parkerismus after Reg Parker who had through his meticulous record-keeping spotted this phenomenon.

- The re-use of a permutation in the German Air Force METEO code as the Enigma stecker permutation for the day.

Mavis Lever (a member of Dilly Knox's team) recalled an occasion when there was an extraordinary message.

> The one snag with Enigma of course is the fact that if you <press> A, you can get every other letter but A. I picked up this message and -- one was so used to looking at things and making instant decisions -- I thought: 'Something's gone. What has this chap done? There is not a single L in this message.'
>
> My chap had been told to send out a dummy message and he had just had a fag [cigarette] and <pressed> the last key on the keyboard (the L). So that was the only letter that didn't come out. We had got the biggest crib we ever had -- the encypherment was LLLL -- right through the message and that gave us the new wiring for the wheel [rotor]. That's the sort of thing we were trained to do. Instinctively look for something that had gone wrong or someone who had done something silly and torn up the rule book

Deep Mind Alphafold Approach Towards Solving Protein Folding

In 2019, the Deep Mind Alphafold approach was summarized. Alphafold 2 was an updated version that got even better results.

Deep learning was just one aspect of the structure prediction process and the final structures were actually a result of gradient descent optimization. The DeepMind team tried a 'fancier' strategy involving fragment assembly using Generative Adversarial Networks (GANs). Bbut in the end, the best results were obtained by gradient descent optimization.

Method 1: Fragment Assembly

The overall protein structure of a protein is a combination of smaller fragmented units of structure. These sub-units are somewhat modular and form motifs that are re-used with slight modifications across different proteins and protein families. This is a major reason for the use of multiple sequence alignment as part of most protein structure prediction models. By comparing a novel protein sequence to a database of sequences that have known structures, an estimate of the sub-structure in the new protein can be inferred by taking the structures formed in proteins with similar sequences as templates.

Notably while DeepMind did use multiple sequence alignment in their approach, they did not use any templates. That is while they did compare the sequences to databases of known protein fragment sequences, they didn't borrow structures associated with those sequences directly. They used a generative neural network to come up with fragment candidates for insertion into an otherwise more conventional structure optimization workflow using simulated annealing.

Although GANs have matured substantially in the past few years with impressive results on tasks like de novo image generation, in this case, the results were not state of the art. For that, DeepMind

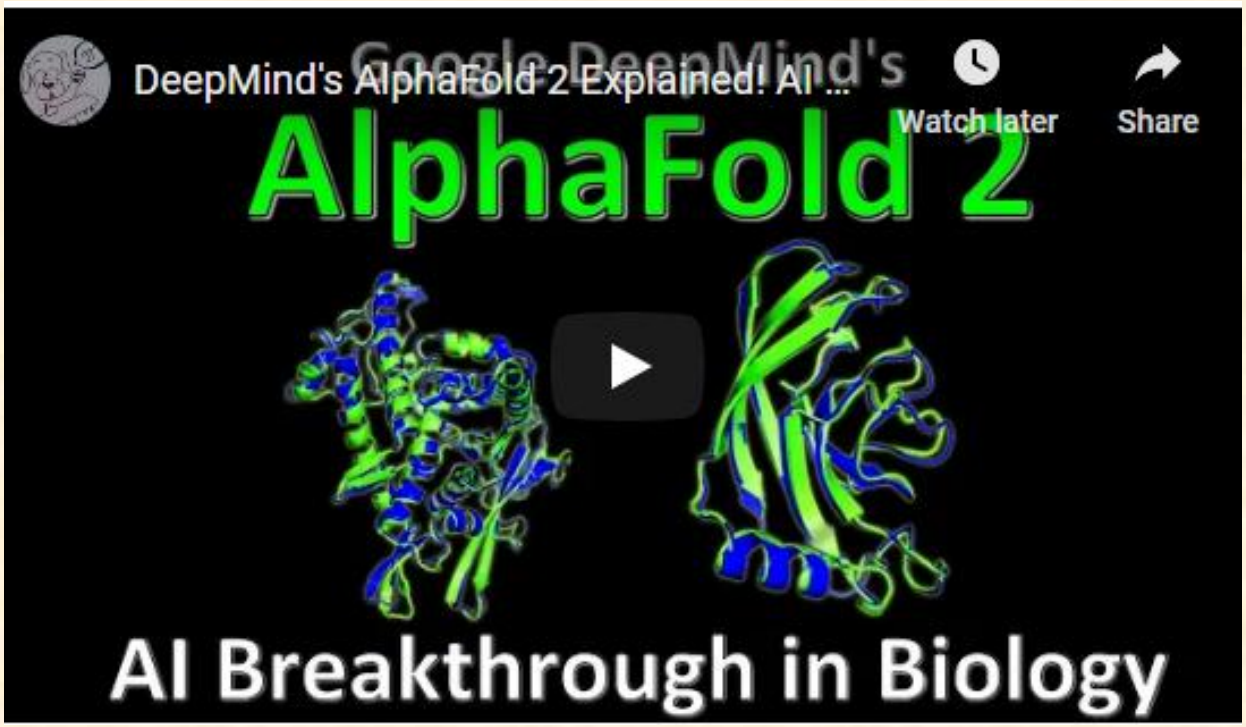
would combine old and new with a pipeline incorporating deep learning for scoring and gradient descent for objective optimization.

Method 2: Gradient Descent on Deep Learning Scores

The core deep learning aspect of DeepMind's winning entry was a neural network that predicted likely distances between amino acid pairs, as well as the angles of each peptide bond linking amino acid residues. These two predictions were then incorporated into a score along with 'score2' from Rosetta modeling software followed by gradient descent to minimize the combined objective cost directly.

This is a bit of an unfair simplification of all the engineering work that went into making the process work so well. But conceptually, the winning prediction strategy was surprisingly simple. It wasn't an end-to-end deep learning project and the fancier method involving GANs didn't perform as well.

Taken together, the DeepMind entry suggests that clever strategy can still beat brute force computational resources applied with big deep learning models and emphasizes the need to maintain some flexibility in your machine learning hardware and software stack. This is somewhat in contrast to the approach espoused by OpenAI.



Reader Comments

1. Catkuma

Funny thing about the Enigma. Its workings were actually filed as patents including its evolution over time. Those were unclassified and generally available and could have helped the War effort if someone thought to look them up.

I actually sell a decorative print I made from the Enigma patent filings although I won't <link> it because I'm classy and also it's probably not allowed. If you go searching, don't fall for all those other idiots that'll try to sell you a print of the SIGABA machine (the Allied cipher machine) and call it an Enigma. It was also a cool device but totally on the opposite side in the War :).

if on the Internet, Press <BACK> on your browser to return to the previous page (or go to www.stealthskater.com)

else if accessing these files from the CD in a MS-Word session, simply <CLOSE> this file's window-session; the previous window-session should still remain 'active'