

5 A Duel: Science Versus Technology

The idea here is food that I love eating but it doesn't give me a nervous breakdown to cook. What I'm after is minimum effort and maximum pleasure, in both the cooking and the eating.

—Nigella Lawson, *Nigella Bites*, season 1, The Food Network, 2014

Chomsky did amazing things, but his time is over.

—Geoffrey Hinton, Romanes Lecture, Oxford University, 2024

Two Different Storyboards

The time has come for an evaluation, so let's wrap up. Although boundaries may be vague at times, the division of labor between science and engineering is this: engineers design and build tools, for us to use; scientists construct and test theories against observations and results, with the hope of understanding the world around us and exploring it systematically.

Engineers, who have endowed us with the gift of the new AI, with LLMs as its current top representatives in the language domain, offer a plot that's easy to follow: a brain-like model, solely based on vectorially represented text statistics, implemented in working computer programs that do things, allegedly with no symbolic rules. LLMs classify texts; engage in conversation, translation, text generation and summarization; and perform other tasks, most prominently, next word prediction. Their current makeup has rendered many objections to the old AI obsolete,¹ and a new revolution, we are told, is already here, with consequences that are not only practical but also scientific, aiming to change cognitive neuroscience. Symbolic rule systems such as those I presented in part I are obsolete.

“We now have another conceptual framework alternative,” wrote AI leader Terry Sejnowski recently, “based on learning rather than writing logical computer programs.”² Pretrained on gigantic input sets, LLMs’ outputs appear human-like in a variety of tasks, which makes them a winner in any public opinion poll, including the scientific community.

Still, despite performing well, these models have not offered explanations for natural linguistic phenomena (though they may have offered explanations in other domains), nor have they been geared to discover unknown phenomena, nor have they offered more parsimonious theories (more on this later).

Linguistics as a science presents a different storyboard, one whose main aim is explanatory—to make us understand our own knowledge of language, its development, and its daily use. Indeed, it offers a theoretical apparatus whose goal is to analyze and discover our linguistic cognition. It claims that we humans possess a body of abstract yet highly constrained and precise symbolic knowledge in the form of grammatical rules, generalizations, and constraints, some of which is said to be innate; and whatever knowledge develops in the child, it does so through a rapid process, which occurs with a fairly lean input set. Language scientists formulate the shape of this knowledge and try to understand *why* it is structured the way we describe it. Common to virtually all theoreticians of virtually every linguistic color is the belief in the critical theoretical roles of symbolism, of hierarchies among symbols (at both sound and form levels), and more technically, of abstract syntactic displacement (instantiated in *MOVE*, for example) and the need for a semantic theory in which the meaning of a sentence is a composition of the meanings of its parts.

TGG is one such approach. It presents a theory that is grounded in formal, logic-inspired tools that give way to deterministic rules for the concatenation of symbols into meaningful strings and for the composition of their meaning representations. Probabilities play a very limited role. The claim is, moreover, that certain symbolic knowledge resides in the brain of every human baby at birth, that it is rather abstract (hence difficult to detect and characterize), that it becomes more detailed during the child’s interaction with the environment during language development, and that it governs our linguistic ability as manifest in our language behavior. The reasoning behind this radical claim is the “poverty of the stimulus” argument. This class of hypotheses about linguistic knowledge and its origins is

supported by a broad empirical base and multiple discoveries and is justified by solid methodological considerations. Sadly, it can take credit for few successful technological applications or accomplishments, despite multiple attempts.³ This may be a serious weakness.

The two approaches seem incommensurate. For one thing, they disagree on the formalism (symbolic vs. probabilistic). At the same time, it is important to underscore commonalities. They give us a ray of hope for a collaborative, interesting, insightful and effective future:

- *Computationalism*. Both frameworks assume a formalism that enables computations over abstract representations.
- *Representationalism*. In both, computations are carried out over these.
- *Predictiveness*. The formal nature of the theories enables the derivation of precise predictions regarding future results.
- *Measurable biomarkers*. These computations result in measurable behavior or other precisely quantifiable biomarkers, such as varied measures of brain activity.

These commonalities are critical, as they enable a dialogue. And yet, apart from the symbolic–probabilistic difference, there are important contrasts between the two approaches, which are the topic of this chapter and the subsequent one:

- A. *Innateness*. TGG takes certain grammatical knowledge to be in our head at birth. LLM proponents either deny this or blur the issue.
- B. *Competence versus performance*. Chomsky’s distinction between what we *actually* do and what we have *the potential* to do is a central tenet of TGG.⁴ LLM proponents deny its relevance (or at least blur it).⁵
- C. *Constraints*. Scientists apply various general methodological principles (e.g., parsimony, generality, transparency) to guide their exploration, and they consider the interface of their theories with other aspects of the human science (e.g., neuroscience, evolutionary sciences). In the sciences, parsimony—a theory’s ability to capture more data with less theoretical machinery—is a virtue. Engineers tend to be constrained only by hardware specifications, which may force their designs to be more computationally efficient. Engineers who follow Rich Sutton’s “bitter lesson” (see the epigraph to the prologue of this book) invariably prefer machines with more power, memory, speed, and computational cores to those with less.

