Complexity: Why Our Brain Succeeds and AIs Fail

Luca Dellanna
Email: Luca@luca-dellanna.com
Website: www.luca-dellanna.com
Twitter: @dellannaluca

In this paper, the author examines the differences in handling complexity between the human brain and artificial intelligences.
It will be shown how the current modularity and tendency to reduce complexity of most current AIs is a limit to their potential to reach artificial general intelligence (AGI).
Finally, techniques to address these limitations and to properly address complexity are addressed.

The main reason current AIs fail to achieve intelligence outside of very narrow use cases is that their developers tend to reduce complexity too much.
Many software developers perceive coding as a satisfying activity, as it reduces complexity through modularity. Software programs consist into a set of modules which communicate between each other through protocols called APIs or function signatures. (I am making a huge simplification here; I do not aim to be fully exact with terminology here but to be clear with the matter at hand: how software handles complexity.) Each module a software developer writes has to exhibit a given behavior dependent on its inputs and otherwise independent of the state of the system. By "module" here I intend any method or function.
The modular approach has many advantages. It declutters the mind of the programmer, requiring him to handle only a few variables at a time. It allows to test code in a reliable way ("unit tests"). It allows for large teams to split the work efficiently. It allows for developers with different skills to work on different parts of the program.
What modularity does not allow, is progress towards artificial general intelligence (AGI). Let me explain.

Skin in the game

Evolution works through natural selection. The unfit organisms die and the fit ones survive.
The key here is the word organism. If an organism develops through a mutation an organ which performs its main function better but somehow leads to lower overall fitness, then that individual results unfit and dies.
The mutations are purely evaluated on the basis of whether they increase the survival of the organisms or species bearing them. The mutations have skin in the game of their hosts.
Conversely, in modular software, a module (which can be any block of code which expects one or more kinds of inputs and is expected to produce one or more kinds of outputs; for example, a function or a method) is only judged on his behavior.
Modules which perform their job better (as dictated by the programmer) but which perform their job worse (as dictated by the consequences of the behavior of the whole AGI entity) are retained.
This is not how nature works. In nature, mutations that brings a negative contribution to the organism or to his species disappear.

The human brain is made to act. Those who believe that the job of the human brain is to perceive correctly are puzzled by the many biases and glitches it has. Why do we see human faces in clouds? Why do we exhibit "irrational" behaviors such as risk aversion in prospect theory?
Instead, those who believe that the job of the human brain is to act in such a way to maximize survival do not see any contradiction in its behavior. We see human faces in clouds because that's the same bias that allows us to spot the silhouette of a tiger hidden in the vegetation. We show risk aversion because in real life we do not know the exact probabilities and our individual lives are non-ergodic.

Current AIs are made to perceive. Conversely, software is not only judged on its ability to act correctly.
In most current AI systems, at least some of their modules (and in some cases, the AIs as a whole) are judged on their ability to perceive.
The mere existence of supervised learning (where the AI or one of its modules tells what it perceived and a human provides feedback on whether the perception was correct) is evidence of this.
Of course, there are modules which are judged on their ability to act – for example, those charged with commanding actuators such as pistons or electrical motors. But the mere presence of at least one module which is judged a parameter different than whether the behavior of the system as a whole increased its survival compromises the potential to the general intelligence of that system.

How human brains handle complexity

In this section, I will introduce a lot of simplifications, for the sake of brevity and clarity. In luca-dellanna.com/research I explain these contents in larger detail.
The human brain is made of regions of neurons. Sensorial input traverse multiple regions, ascending a loose hierarchy. In each region, sensorial input undergoes a compression and an expansion. To understand why, the layout of neurons in a region matters.

The layout. Input to a region comes from a large number of neurons, which are called "projecting neurons". At any given moment in time, some of such neurons are active (they send electrical impulses) and others are inactive. This means that the input to a region can be represented by a string of bits, each bit being one of the projecting neurons and its value depending on whether the corresponding neuron is active.
The neurons in a region are organized in columns. Each neuron in a given column receives input from the same subset of projecting neurons. Each column fires if it enough of its inputs are active. In other words, a column fires it recognizes a given pattern in the whole input to the region.
The number of columns is much lower than the number of projecting neurons. This means that the pattern of active columns can be represented with a much lower number of bits than the pattern of active projecting neurons.

Compression. Because the pattern of active columns in a region is expressed with a lower number of bits than the pattern of active projecting neurons (the input), it can be said that each region performs a compressing function.
However, this is only half of the signal processing performed by the region. The other half is represented by expansion.

Expansion. Each neuron in a region, in addition to receiving feedforward input from projecting neurons, also receives modulation input from other regions of the cortex. This modulation input kind of represents the status of the other regions of the brain.
The inhibition mechanisms present in the brain ensure that for each active column, only the neurons which recognize a pattern in their modulation inputs fire.
The following analogy can be made:

1. Individual columns fire if they recognize a pattern in the input to the region ("if they observe a given feature").

2. Individual neurons fire if the column they are in is active ("if the feature they represent has been observed") AND if they recognize a given pattern in the output of the other regions ("if they observe a given context"). Therefore, the output of each neuron is a "contextualized feature".

