Intelligence from Model Theory

by Leo Sousa

July 30, 2018

Part I. Meta-models

First thing I want you to figure out is 'What is meta-thinking', and to know that this is what you will be doing here: Meta-thought, meta-cognition, meta-memory, is when two models interact unidirectionally, in example, when a model of a mindset of a parent simulates or at least emulates the mind of a child. On a more subjectively interesting context meta-thinking is simply put the result of questioning yourself "What was I thinking on a previous moment?", which suggests cognition fluctuates with time. Well, it does fluctuate and this is not simply a result of mental system instability. This is rather a positive net function of time, a function $f(t)$ that quantifies overall problem solving which you define intelligence with. Exemplifying this definition of intelligence, let's consider the mental function $g(x)$ that yields how much time it would require your brain to come up with a solution made with elements of x non-familiarity. To make this written monologue twenty minutes long, you could acquire yourself a text of similar reading speed and measure how much you read of it on the goal time of, again, twenty minutes. If you are writing the video monologue on a text editor, you could now copy such text here and do a simple visual comparison. Another solution that doesn't require external sources of similar texts is to start writing the overview, or the introduction, or just what is on the top of your head concerning the monologue, and to measure your well-pronounced reading speed S of it, only to divide S by number N of written lines, or any unit of text, resulting in a variable V. To divide the goal time T, by V, is to find the total number of lines you want to type in. These two solutions correspond to x_1 and x_2 for each individual, so in practice $g(x)$ is actually a function of at least two variables, $g(x, y)$, for y equals the number corresponding to each individual's intelligence. The higher the y, the lower x_1 and x_2 must be if

$$g(x_n, y) = g(x_n + i, y + 1)$$

so $g(x, y)$ can be referred to simply as $g(x)$ when $i/1$ in complexity/intelligence units is known. Thus starting from intelligence defined as quantified problem solving we get a proportional relation of intelligence and complexity. So even though quantifying complexity is a long term goal complexity can be easily increased and decreased enabling intelligence tests by progressive hierarchy. This leads to the idea of intelligence defined in active memory terms: the ability to hold an α number of elements simultaneously in your mind. On this category a very popular test is the count of digits correctly reproduced by memory after display. In its many variations this test should ideally not compensate for differences in information coding but simply include the common elements of a set of problems. That is, for a man who is able to memorize 76 digits but only 12 random letters it is simply expected a better performance at problem solving involving digits. The same memory capacity can store more elements if they are better encoded. This is how complexity subjectively decreases with familiarity. Considering the size of the median brain the median 7 digits span can only imply a lack of encoding. To improve encoding is effectively to increase your vocabulary, id est: It is simpler to remember known numbers, words, symbols, pictures than to consistently remember novel ones. This rule applies even to recently learned elements and explains Psychology's phenomenon called Priming. It also proves how much problem solving skills relate with pattern recognition volume because patterns correspond to mental vocabulary, thus validating the Raven's Progressive Matrices(RPM) also known as Raven's IQ test. Moving on, you need to model $f(t)$ and relate it with $g(x)$, then relate $g(x)$ to a biological model of working memory for $f(t)$ theory validation. Think of an abstract two opponents game played to its finish. The first round ever is determined by luck, the winner randomly produced action orders and so did his fellow. From then on the loser proceeds to the next round by analyzing his opponent previous moves and by creating a winner strategy assuming the same moves will happen. The winner can either repeat actions or not based on what he expects the loser to do. If this game is rock-paper-scissors an original winner that thinks losers change strategies like that will always win. Otherwise they will perfectly alternate in victory. Now if you increase the complexity cap on the original loser's mind he understands losers of lower complexity who go for the next win assuming exact same conditions, so he must base his action on what his opponent effectively thinks of him. And here is the logical rule: the Opponent can only model you as less than him, so you don't know the exact model the Opponent has of you but you can start at the hardest however possible conditions as long as you know how smart he is and you will eventually always win. Starting on your own level is less productive so you want to correctly guess your opponent's intelligence. Exempli gratia: On the Rock-Paper- Scissors game you have 5 points in intelligence, that is, you can model 4 levels of lower players. Your opponent has 3 levels on intelligence and let's say he starts winning with a Rock. You think of Loser[1], or Loser I, who would plan for a Paper now; You think of Loser[2], or Loser II, who would plan for a Rock; Loser III plans for Scissors; Loser IV plans for Paper; Loser V plans for Rock. You also think $Winner[N] < Loser[N]$ and $Loser[N] < Winner[N+1]$. You can see the Loser wins with the same N intelligence because he has a simple motivation to change and the winner has a simple motivation to not change. So the worst possible winnable scenario here is to model his intelligence as IV and expect him to win against Loser III, so you go with Paper. He's a winner III so he modeled you as a Loser II, so he actually goes with Paper. It's a tie, that is, you both lose. But now you know he acted against Loser II, so he must be Winner III, thus giving you the edge over the rest of the game. Of course, the Opponent is now losing and will rely on chance to reduce his loss. Realizing that for