D. *Benchmarks and empirical tests*. TGG and LLMs evaluate their systems on very different tests and use very different success criteria.⁶

Who, if anyone, is right? This chapter discusses issues A–C; the next one will address issue D from several angles. To forecast, the conclusions are that while LLM developers have given us great applications, the performance of these machines does not bode well for their status as scientific models; TGG’s proponents, by contrast, have failed to build tools, but their work nonetheless constitutes a tightly constrained explanatory theory of human language, with deep insights into the mind and highly testable predictions. Let’s see.

A. The Innateness Hypothesis—An Attack and Its Falsification

I have already mentioned the debate on the innateness of certain linguistic knowledge. Here is a slightly more philosophical perspective (once dubbed *Plato’s problem* by Chomsky):⁷ how come we know so much, given so little? That is, how does our limited linguistic knowledge empower us with the freedom to construct sentences and stories that we have never heard? Chomsky attributed this perspective on language to nineteenth century traveler and polymath Wilhelm von Humboldt, who had commented that “language is the infinite use of finite means.”⁸ Chomsky called this special ability “the creative aspect of language use,” and he noted that we have the capacity of producing (infinitely) many sentences of arbitrary length, despite the fact that the means our head avails us are finite and that we have only been exposed to a finite set of language materials. He famously ascribed this ability to an innate “language organ” in our head, by virtue of which we have this capacity.

An Assault on Innateness and TGG

To many, the idea that we come to the world equipped with *knowledge* sounds plain crazy (though since Plato’s *Meno* it has been repeatedly proposed, in different formats over the centuries). As Geoffrey Hinton recently commented, “There’s this crazy guy at MIT called Chomsky, who has been claiming it’s all innate. We know now it doesn’t have to be the case, and Chomsky’s whole view of language is kind of crazy.”⁹ Indeed, people tend to believe everything we know is learned, and they therefore abhor the thought of innateness. This antagonistic sentiment may have helped a recent assault on TGG to rapidly gain popularity (over 30,000 downloads

in early 2025, the time of this writing). Authored by UC Berkeley's cognitive psychologist Steven Piantadosi, it is titled "Modern Language Models Refute Chomsky's Approach to Language," no less.¹⁰ In a nutshell, its main claims are that the success of LLMs dispenses with the need for a symbolic grammar TGG-style; that LLMs "subvert and bypass the entire theoretical framework" of TGG; that the competence–performance distinction is irrelevant to an understanding of the human mind; and moreover, that in the face of LLMs' performance, the Innateness Hypothesis is falsified. This contentious paper thus touches on every item on the list of contrasts above. To bolster his claims, the author uses benchmarks that are quite different from those commended by TGG. The plan here is to examine these points by taking a careful look under the hood of the LLMs at issue.

A Quick First Falsification: LLMs Do Possess Knowledge Not Acquired on Input

From under the hood, things look a bit different. It turns out that LLMs presuppose inescapable grammatical assumptions, as they are not trained "from scratch" as is commonly claimed. Some critical tasks that LLMs are designed to perform are based on part-of-speech labeling of words (and to some extent, word parts). Linguistic labels used are Noun, Verb, Adjective, and other categories we saw in earlier chapters. Obviously, these category labels do not appear in actual texts, like wiki pages, that are used for pre-training. So where does the annotation of training sets with part-of-speech labels come from? Answer: as of yet, it is not possible to realize the dream of "*You shall know a word [only] by the company it keeps*" in full. Therefore, an automatic annotator, or tagger, must rely, in part, on a predetermined inventory of lexical category labels for parts of speech, which is found in manually annotated—and later corrected—datasets (a.k.a. treebanks).¹¹ These are used at the learning stage of tagging algorithms. The part-of-speech inventory is preassigned to datasets by their curators, who decide on the tag set—the set of category labels. For example, the leading Penn Treebank has dozens of part-of-speech labels as well as syntactic tag sets for phrasal categories that its authors invented. The automatic tagging program is trained on such a manually labeled dataset and is then free to automatically annotate texts in a nearly errorless manner.¹²

Do LLMs like ChatGPT use tagged input data directly? Answer: most likely they do not (though this is a question that's a bit difficult to answer), but as far as one can tell, they all piggyback on datasets that used part-of-speech

tagging in the past. Most language generation tasks and, notably, next word prediction and coreference assignment tasks rely on part-of-speech pretraining, which presupposes access to tagged input.

Take coreference assignment as an example. It links pronouns to their antecedents, as previously explained. Figure 4.4 illustrates this with the sentence *The robber who the policemen are holding is giving them his loot*, which contains two number agreement relations with auxiliary verbs (*robber-is*, *policemen-are*) and two pronoun-antecedent relations (*robber-his*, *policemen-them*). For the task to be performed successfully, agreeing verbs have to be tagged for number, and antecedent nouns, for gender, which guides the attention heads highlight them as it scans the context.¹³

Scientifically, then, there is no escape from the inevitable: a task that uses predetermined labels (whether directly or indirectly) injects the system with linguistic knowledge that did *not* come from the input. From an engineering point of view, such tagging is an efficient solution.

