ABABABA-Grammars: The Shallow Computation of AI.

<>

How linear-capture, Frequency and Semantics fail to support a recursive linguistic mind.

Joseph Galasso https://orcid.org/0000-0002-8426-6281

(revised draft: 3-11-25)

Let's utilize as our point of departure the classic Berko's 'Wugs Test' (1958) which forced us to reexamine previously held assumptions regarding linguistic theory:

(1) firstly, by dismantling the very naïve theory of language acquisition and usage as mere *imitation* [X] > [X] (sound to meaning);

(2) secondly, by weakening imitation only to replace it within similar behavioristic assumptions based on *analogy* [W[XYZ]]>[V[XYZ]] ([_[ing]]>[_ [ang]]>[_ [ung]] (sing, sang, sung > *bring>brang>brung) (cf. and therefore attempting to explaining-away the Wugs test by simple sound analogy: [_ [ug]] ({-Pl}> [_[ugs]]) {+Pl}, ([b[ug]]> [b[ugs]]);

(3) thirdly, to ultimately delivering a generative computational assumption of language X+Y=Z (e.g., [N] + {s} = +Plural (etc.), this latter analysis being free from +Frequency sensitivity and/or semantics.

It was at this final evolutionary phase of linguistic theory, brought on by the newly devised generative framework, that the word 'computational' started to be bandied about. But what exactly does 'computation' mean? Sure, the word computer comes to mind, but all that alludes us to is some form of reasoning or reckoning. David Marr (see Endnote at appendix) (one of my most favorite people, though who regrettably died much too early in the mist of his most creative/genius research on cognition, vision and their neurological bootstrapping)—once expressed the distinctions between the broad term of *computation* to the more narrow and interesting term *algorithm* by his use of the 'cash-register' analogy:

Cash-Register Analogy.

(i) The computation of the **cash-register** is to simply add: a 'logical-and' computation which forces a linear-grammar [ ]+ [ ]+ [ ] (see ABABABA-grammar below).

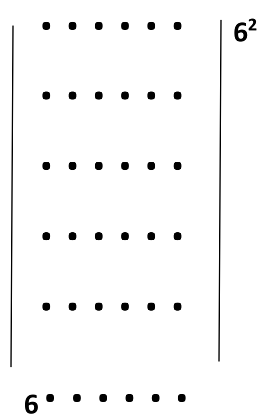
Note*: 'logical-and' : [an enemy] **and [an enemy] = [an enemy]
Or [-6] plus [-6] = [-12] (keeping the recurrent negative [-] operation).

(ii) However, the real interesting aspect is that a cash-register has a number of Algorithms it can avail itself to in order to count. Let's understand the term Algorithm here as a distinct manner of computation. For instance, a rather inefficient manner would be to express 'logical-and' via binary mode: 1=1, 10=2, 11= 3, 100=4 (1000=8) etc. Notice that by occupying three spaces (from right to left) only gets you to four (4). Compare that to the algorithm we are all used to, what is called 'base-ten': move 'vertically' digits 1-9, then move 'horizontally' once to the right (1-9> 99> 999). Note that with the same three spaces from right to left we can compute 999 (close to a thousand), as compared to binary's three horizontal spaces which yields four (4). This is quite a quantum-jump of computational power and the sole difference is due to the type of algorithm in use.

(iii) We can use a notation here to express the difference: while binary is limited to 0 and 1's, thus limiting its 'horizontal-capture' (horizontal in the sense of rules: X+Y=Z (as opposed to iconic 1-to-1 vertical-capture). (Note: perhaps the most iconic counting algorithm would be a strict '1-to-1' as partially found in the first three generations of Roman numerology one-through-three (I, II, III (but where IV (four) starts to take-on *symbolic representation*). Hence, the use of Algorithm is the essential question in determining the generative power of computation.

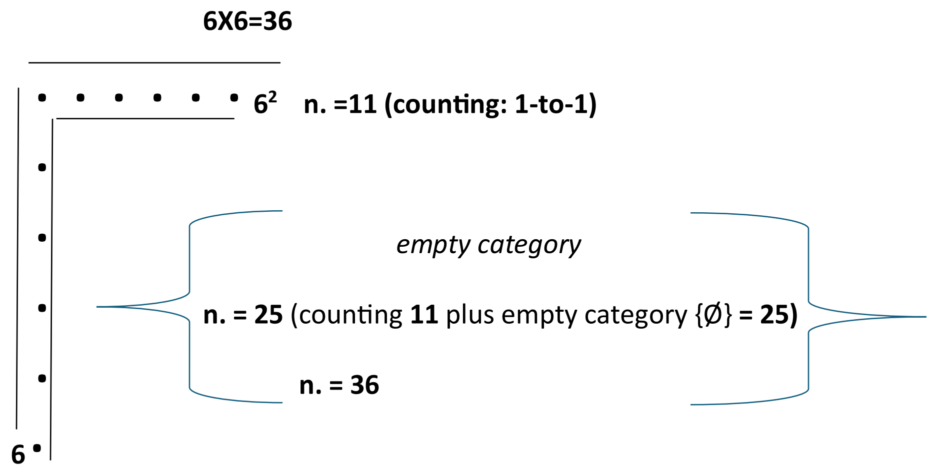
(iv) Now let's compare the algorithm **'logical-and'** (an addition X+Y algorithm)' with that of a **'logical-of'** algorithm (the latter which is recursive in nature and which contains an multiple-operator effect).

