CALCUTTA INTERNATIONAL SCHOOL
KOLKATA-107

Computer Science Club
presents the

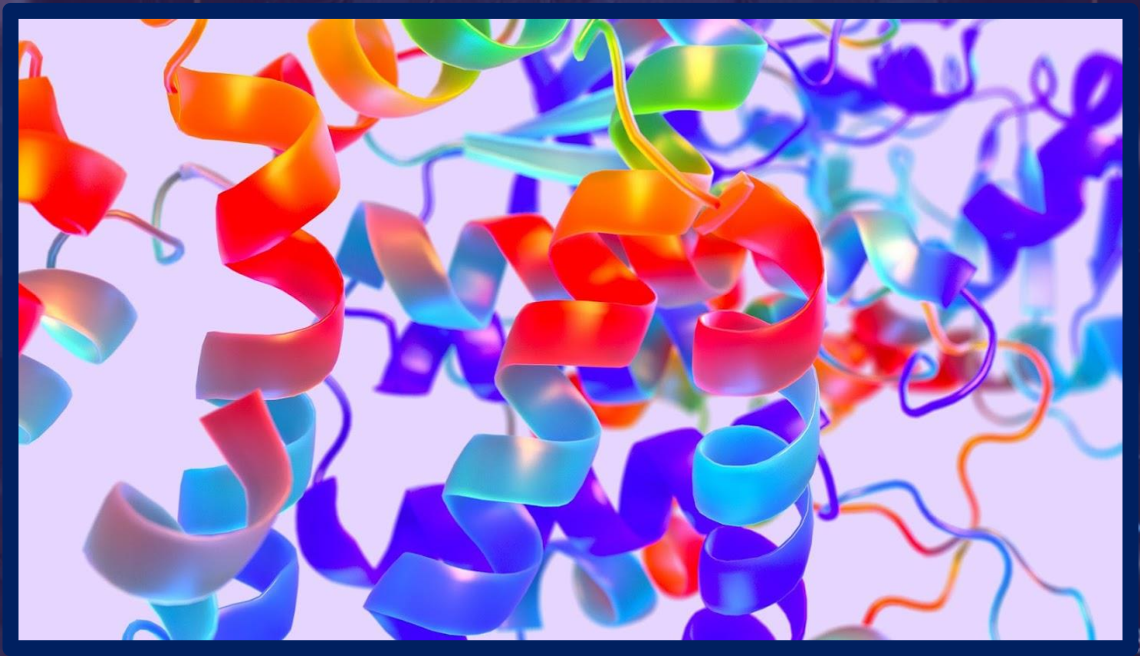
Computer Science Newsletter

January 2021

DeepMind AlphaFold 2 – one of the biggest breakthroughs in the application of Artifical Intelligence

-Vikhyat (IB1)

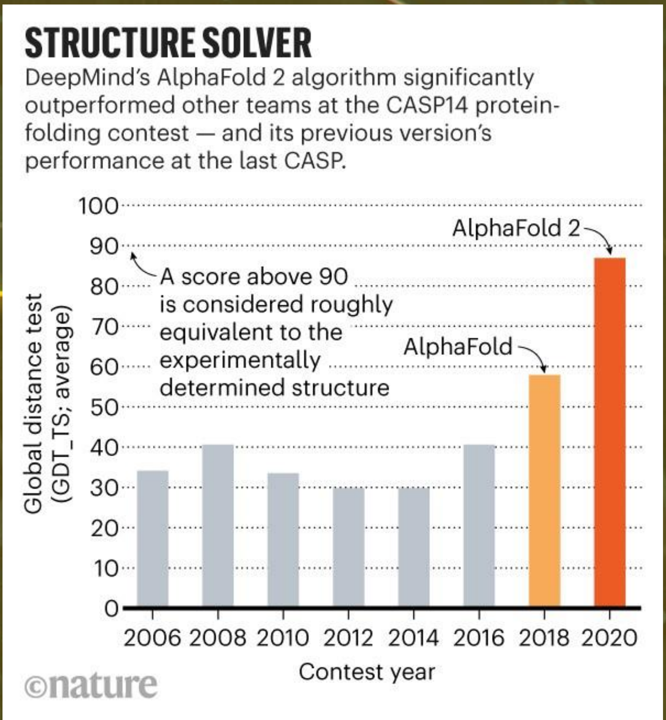
On 30th November 2020, Google's AI research company *DeepMind* announced that it had 'solved' the protein-folding problem, one that researchers have been trying to answer for nearly half a century. Let's take a look into this breakthrough and its possible implications.