There may be an alternative to tagging, but it would require much more pretraining. BERT, a flagship LLM, is pretrained on a collection of texts that holds one billion words (much more than any human being is exposed to in a single lifetime)¹⁴ and still requires some “fine-tuning” with curated, that is, tagged, data to boost its accuracy levels. I cannot even imagine the size of the input dataset that would free LLMs from this burden.

Therefore, if LLMs are a model for human language and its acquisition, their functionality cannot be said to have been built entirely on input. But in humans, knowledge acquired not on input has no way of getting into a system, and it therefore must be in it from its inception, in other words, innate. We seem to be back where we started: Piantadosi’s bold statement that LLMs “subvert and bypass the entire theoretical framework” of TGG is false. Engineers may not care much about this issue, but if we are interested in science, we must be mindful of the scientific consequences of engineering choices. Skeptics may argue that this is a mere technicality that a more advanced LLM will solve. Let’s wait and see; but until we have a concrete solution that we can examine, the Innateness Hypothesis is very much alive. As Gary Marcus has rightly pointed out, many Chomsky bashers have made arguments that barely scratch the surface, as the tests to which they subject LLMs are weak, at best.¹⁵ I return to this issue in chapter 6, when I discuss an oft-heard contrarian argument, which MIT linguist Danny Fox has called the “GPT-*n* objection”: *What would you say if a future version of ChatGPT gets it right?*

B. Competence and Performance: What We Do Versus What We *Can* Do

At the heart of TGG (but not LLMs) is a distinction between what speakers know about their language and what they can actually do with this knowledge. Crucial to TGG—and a major bone of contention—is its underlying claim that the linguistic *competence* we possess (the grammar we *know*) needs an implementation that turns this knowledge into action, that is, *performance* (our *use* of grammar, through our processing system). This distinction helps linguists to set up conditions to test specific grammatical rules and principles we know, while attempting to put the processing device on hold, as it were. We thus attempt to devise experiments that test the boundaries of the grammar under close-to-idealized processing conditions, in which participants (ourselves included) make judgments of plausibility, synonymy, grammaticality, and felicitous meaning and sound shape.

But in the reality of language, noise-free near-ideal test conditions are not easy to find. There are time constraints, lapses of attention, memory limitations, slips of the tongue, and other error-prone behaviors that humans exhibit in reality, which do not necessarily impinge on their knowledge of language. For example, when we hear someone say “I’ve been *drunking* all night” (a famous example given by the late UCLA linguist and speech error collector Victoria Fromkin),¹⁶ we recognize a highly selective impairment to language mechanisms, and do not suspect that the speaker’s grammatical knowledge is wiped out; rather, we attribute the error to a temporary inebriated state affecting the execution of this knowledge. Likewise, we may never be able to fully test the claim that syntactic rules can recursively build arbitrarily long sentences, because we cannot observe sentences whose number of words exceeds the number of seconds in an average human life.¹⁷

LLMs are not built to accommodate the competence–performance distinction. By design, they possess no grammatical knowledge (which suits the assumptions of many cognitive scientists). Indeed, we do not expect LLMs to ever make speech or language errors under conditions in which the processing device, not the grammar, is taxed. In this respect, their errorless behavior will *never* be human. Lest it be said that I am suggesting that you give your computer alcohol before you test it, let me provide a clearer, albeit somewhat more complex, example of a place where LLMs fail to simulate us and where they will keep on failing, or so it seems.

What Computers Overdo: The Embedding Paradox Speaks for the Distinction

Let me illustrate the significance of the competence–performance distinction with a sentence containing a triply embedded object relative clause (a construction that psycholinguists have been preoccupied with for years). We shall get a bit technical, but I believe that the conclusion is worth a reader's effort. In chapter 1, I discussed syntactic embedding of sentences within sentences, e.g., *John said that Bill thought that Mary loves cookies*, which is built out of 3 sentences (*John said that X*, *Bill thought that Y*, *Mary loves cookies*). Relative clauses are another type of syntactic embedding, in which a sentence describes a noun. Take $S_1 = \textit{The man fell}$, $S_2 = \textit{The child pushed the man}$, and $S_3 = \textit{The girl kissed the child}$. You can easily understand sentence S_1 when it embeds S_2 : [$_{S_1}$ *The man* [$_{S_2}$ *who the child pushed*] *fell*]. Note that the object of *pushed* is not there, as it had been displaced by MOVE). You can likewise understand sentence S_2 when it embeds S_3 with the same type of displacement: [$_{S_2}$ *The child* [$_{S_3}$ *who the girl kissed*] *pushed the man*]. But as known at least since the 1950s, we cannot combine the two and expect our interlocutor to quickly understand what we are saying: the triply embedded [$_{S_1}$ *The man* [$_{S_2}$ *who the child* [$_{S_3}$ *who the girl kissed*] *pushed*] *fell*] is very difficult to understand.¹⁸ In all cases, the embedded clause describes the subject (cf. *Which man did it? The man who the child pushed*, etc.). And yet, although the exact same recursive rule of embedding applies in all instances ($NP \rightarrow \text{Noun} + \text{SG} + \text{Sentence}$, where $\text{SG} \rightarrow \textit{who}$), the result of a second application of the rule is not an acceptable English sentence (nor, certainly, is the result if you keep on embedding, as in [$_{S_1}$ *The man* [$_{S_2}$ *who the child* [$_{S_3}$ *who the girl* [$_{S_4}$ *who the dog bit*] *kissed*] *pushed*] *fell*]).