$$Loser[X + 3(N - 1)] = Winner[X - 1 + 3(N - 1)]$$

for X = [1,2,3], which you can call The Simple Rock-Paper-Scissors Equation(SRPSE) corresponding to the three possible moves, only evolves complexity to another hierarchy still based on N: On this new hierarchy Loser I assumes one static pattern consisting of more than one move, it can be the assumption of an observed perfect randomness by Winner I which is what he could be doing after realizing the SRPSE and not going any beyond that; Loser II assumes an alternation between two patterns by Winner II. From this position you can see that assuming no pattern whatsoever and accepting pure chance in results is the worst strategy, unlike popular belief. Also, you now understand how the winner complexity increases as both player's N go up. Since realistically their N's will always be different, a real two opponents game can be always summed up to a model containing a smaller model. So if $f(t)$ increases with time, then you now know that the simple self-interaction of a meta- cognitive mind increases its own complexity. Biologically, meta- cognition could happen between many pairs of brain parts, for example: the inner primitive brain vs the outer modern brain; the left hemisphere vs the right hemisphere; the visual cortex vs the frontal lobe. To determine a winner model between two parts a third one is called and the majority defines a win: The visual network says there is something moving close in front you, the audio brain says there is nothing producing sound there, and the tactile neurons confirm its existence. From this simple sense-model contrast a whole new theory arises to explain the loser model from the winner model's perspective, in this case, a completely visual, geometrical, theory of sound as waves of vibrating particles emerges.