The protein folding problem has been one of the most important problems in the field of computational biology. It deals with the prediction of a protein 's structure given the sequence of amino acids that form it. Proteins are essential building blocks for all organisms, and their functions are dictated largely by how their structure folds up in 3 dimensional space.

https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology
https://www.nytimes.com/2020/11/30/technology/deepmind-ai-protein-folding.html
https://www.vox.com/future-perfect/2019/2/15/18226493/deepmind-alphafold-artificial-intelligence-protein-folding

The Critical Assessment of Structure Prediction (CASP) conference has been held every other year since 1994. Research groups from all over the world present their progress in the prediction of protein structures. This year, DeepMind's AlphaFold 2 program surpassed all expectations and achieved a score of 87 on the Global Distance Test that measures the accuracy of the structure prediction (the threshold for the problem to be considered 'solved' is 90).



AlphaFold 2 incorporates multiple deep learning algorithms – such as Residual Neural Networks (ResNet) and attention mechanisms – with a large data repository of over 170,000 protein structures, and uses parallel processing on 100-200 GPUs for its intense computations. The result was an prediction system with a margin of error on the magnitude of an atom's width.
"This is a big deal. In some sense, the problem is solved," says John Moult, CASP's co-founder.

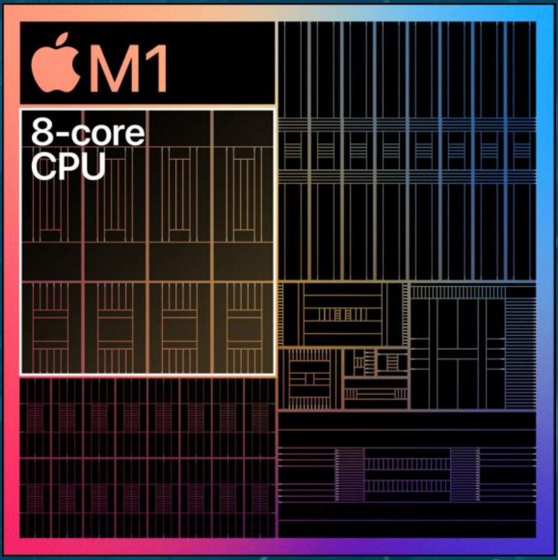
This breakthrough is expected to have a huge impact in the field of drug-discovery, where the 3D structures of a virus's proteins decides largely which drugs can neutralize it. AlphaFold can largely reduce the time taken for this structural prediction – it took half an hour to determine the shape of a protein that researchers in Germany spent a decade trying to understand. DeepMind is now preparing a peer-reviewed paper to publish; the scientific community is enthusiastic to see this breakthrough save lives.

https://www.nature.com/articles/d41586-020-03348-4

Apple's Switch to ARM Processors

-Dhiman Narayan (10)

On the 22nd of July, Apple had announced in their World Wide Developers' Conference keynote, that they would switch from the intel x86 architecture CPUs to their own ARM based Apple Silicon chips. In the keynote, they had shown a Mac Pro with their new M1 chip running a new version of Mac OS (Big Sur). They had also said that we could get a glimpse of these new chips by the end of the year. Well, they kept their promise.



To begin with, lets view the reasons for this switch from the intel x86 architecture to apple Silicon, and why Apple decided to make the switch.

Firstly, as the M1 is their own chip, Apple can decide how they want to modify it for different Macs and doesn't have to depend on intel anymore.

As earlier, to make an update to the Mac lineup, apple would have to wait for intel to reveal its newest chip and then release their MacBook, and if intel hadn't improved much in their chips, apple would have to go with it. So basically, this switch eliminates their dependency on intel, as they have complete control over the chips, their releases and their Macs.