Why can certain rules of sentence embedding be applied once but result in incomprehensible sentences when applied twice or more? Do these observations speak against the embedding rule, perhaps even against recursiveness? Abolishing this rule or somehow restricting its application to a single embedding would leave the puzzle unsolved.

Moreover, here are facts that would fly in the face of this move, and lead to an Embedding Paradox: if you apply an embedding operation on the *object* of each clause (rather than embedding on the subject as above), you are free to do so many times and will still get a well-formed, easy-to-comprehend sentence, such as [$_{S_1}$ *I saw the man* [$_{S_2}$ *who pushed the child* [$_{S_3}$ *who kissed the girl* [$_{S_4}$ *who bit the dog*]]]]. The subject-object contrast that the Embedding Paradox presents is important because it demonstrates that,

while you are free to use recursive rules and embed as many times as you want, the freedom to embed nevertheless has an odd limit that applies only when the embedding is in the subject position of the main clause, not in the object position.

TGG resolves the paradox by distinguishing between the rules and their realization in the human head. It divides the comprehension machine into two parts: grammatical competence, which preserves the freedom to embed everywhere, and actual performance, implemented in a processing device in the human head, which has limitations that stem from architectural considerations such as working memory size. The latter is said to be responsible for the comprehension difficulty in this case. On this view, the rule applies freely, but certain resulting sentences overtax our memory and related devices, leading to a comprehension breakdown, which at times deprives us of the freedom to comprehend. Next, we turn to LLMs, to see whether their performance on the Embedding Paradox is human-like.

ChatGPT and the Embedding Paradox: Success or Failure?

LLMs are said to have no grammar. Therefore, they cannot distinguish between the grammatical freedom to embed and its processing limitations. With this in mind, I tested the Embedding Paradox with the advanced AI chatbot ChatGPT-4o—the “omni-capable” version of GPT-4—stretching it a bit with a query that contained six embeddings of this kind: “*The man whom the girl whom the boy whom the cat whom the dog whom the puppy hurt chased bit kissed pushed cried.* Who did what to whom in this sentence?” ChatGPT-4o quickly peeled the sentence like an onion, listed the actions (without assigning any grammatical structure), and concluded: “Summarizing who did what to whom in simpler terms: The puppy hurt the dog. The dog chased the cat. The cat bit the boy. The boy kissed the girl. The girl pushed the man. The man cried.” Note that the full response throughout is more complex. That is, the full answer must be that *the dog* stands for *the dog whom the puppy hurt*, that *the cat* stands for *the cat whom the dog whom the puppy hurt chased*, and so on. But one should not be overly pedantic here—the machine’s parsing of this complex sentence was quite good.

Why should anyone care about the fact that such sentences are parsed correctly by the LLM? They don’t ever appear in texts or in conversation anyway. Am I not being overly picky? Well, if you want the LLM to exhibit human-like behavior, then here is a case where it does not—perfect

comprehension behavior on sentences with multiple embedding is *not* human performance. Importantly, the Embedding Paradox seems to point to a problem that is unsolvable for LLMs: how to make the machine *weaker* in this specific case, at a time where computational power of chatbots becomes ever stronger, which will likely make the problem more severe. Note that, in the absence of a grammar-based discovery method, this problem would likely never have been detected. Should we condone an electricity-hungry industrial effort that builds such overpowerful devices (which in addition suck up trillions of gallons of cooling water from local wells)?

To borrow a phrase from Chomsky—and as visualized in figure 5.1—the LLM is a supermachine bulldozing its way through an utterance. From the perspective of syntactic embedding, it is plain superhuman. This superfluous strength should fly in the face of psychologists who jeer the competence–performance distinction and cheer ChatGPT. TGG is often mocked as being too abstract and unrealistic. And here we have an important instance of a mismatch between the machine’s success and the human



Figure 5.1

A bulldozer is a machine that can lift heavy weights and push heaps of soil. Humans can also lift weights and push soil. Is the bulldozer a model for human motor skills?

failure.¹⁹ While TGG and the distinctions it makes account for human behavior, LLMs do not.

Treat yourself to a test of ChatGPT with an even larger number of embeddings. I see no reason that the LLM's performance would go down (perhaps up to its current string length limit, which reportedly is over two thousand words).

The abolition of the competence–performance distinction is unmotivated for two additional conceptual reasons that regard abstraction:

I. Think about the addition operation in the equation $a=b+c$. For reasonably small numbers, we can calculate the sum in our heads; for larger ones, we might need a Babylonian abacus; and for yet bigger ones, a supercomputer. But the facts about execution do not change the nature of the operation. The distinction between an abstract relation and its implementation thus cannot be suppressed, as it cannot in the grammatical context.