3. For example, a given region might receive inputs from millions of neurons encoding some perception. With compression, the region recognizes some of the few thousand features it can recognize. With expansion, the region recognized one of the few hundred of thousands contextualized features it can recognize.

(There is a complication: perceptual objects are not represented by single neurons but from sparse distributed representations. But, for the matters of this article, this complication can be ignored.)
Because the output of the region is the output of all its neurons, it is represented with a higher number of bits than the number necessary to represent the output of the columns (each column contains multiple neurons). Therefore, the signal traversing the region gets expanded as it exits it.

Alternating compression and expansion. As shown in the previous sections, as the signals from our peripheral nervous system traverse the brain from one region to the other ascending a loose hierarchy, each region applies two transformations to it: a compression and an expansion.
The question is: upon which dimensions does the compression happen, and upon which the expansion?
The answer is: the compression happens on the dimensions of the space in which the patterns learned by the region at hand are represented, and the expansion on the dimension of the space in which the patterns learned by the next regions in the hierarchy are represented.
The demonstration is quite long and complex; the interested readers can find it in (Dellanna, 2019).
There are some important consequences:

1. How a region compresses information (which dimensions it discards in choosing the space in which to represent the info), and how it expands it (which dimensions it adds) are all variable in time and depend from learning.

2. In particular, which dimensions are discarded depend from what the region learns does not contribute to meaningful patterns and which dimension are added depend from what the other regions learned it contributes to meaningful patterns.

3. From the previous two points, it follows that no region communicates through static APIs to other regions. Rather, through learning, each region adapts not only its processing but also its communication protocol to other regions (its APIs).

4. The two points above result in an impossibility to "train" each region in a vacuum and with static dimensions on which the input and the output are represented (ie, "static APIs"). This strongly contrasts with modular programming, which requires static APIs between modules.

In common Neural Networks, weights are dynamically adjusted to modulate the flow of information between two nodes. This is not enough. In our brain, it is not (only) the weight between neurons that is adjusted, but the dimensions over which information represented by the output of regions is encoded.

How the human brain adds meaning

In the first region through which the human brain receives sensory information, all columns (ie. all dimensions) represent sensorial information.
As the signal climbs the loose hierarchy of regions, each region removes some dimensions and adds new ones. Most of the ones it removes are sensorial, most of the ones it adds are semantic.
The dimensions added depend on the context (the output of the rest of the brain); and context is meaning.
As sensory signal undergoes an alternation of compressions and expansions, meaningless sensory information is filtered out and meaningful semantic information is added (context). This process is done through multiple iterations in order to allow recursivity. What is context for a region, if it contributed to the expansion, becomes content for the next one.
The human brain does not remove meaningful complexity. Rather, it changes the dimension over which complexity is expressed, simplifying what is not important and adding complexity when it is meaningful.
Whether something is meaningful, is determined by whether it contributes to taking actions which are of benefit for the organism and/or for its species. (Because the feedback to learning is always feedback which impacts the individual as a whole.)

How AIs handle complexity

In the previous sections, we saw how the human brain does not only use experience to determine which input bits are important but also which dimension of encoding are. It uses experience and learning to determine not only the dimensions over which to remove complexity but also those over which to increase it.
Conversely, because most current AI uses modules, they tend to compress the bandwidth of information exchanged between one module and the other.
Let me clarify: brain regions are not modules, for they receive input from the rest of the brain and for they do not reduce the complexity of their input, only change the dimensions through which it is expressed. The different areas of the human cortex are specialized but not modular.
Because most current AIs tend to reduce complexity, they cannot learn complex combinations of input and context – combinations in which the output is of similar or larger meaningful* complexity than the input. They cannot properly process context. They behave, in a way, as if they suffered from low-functional autism spectrum disorder.
(*): something is meaningful if it informs behavior in a way to increase the probabilities of survival (or if it shortcuts the reward function of the organism, as in the unfortunate case of addictive substances).
In the rare cases of AIs which do not reduce meaningful complexity, if they are modular, they will add complexity over dimensions which are meaningful to them rather than to the AI system as a whole.
The fact that behavior is the output upon which humans are judged ensures, in some measure, that complexity is added upon those dimensions which lead to a more effective behavior.
When some form of modularity is added, this falls apart. When modularity is added in the form of a lack of direct consequences / lack of skin in the game, Intellectuals-Yet-Idiots emerge, creating complex theories with no improvement to the real world. When modularity is added in the form of a modular software architecture, software modules which do not improve the overall behavior of the AI system emerge.

Conclusion

If a machine will ever reach general artificial intelligence (which, by the way, I consider an existential risk), I predict that it will have the following characteristics:

1. No software modules (at least for its whole core software).

2. Nodes which do not only decrease complexity of the information arriving to them but also increase it according to information outside the feedforward input to the node (example).

3. An intelligence limited to the output expressed through actions over which it has some form of skin in the game. I.e., its output over which it won't have skin in the game will be dumb.

References

Dellanna, L. (2019, Jan). *Techniques for the emergence of meaning in machine learning (ml).* Retrieved from `https://osf.io/4g56t/` doi: 10.17605/OSF.IO/4G56T