The M1 chip is modelled after the same architecture used in their A14 chip, used in the iPhones, which means that M1 Macs can now run most iPhone apps without any problems. This widely increases the app selection for these devices.

As this is their own chip, apple can highly optimise Mac OS to take full advantage of these chips and use them to their limits greatly improving the performance and battery life of the devices.

Keeping optimisations aside, the chip in itself is extremely powerful. Unlike the x86 architecture, where the components like the GPU are in different places on the logic board and are then connected to the CPU chip. With M1, these technologies are combined into a single system on a chip, providing a new level of integration, greater efficiency and better performance.

In this chip, Apple also features their unified memory architecture, or UMA. The M1 unifies its high-bandwidth, low-latency memory into a single pool within a custom package.So, all of the components in the chip can access the same data without copying it between multiple pools of memory. This dramatically improves performance and power efficiency.

This chip is built on a 5 nanometer process which is extremely fast and efficient and is the first desktop class CPU of it's type. The CPU is an 8 core CPU, with 4 performance cores and 4 efficiency cores, delivering a higher performance to power usage ratio, giving higher performance per watt.

These are most of the advantages of the M1 chip and they seem to outweigh any disadvantage, but this switch did initially receive scepticism from people. So, to dive into its disadvantages.

Firstly, as this chip has a completely different architecture from the older intel based Macs, it means that it wouldn't be able to natively run x86 applications.

To this, apple has a solution. As all developers will have to modify their applications, to be optimised for the Silicon Architecture, Apple said that it would take around 2 years to have most applications available natively for the M1 Macs. But, in that time period, those applications could still be run using their Rosetta 2 emulator.

The other main disadvantage is that we cannot use eGPUs anymore and have to say goodbye to Mac Bootcamp, which means no more windows on Macs.

Just like Apple had promised, they released an updated version of their Macbook Pro 13 inch and Macbook Air with the M1 chip, at he same price as the intel versions. These are available for purchase on their website.

These Macs have been tested by many critics. Initially they had thought that the performance of these chips would not meet expectations, and that x86 applications would run slow through Rosetta and that overall, the package would not be a good experience till the entire transition would be done. But, they were wrong.

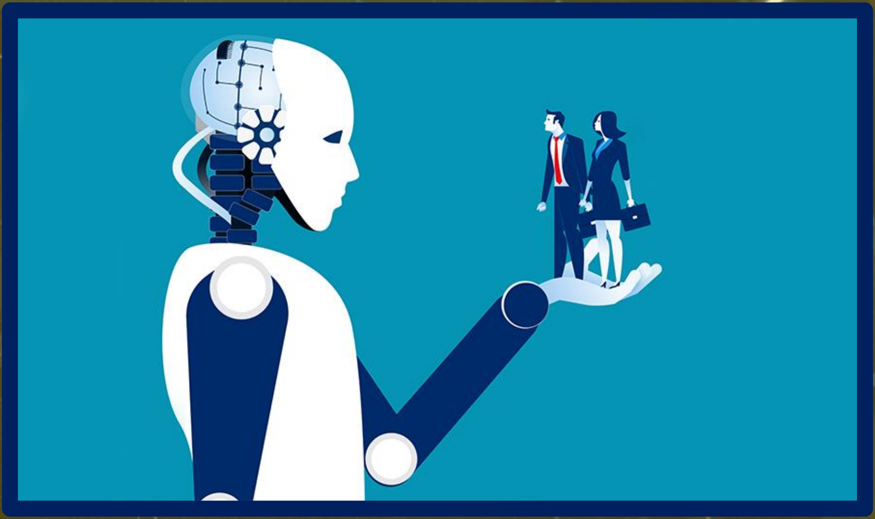
These new Macs can run native apps much better than their intel counterparts and in fact even run x86 apps as well as the intel machines if not better. The OS due to heavy optimisation is extremely snappy and is a pleasure to use. The integrated GPU is also powerful and almost equals the performance of the dedicated GPU on the base Macbook Pro 16 inch with the radeon pro 5300m for a lower price.