A similar observation relates to a lesson one implicitly learns in high school physics: through noise reduction by isolation of confounding factors, one creates an idealized version of nature in which everything is removed but the factor of relevance, which can then be carefully studied. In my 10th grade physics lab, we tested Newton's laws of motion on a nearly frictionless air track like the one in figure 5.2. A body traveled on a rail. Air coming from underneath the body through small air holes in the rail minimized friction, allowing us to approximate the idealized conditions that the laws of motion require, as we measured velocity and acceleration. I

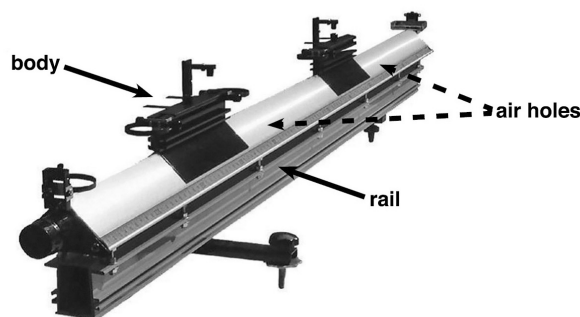


Figure 5.2

An air track that is used to test Newton's laws in high school physics laboratory experiments. Air comes through the holes and minimizes friction, but no one protests that the experimental conditions are not realistic.

do not recall any student rebelling against the idealization and arguing that Newton's laws of motion are false (or at least untestable) because they do not take friction into account. High school science teachers implant within you good principles of theory construction!

II. One of the pillars of cognitive science since the nineteenth century is the Weber–Fechner Law, which expresses the way we (and many other species) make quantity comparisons.²⁰ At present, some neural mechanisms that mediate it in our mind/brain are already known.²¹ It was as clear to its originators then as it is to us today that the law is general, even though it requires adjustments for specific domains—each test condition and cognitive domain (whether visual, auditory, or sensorimotor) requires specific coefficients. However, it is universally accepted that these performance factors by no means diminish the generality of the law, which codifies competence in quantity comparison. It is difficult to understand why among some of those who study language, opposite positions have taken hold.

C. Constraints and Explanatory Power

What makes a theory scientific? This book is not a methodological treatise, so I will answer briefly. A description of a world can be an enumeration of the objects of that world and their properties or an abbreviated list thereof. But a scientific theory of the structure of a world explains why it is the way it is, not otherwise. A shorter description is preferable to a longer one (provided it accounts for exactly the same phenomena) because it is more *parsimonious*; and a description that finds connections between otherwise distinct phenomena is more *general*. But a theory of these phenomena *explains* why it offers the best perspective (“inference to the best explanation”); it provides an account that tells us why the world is structured as it is, not differently. An example might elucidate these points.

The Periodic Table in Chemistry—A Severe Constraint on Possible Elements

In the mid-nineteenth century, chemists were investigating the properties of the elements through the construction of “periodic tables.” Years of deliberation, combined with ingenuity, led to the periodic table high school students use to this very day, which is most strongly associated with Dmitri Mendeleev. The idea is to organize elements in a two-dimensional

matrix in a manner that groups them by properties: atomic weight, metal versus nonmetal, and so on.

Mendeleev was not the first to use a table of elements. Chemists before him had arranged elements by their physical properties, with the aim of defining axes that determine the degree of proximity between related elements. But while they made lists and stopped there, he was the first to understand the table's predictive value. His table, shown in figure 5.3,

ОПЫТЪ СИСТЕМЫ ЭЛЕМЕНТОВЪ.

ОСНОВАННОЙ НА ИХЪ АТОМНОМЪ ВѢСѢ И ХИМИЧЕСКОМЪ СХОДСТВѢ.

			Ti = 50	Zr = 90	? = 180.
			V = 51	Nb = 94	Ta = 182.
			Cr = 52	Mo = 96	W = 186.
			Mn = 55	Rh = 104,4	Pt = 197,1.
			Fe = 56	Ru = 104,4	Ir = 198.
			Ni = Co = 59	Pd = 106,8	Os = 199.
			Cu = 63,4	Ag = 108	Hg = 200.
H = 1	Be = 9,1	Mg = 24	Zn = 65,2	Cd = 112	
	B = 11	Al = 27,1	? = 68	Ur = 116	Au = 197?
	C = 12	Si = 28	? = 70	Sn = 118	
	N = 14	P = 31	As = 75	Sb = 122	Bi = 210?
	O = 16	S = 32	Se = 79,4	Te = 128?	
	F = 19	Cl = 35,5	Br = 80	I = 127	
Li = 7	Na = 23	K = 39	Rb = 85,4	Cs = 133	Tl = 204.
		Ca = 40	Sr = 87,6	Ba = 137	Pb = 207.
		? = 45	Ce = 92		
		?Er = 56	La = 94		
		?Yt = 60	Di = 95		
		?In = 75,6	Th = 118?		

Д. Менделѣевъ

Figure 5.3

Mendeleev's original periodic table.

was rather precise and constrained, and as a result it had empty positions, which is a property that a mere list does not necessarily have.

Mendeleev noticed the empty positions and tried to understand why they were there. His brilliant idea was to realize that these gaps stood for missing elements. Each position denoted a concrete cluster of properties of unknown but chemically possible elements. That is, the empty positions not only revealed that the table was a coherent perspective on the nature of the elements that make up the universe; it was also predictive of elements that had not been found at his time. Such was the story of gallium, whose atomic weight and density Mendeleev predicted and which was isolated a mere few years later. Such too were many heavy elements, which could only be produced and detected in laboratory conditions, due to extremely short half-lives. Without a constraining periodic table, scientists would have had no guidelines that would lead to the discovery of these elements, as nothing would have told them where to look. Yet the periodic table set up clear constraints, which led them to conclude that elements in the empty slots must exist and to go hunting for them in the lab.