Part II. Intelligence and Language

Now to move forward it makes sense to relate this modeling with language: language defined as a set of references of syntactical signifiers and semantical denotations. A signifier requires a denotation without exception, thus it locally duplicates the initial data. This system enables access to the data if the initial source is unavailable. With this system data can be combined in ways that don't happen naturally through the senses. This combination is made of abstractions into a complexity that corresponds to a number. Each abstraction implies the existence of its logical negation eg: 'five' implies there is 'no- five'; 'star' implies there is 'no-star'; 'luck' implies there is 'unluck'; 'predictable' implies there is 'unpredictable'. This binary rule, called Sentential Logic, qualifies the mind with semantical freedom in a way that every socially imposed idea can be compared with its negation. So ideally a mind cannot be reduced with Sapir-Whorf Theory in an Orwellian scenario. But non-ideally it happens, and with your previous definition of intelligence you can assign 'smart' to the ideal mind, capable of unbound sentential logic, and you can assign its negation to non-intelligence. From Non-intelligence's perspective, the meanings of 'no-five', 'no-star', 'unluck', 'unpredictable' are not easily reachable from 'five', 'star', 'luck' and 'predictable' alone, but are instead independently conditioned or not in their minds. The non-intelligent mind will not understand simple unknown negations like 'no-five' and 'no-star' and will probably fight against such word usage. You expect these irrational minds to be subjects of Orwellian abuse: they will learn about slavery, servitude, inability and subordination, and will be unable of thinking about their antonyms. In the book 1984, antonyms are put together in a sentence to be equal for this very purpose: war is peace, freedom is slavery, ignorance is strength. Orwell explains that the Party could not protect its iron power without degrading its people with constant propaganda, which main premise here is that there are no logical negations since the logical negation of X is equal to X. Language can also be built in a distracting manner creating a net of useless relations e.g. you know the antonym words 'wrong' and 'right', however 'right' has also the unrelated antonym relation of 'right' and 'left'. This limits language to a contextual ground and if the context can't be built for one of the parts then this part is left out of meaning. Also, the cost of this context setup can be enormous and reduce efficiency even if the communication happens successfully. Undefined contextual words were the mark of uncivilized barbarian languages back in Classical Greece. Intelligence then proceeds to eliminate this and define itself without it. Thus there is a relation between Sentential Logic and non-contextual languages. You could think that this relation is: context definitions make it impossible for assigning True or False values given there are always two subjective contexts that assign opposing values to the same statement. This puts a contextual speaker in a position of immunity from refutation and at the same time nullifies the idea of tautology in formal logic. in other words, contextual language makes it that no speaker is ever wrong and no statement is every right on itself, which is an interesting contradiction. However, the negation of contextual language is possible: that not all speakers are perfect and statements can be universally correct, in other words, that tautologies exist. With the existence of tautologies the mind is able to assign arbitrary truths and build a relative system on top of it. With this seed set of initial truths you can arrive at Model Theory avoiding meaningless questions that forbid you to define things on each other, which for example would make science, physics, and dictionaries impossible. Again, the negation of that then is to see each thing as fundamentally separate of all others and even to have no belief system whatsoever; Both these requirements of the negation are physically and logically impossible. Every possible physical event is already part of your belief system, whether your brain has its specific value or not. Eg: You may spend your whole day without thinking about it, but you believe there won't be a meteor crashing your house tomorrow otherwise you would move out today if you wanted to avoid the hit. Your brain has a default belief for different categories, either you are specifically aware of them or not. There is not a single belief system that operates with a third value different from both Yes and No. The ideal mind is aware of this binary law and is able to model it even before it learns the Physical laws in your Universe. The conscious operation with third and n-th values will have to be translated into Yes or No when decision making is required under a time limit. This creates the winning difference between two minds of same computational power when only one is illogical. For example: after learning about probability you start to assign values with a range of one hundred units that correspond to the statistical chance of future concerning events. When deciding to walk out with an umbrella you remember the News mentioned a 34 percent chance of local rain this morning but you either prepare yourself for it or not. So this number needs to be processed into a yes or no and this could be done much earlier if you were aware of this final binary solution. Most people are inclined to believe that low chances mean 'No' and higher than 50 percent chances mean 'Yes'. This would lead you to bring out your umbrella 100 percent of the times when 50 percent chances are given. It would also lead you to never bring your umbrella at low chances, which could become an issue. Knowing in advance of this binary solution you would however rely on a more accurate system, one that gives you either a Yes or No with much higher rate of success. This cognitive difference applies to every action you take since every action is a decision. And the lack of translation wouldn't consume mental resources on the fly, freeing them for your momentary desire. The many-valued-To-binary translation in fact is already an indicator of a bad solution because it implies the solution machine does not operate logically and doesn't communicate with the physical action system. Taken with the different-mental-parts-modeling theory presented in Part I of this video, this indicates again that a lack of inner interaction corresponds to low intelligence. This explains how supposedly Einstein's brain had success in problem solving and Physics: his left and right hemisphere, along with other parts, had greater inter bandwidth than the median brain. Applied to AI research, this theory indicates the path of how to reduce a computer mind whilst increasing its effectiveness towards a 20 watts brain emulator: the processing must involve the final binary output from start and the feedback should improve the rate of prediction. An artificial algorithm that could come to your mind is what was introduced in 2014 as Generative Adversarial Networks. "[] Used in unsupervised machine learning, it composes of a system of two neural networks competing against each other in a zero-sum game. One network is Generative and the other is the feedback, called the Discriminative Network. The Generative Network is taught to map from a latent space to a particular data distribution of interest, and the Discriminative Network is simultaneously taught to Discriminate between instances from the true data distribution and synthesized instances produced by the Generator. The Generative Network's training objective is to increase the error rate of the discriminative network. a particular dataset serves as the training data for the Discriminator. Training the Discriminator involves presenting the discriminator with samples from the dataset and samples synthesized by the Generator, and backpropagating from a binary classification loss. In order to produce a sample, the Generator is seeded with a randomized input that is sampled from a predefined latent space, e.g., a multivariate normal distribution. Training the generator involves backpropagating the negation of the binary classification loss of the Discriminator. The generator adjusts its parameters so that the training data and generated data cannot be distinguished by the Discriminator model. The goal is to find a setting of parameters that makes generated data look like the training data to the Discriminator Network. In practice the Generator is typically a deconvolutional neural network and the discriminator is a convolutional neural network. It is termed Turing Learning, as the setting is akin to that of a Turing test.