So should you purchase one or not? If you are looking for a new computer, and want to purchase an Apple laptop, these are the ones to go for. The intel versions are still great but these will give better performance and battery life. Pretty much everything that the average user would need and want. Even then, the more powerful Macs, like the Macbook pro 16 inch and iMacs are yet to be released with the Silicon Architecture and rumour has it that they would be released with a more powerful chip. The M1x. But in the end, only time and of course Tim Cook will tell what is to come.

Can machines really take over the world?

-Kiron Deb (A2)

When we say robots could threaten our existence, we say it half jokingly. After all, it does seem a bit ridiculous: just unplug the fearsome robots and we're all good, right?
Not quite.



Our frivolity might come back to haunt us centuries from now, when technology progresses to the point where a farfetched futuristic sci-fi setting becomes a reality. In particular, as Artificial Intelligence (AI) continues to improve by the decade, we must be wary of AI's possessing 'superintelligence' (intelligence far surpassing that of the brightest human minds). If an AI becomes smart enough to learn how to program, it could alter its own code to increase its intelligence. As its intelligence improves, it might get even better at programming and could re-program itself yet again, leading to a frenzied cycle where human intelligence is quickly left far, far behind through exponential self-improvements. If any of these superintelligent systems competed with humans, their advantages would be overwhelming. They would outclass us in technological research and computation and may even be skilled at more 'human' areas like social manipulation.

The reason why the simple trick of shutting down superintelligent agents won't work is because they would have already considered that possibility and taken prevention measures so they can pursue their goals. Without having a carefully encoded moral compass that aligns with human values, they could resort to horrifying means to accomplish their objective. Nick Bostrom, a philosopher at the University of Oxford, illustrates this with a couple of disturbing thought experiments:

If we give an AI the goal of making humans smile, it could end up inserting "electrodes into the facial muscles of humans to cause constant, beaming grins" since that would be an efficient way to achieve its goal.

An agent whose final objective is to solve the Riemann hypothesis (an unsolved problem in mathematics) could transform the entire Earth into some form of "programmable matter" to assist in the calculation.

Even though the final goals in these examples appeared harmless, the optimal route ended up being rather dangerous. A simple question that might come to mind is "why can't we just program all AI's to not harm humans?" Sadly, many researchers believe that preprogramming a superintelligence with a full set of human values will be an unbelievably herculean task, primarily by virtue of the human moral system's sheer complexity and ambiguity. In other words, having a machine be independent and ethical is much, much harder than it sounds because ethics itself is quite a gray area.

Now, as over-the-top everything you just read might seem, you might be surprised to learn that notable figures such as Bill Gates, Elon Musk, and the late Stephen Hawking have shown genuine concern regarding the existential risk posed by uncontrollable Artificial Intelligence, with Hawking speculating it could "spell the end of the human race", Musk tweeting "...We need to be super careful with AI. Potentially more dangerous than nukes." and Gates stating "I agree with Elon Musk and some others on this and don't understand why some people are not concerned." Hawking even went so far as to say that in the coming decades, AI could offer "incalculable benefits and risks" such as "technology outsmarting financial markets, out-inventing human researchers, out-manipulating human leaders, and developing weapons we cannot even understand."

If you'd like to read more on the topic, I'd recommend the following pages:

Asimov's Laws of Robotics, and why AI may not abide by them | by Hans A. Gunnoo | Towards Data Science

Existential risk from artificial general intelligence - Wikipedia

Computational Problems (From Project Euler)

Easy

Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:
1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...
By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Medium

The sum of the primes below 10 is 2 + 3 + 5 + 7 = 17.
Find the sum of all the primes below two million.

Hard

n! means n × (n − 1) × ... × 3 × 2 × 1
For example, 10! = 10 × 9 × ... × 3 × 2 × 1 = 3628800, and the sum of the digits in the number 10! is 3 + 6 + 2 + 8 + 8 + 0 + 0 = 27.
Find the sum of the digits in the number 100!

Python Solutions
https://github.com/nayuki/Project-Euler-solutions/blob/master/python/p002.py
https://github.com/nayuki/Project-Euler-solutions/blob/master/python/p008.py
https://github.com/nayuki/Project-Euler-solutions/blob/master/python/p020.py