These stories make the periodic table easy to accept and be taught in high schools around the globe. But what is it, if not a mere abstraction? It is part of a theory of the *chemically possible*—a major step toward an understanding of the structure of the world.

Constraints on Rules and Relations in Syntax

How does Mendeleev's periodic table connect to rules of grammar? A theory of language seeks to understand the nature of linguistic knowledge, and it is therefore interested in its constitutive principles, just like the periodic table. It seeks this understanding by striving to be general and parsimonious, therefore more explanatory. Like Nigella Lawson's cooking, the linguist seeks to explain more, with less. The quest is therefore not for grammatical rules but for abstract principles that define the notion "a possible rule of grammar." Thus, a theory is explanatory if it successfully excludes all descriptive grammars on principled grounds and at the same time selects the one that describes the empirical phenomena (i.e., that is *descriptively adequate*).²² To illustrate this rather difficult point, I will give two brief examples from TGG and then discuss the LLM approach along these lines.

Example A: X-bar theory and MERGE. In chapter 1, I introduced phrase structure rules, the rules that build basic sentence structure (e.g., Sentence $\rightarrow$ NP + VP;

$VP \rightarrow V + NP$; $VP \rightarrow V$; $NP \rightarrow \textit{John, Bill, Mary, cookies, etc.}$). At no point did I consider why these rules and not others should be chosen or whether there are impossible phrase structure rules, that is, rules that no language could possibly have: for example, $V \rightarrow NP + VP$; $VP \rightarrow N$; $V \rightarrow \textit{John, Bill, Mary, cookies}$. These latter rule-like monsters are not part of the syntax of any human language. But while not incorporating such rules into our inventory of rules is right (in Chomsky's wording, it is right for a theory to be "descriptively adequate"), we have no understanding of *why* these monsters are excluded (which would amount to "explanatory adequacy" of the theory). For that, we need a theory of what a *humanly possible rule* is.

Returning to Mendeleev, his periodic table had the same elements as his predecessors', but in addition, there was his profound insight that "gaps" in the table were meaningful and, moreover, useful. In linguistics, a rule format called X-bar theory does that (albeit in a limited way). It provides a tightly constrained format for all phrase structure rules, one that is tailored to express the intimate connection between the term on the left of the arrow (e.g., VP in the rule $VP \rightarrow V + NP$) and the term(s) on its right (here: V + NP). Details aside, this theory (currently rolled into the MERGE operation) is stated in a highly constrained and general manner, so as to enable the linguist to predict not sentences but *rules*. As gallium was a prediction of Mendeleev's approach, so does X-bar theory have the capacity to point to gaps in the X-bar format, as well as to barriers that exclude monsters such as those above.

Example B: constraints on MOVE. In chapter 2, I described MOVE and introduced a constraint on MOVE, which I called Relativized Minimality. I described transformational rules that help us to understand complex relations, for example, between a question and an answer (Q: *Who does Mary love?* A: *Mary loves John*). These rules relate positions in a sentence (object position and presubject position in this case). I proceeded to discuss the constraint Relativized Minimality sets on this rule: it cannot relate two positions if an element similar to the elements in these positions intervenes between them. For example, given the answer A: *I wonder who could solve the problem in this way*, the question cannot be the ungrammatical Q: **How do you wonder who could solve this problem in this way?* This string of words is an impossible sentence in English (and many other languages) because the MOVE relation that results in the question

(continued)

word *how* being on the left edge of the sentence crosses the question word *who*. Details aside, here, too, Relativized Minimality is a severe constraint on MOVE. Once formulated generally, Luigi Rizzi's move was to pursue the consequences of this constraint elsewhere, in surprising environments. Indeed, he was able to show that Relativized Minimality constrains the English rule of yes–no questions that was thought to be quite distinct from the question rule. Specifically (as noted in chapter 3), he discovered that the same generalization explains the contrast between the grammatical *Has the student been sailing?* And the ungrammatical **Was the student have sailing?* This connection has since become central.

Constraints, then, are at the heart of scientific theory and scientific discovery. In a sense, they provide a bird's-eye view of the scientist's daily toil and focus her attention on the best direction in which to proceed with her inquiry. Constraints on a theory also lead to testable predictions, as in the two syntactic examples I just presented.

In LLMs, It's All About More, More, and More

Can such constraints be found in LLMs? The answer is both *yes* and *no*. *Yes*, because LLMs are subject to hardware and run time constraints—a component that is not in keeping with the specifications of the machine on which it runs will either never get off the ground or start and crash. At times, machines are also subjected to biologically motivated learning constraints. This is true of convolutional neural networks, where interlayer connectivity is restricted, an architectural feature said to constrain the range of the learnable.²³ A possible additional example is provided by certain convolutional neural networks that deploy “constrained optimization” by setting limits on the range of image labels, as well as on other parameters.²⁴ There are examples of constraints that have been set up in order to fit properties of animal and human visual object recognition. In language, one finds very little of that.

The development of network architectures can be described as a sequence of additions and augmentations that improve performance, but in which language processing has until recently played a minor role. First, as I described in chapter 3, Rosenblatt proposed his perceptron as an input–output interconnected network; later, as problems were found, the concept of hidden layers was added by Minsky and Papert, who showed that such a perceptron could solve the XOR problem (which Rosenblatt's had not been

able to solve); this opened the way to a host of feedforward architectures, in which connectivity and layer size could be experimented with, with no limits set on size and type; later, the idea of recurrence was added, further empowering the network. Then, networks with multiple hidden layers were coupled with AlexNet, a vast dataset of 1.2 million images collected from the web and manually categorized with one of 1,000 predetermined labels; deep networks emerged, which paved the way for the addition of LSTM and later attention components, as I described in chapter 4; finally, the quest for language modeling led to a huge increase in the capacity to represent the context, combined with multiple attention “heads” and bidirectionality, and gave us LLMs.

These are truly genius engineering developments, made possible by breathtaking, ever-growing hardware inventions. And yet, the construction of these powerful machines is not constrained by anything but what the hardware enables and the vast amounts of energy these machines require. And so, even if taken as scientific theories, LLMs fall short of explaining why the world is structured the way it is, because they are often unable to capture generalizations. Add to that the unconstrained size of sets that LLMs use for pretraining, which no human can spare enough time to process, and the humanly impossible attention mechanism, which can scan thousands of words that are stored, and you are back with Chomsky’s bulldozer metaphor.

Runnability Versus Transparency and Interpretability

An oft-mentioned advantage that’s attributed to LLMs is that they mostly meet a “runnability” constraint.²⁵ That is, they are actual programs that run on a computational platform and produce output. This property may be instrumental in making them “predictively adequate,” allowing tests of novel predictions. Predictiveness of this kind is an especially important feature of devices whose internal structure is non-transparent and poorly understood. By their developers’ admission, most LLMs are not “interpretable.” That is, the LLM is so complex that it is not always clear which part of it is responsible to which function, which has led to a host of “interpretability” efforts, attempting to be more analytic.

Linguistic theories are not always “runnable”, in that they do not easily lend themselves to be implemented as computer programs. However, these theories tend to be modular and transparent. The internal structure of most linguistic theories—and consequently, of your typical research paper in

phonology, syntax, or semantics—consists of an articulation of a problem, a proposed theoretical solution, typically in the form of a set of grammatical modules that interact in ways that are relatively well understood. This structure and understanding lead to crisp predictions that are derived and reasoned, and lend themselves to a host of empirical tests. If an alternative theory exists, a comparison between it and the proposed one is in order. Thus, making a theoretical claim and then immediately deriving testable predictions is part of the daily toil of a (good) linguist.

This contrast in internal structure between the approaches could be seen earlier in this chapter, when I discussed the Embedding Paradox and pitted LLMs against the theory of syntax. The former was tested via ChatGPT-4o, whereas predictions of the latter were easily derived by looking at the structural analysis of the sentences at issue. We were able to pit the predictive adequacy of the two approaches even though only one of them was “runnable”.

The runnability requirement may lead to further problems. Requiring theories to be runnable runs the risk of arbitrary, ad hoc technical assumptions concealed in the code that maps the theory onto computer action. We may call this danger *the Implementation Bias*, which pushes the quest for a running program to unintended, at times even unnoticed, ad hoc moves. Here is an example. Hebrew University’s AI expert Yair Weiss and his colleagues examined the inner workings of certain programs for image identification. They discovered that tiny modifications of the method for image translation (e.g., shifting the crop by one pixel) lead to sharp performance drops in image classification by certain machines. In the cases Weiss and colleagues studied, runnability or lack thereof depends on ad hoc actions. Indeed, some of the programs they scrutinized had disappointing learning performances: a machine that was trained on 400 million images failed to learn that a tiny change in crop does not modify the image.²⁶ Such demonstrations point to a serious gap in LLMs’ ability to generalize and transcend the specific engineering decisions that are made by designers.

In sum, a modular theory may be predictively adequate if its structure is transparent, and the functions of its pieces (or modules) can be analyzed, as illustrated by the discussion of the Embedding Paradox; at times, the eagerness to implement may lead to unintended consequences that might blur, rather than sharpen, a scientist’s vision, as Weiss et al. demonstrate.

Coda: A Quick Lesson from Chess and from Weather Forecasting

An international chess scandal erupted in the summer of 2022: Magnus Carlsen, the world champion, withdrew in the middle of a game with rising star Hans Niemann, and he tweeted a link to a video clip in which a famous soccer manager can be heard saying, “I prefer really not to speak. If I speak, I am in big trouble.” A subsequent investigation by chess.com resulted in a 72-page report, which concluded that Niemann “has likely cheated in more than 100 online chess games, including several prize money events.”²⁷

Central among the many criteria that chess.com used for the determination that Niemann had been cheating was the degree of similarity between his moves in 100 online games and those recommended by a “chess engine.” Niemann’s moves were too similar to the algorithm’s, “strongly suggesting that he violated our fair play regulations.” For chess.com, simulation-like human performance is not a sign that the program is a cognitive theory but just the opposite: it is an indication that the human is cheating, because it is clear that, unlike a scientific theory, the program is unconstrained. Indeed, when Deep Blue, in 1997, beat Gary Kasparov (figure 5.4), the then-reigning



Figure 5.4

Gary Kasparov losing to Deep Blue, 1997. Photograph by Najlah Feanny, licensed via Getty Images. © Najlah Feanny/CORBIS SABA.

world champion, no one thought Deep Blue was human. It is this event that led Noam Chomsky to comment that Deep Blue defeating Kasparov in chess was as interesting to him as a bulldozer winning the Olympics in weight lifting.

Consider now a very different research area—that of weather prediction, where LLMs outperform physics-based forecasts. For years, weather prediction has been done by a sophisticated numerical system called the Integrated Forecasting System, “which involves solving the governing equations of weather using supercomputers. . . . [Numerical weather prediction] methods are improved by highly trained experts innovating better models, algorithms, and approximations, which can be a time-consuming and costly process.” A recent alternative has been machine learning–based weather prediction, “wherein forecast models are trained from historical data, including observations and analysis data.”²⁸ It has been shown to improve forecasts by “capturing patterns in the data that are not easily represented in explicit equations,” and “exploiting modern deep learning hardware, rather than supercomputers, and striking more favorable speed–accuracy trade-offs.”

Weather forecasting is an area that is dedicated to prediction, not explanation. Recently, machine learning–based prediction systems have outdone traditional numerical methods which solve thousands of equations. Yet the creators of the machines understand their limitations and they emphasize that theirs is not a physical theory but an unconstrained simulation that works well: “Our approach should not be regarded as a replacement for traditional weather forecasting methods, which have been developed for decades, rigorously tested in many real-world contexts, and offer many features we have not yet explored. Rather, our work should be interpreted as evidence that [machine learning–based weather prediction] is able to meet the challenges of real-world forecasting problems and has the potential to complement and improve the current best methods.”²⁹

In both chess and weather prediction, then, systems are built that may outdo constrained scientific ones. Yet their designers understand that engineered but unconstrained systems may lead to prediction accuracy but not necessarily to an explanatory scientific theory.

We should therefore wonder: given that LLMs applied to natural language are unconstrained, is there a reason to think of them as scientific theories? Is it possible to characterize a component or mechanism that

cannot be part of an LLM? For example, what exactly is the scientific status of the ad hoc, postlearning “fine-tuning” operations (i.e., the addition of new input data) in some of the LLMs to improve performance? These and other questions arise for a reader of the current LLM literature.

Still, one hopes that these models can somehow be linguistically constrained, so as to make them cognitively more realistic. In an effort to get closer to this goal, let’s now move on to examine concrete problems with LLMs’ performance on language tasks.

Taking Stock

This chapter was the first of two to get under LLMs’ hood. We saw:

- Some assumptions that are common to all formal approaches to language
- The falsity of a claim that LLMs’ stellar language behavior “refutes” the Innateness Hypothesis
- Problems that arise once the competence–performance distinction is blurred: LLMs turn out to be too strong, in exhibiting superhuman behaviors in parsing multiple embeddings “correctly,” thereby failing to emulate human behavior on the Embedding Paradox
- The supreme value of constraints in science—the periodic table in chemistry
- Constraints in syntax and how they lead to deeper understanding and new discoveries
- That deep learning has few, if any, scientific constraints, which allows arbitrary decisions
- That deep learning models may meet a runnability requirement, but this requirement may only be due to their uninterpretability. Runnability moreover may lead to an Implementation Bias
- That linguistic theories are typically transparent and modular, hence interpretable, though they may not always be runnable
- That developers explicitly recognize that the predictive success of LLMs may not argue for their validity as a scientific theory—whether of chess cognition or of global weather

This is a section of [doi:10.7551/mitpress/15932.001.0001](https://doi.org/10.7551/mitpress/15932.001.0001)

How Deeply Human Is Language?

Chomsky, the Brain, and the AI Fantasy

By: Yosef Grodzinsky

Citation:

How Deeply Human Is Language?: Chomsky, the Brain, and the AI Fantasy

By: Yosef Grodzinsky

DOI: 10.7551/mitpress/15932.001.0001

ISBN (electronic): 9780262052023

Publisher: The MIT Press

Published: 2026



The MIT Press

The MIT Press
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139
mitpress.mit.edu

© 2026 Massachusetts Institute of Technology

All rights reserved. No part of this book may be used to train artificial intelligence systems or reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

The MIT Press would like to thank the anonymous peer reviewers who provided comments on drafts of this book. The generous work of academic experts is essential for establishing the authority and quality of our publications. We acknowledge with gratitude the contributions of these otherwise uncredited readers.

This book was set in Stone Serif and Stone Sans by Westchester Publishing Services.

Library of Congress Cataloging-in-Publication Data is available.

ISBN: 978-0-262-05200-9

EU Authorised Representative: Easy Access System Europe, Mustamäe tee 50, 10621 Tallinn, Estonia | Email: gpsr.requests@easproject.com