Note here that such multiplication, or square-root (6^2) algorithm assumes *empty categories* {Ø}, categories which don't make-up real iconic 1-to-1 expressions. Let's consider the counting example below as an exercise in square-root recursive expressions (with embedded empty categories—a neither 'here-nor-there' entity):



6X6=36

Taking this example further, it is possible to use a ‘logical-and’ algorithm (in an extreme case, to use the first generation of roman wedges (i, ii, iii, iiii, …) and to count each point in turn (you can imagine that with large numbers, we would be here all day—in fact, potentially taking up multiple lifetimes). But now, let’s turn to a different algorithm altogether: that of ‘logical-of’, which requires the aforementioned embedded empty category:



The above expression, which utilizes the linguistic concept of an empty category {Ø} is a recursive/embedded algorithm. (empty in the sense that we don’t need to count, even realize the addition 25 wedges contained in the category, only its symbolic representation of ‘25’).

*Note how this differs from a mere ‘logical-and’ algorithm when addition is turned into a ‘logical of’ operation as found in multiplication (6 **of** 6, or 6^2 (as opposed to 6 **and** 6)). Consider the famous *Realpolitik* express below:

> [and enemy [of an enemy]] = [a friend] (where the negation operations gets folded and turned onto itself hence cancelling out the negative and rendering it positive, as found in [-6 [x -6]] = [+36] ([[ ]]-recursion).

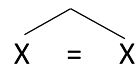
It seems that Artificial Intelligence (AI)—while exponentially vast in its speed and memory-computational capacity—can only resort to mere narrow ‘linear-derived’ grammars (what we term ‘linear-capture’, as notated by ABABABA-computation (AB^n)). By resorting to recurrent ‘logical-and’, a non-recursive algorithm (similar to what we might expect of a language solely based in linear adjacency), we quicky uncover *catastrophic* grammar/language errors made by AI (reminding us that AI is a *formal, non-natural language*, recurrent in its underpinning, and unforgivingly selfish of its brute force sensitivity of frequency and semantic association).

We’ll turn to AI errors just below (see Note-D at appendix). But before that, let’s take one last look at David Marrs’ analogy, this time by using the telephone-call analogy.

Telephone-Call analogy.

Imagine a telephone ringing at your home—so, you pick-up the phone to answer. This is a classic 1-to-1 analogy which equates to our X=X iconic/imitation computation (which uses a classic 'logical-and' algorithm). This equates to Stimulus & Response (S&R) of the utmost Behaviorists persuasion.

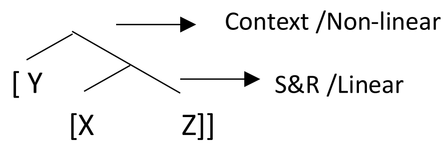
(i) Phone rings = pick it up



Brain area: (PMC: Premotor cortex, Temporal lobe (Wernicke's area): This can be captured by linear-derived grammar.

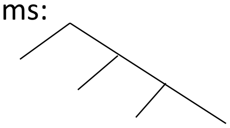
But now notice what happens when the *context* changes and the phone rings at your friend's house: you don't pick it up. Rather, it seems a non-linear grammar of context overrides your previous linear grammar.

(ii) *Phone rings at your friend's house:*



Brain Area: (Brodmann's (BA) area: 44, 45/47). This context-based non-linear algorithm is called 'Contextual' and triggers more frontal-lobe activity.

At this juncture we can see the classic divide between temporal and frontal lobe processing. However, we are not yet done; Let's now imagine that your friend asks you to puck-up the phone whenever it rings and to take a message for him since he will be busy finishing an urgent email. So now you are twice removed from the S&R 1-to-1 computation, and you are prepared to pick-up the phone. This is referred to as an *Episodic* computation that utilizes double recursive branching (which can be the extended in turn by a subsequent request by your friend not to pick up the very next call since he is expecting a call from his girlfriend, but that for all subsequent calls, her maintains permission to pick-up the call (these are so-called branching algorithms: [V [W [Y [X, Z]]]]



Brain areas: (BA 10), Branching shows most abstract control. As we advance up the multi-branch hierarchal syntax tree, brain activity gets more and more abstract, progressively moving from Brodmann's area [44], to [45-46], to BA [10].[1]

[1] Koechlin et al. (2003) delimit areas of BA 44 (Pars opercularis), BA 45-46 (Pars triangularis), and BA 10 (Pars orbitalis) to context, episodic and branching controls respectively.

Recursive embedding of 'context-episodic-branching' controls as part of the language faculty has recently become the one essential ingredient in establishing the definition of what constitutes 'human language' (Hauser, Chomsky, Fitch, 2002)—namely, recursion: that quintessential phenomenon which separates animal communication from human language, stage-1 child utterances from full adult syntax, MERGE operations over MOVE, and human-abstract rules found in the human mind vs Deep-Learning/AI algorithms:

> Why child stage-1 cannot discriminate between the expressions 'boat-house' vs 'house boat' (the former a kind of house, the latter a kind of boat) (see (6) below);
>
> Why regular rule formations such as the prosaic plural {s} as found in English remain productive over an array of *nonce* (never-heard-before) items (Berko's classic 'Wugs test'), and why irregulars must rather be memorized (via reinforcement: Stimulus & Response);
>
> Why Move-operations provoke a non-frequency-driven recursion of the [[ ]] type, while Merge relies on frequency of item for brute memorization..

Some argue that recursion is a recently adapted by-product of a newly emergent human brain, perhaps having arisen as recently as 40KYA (thousand years ago), and perhaps the one feature which gave Cro-magnum an edge-up (in software) over Neanderthal.

The following chapter examines some basic issues surrounding the theme 'Recurrent vs Recursive' within a maturational/developmental progression. The paper examines syntax, long-distant dependency, recursive design, the language-brain mapping, the role of rules {x+y=z} in the underlying grammar, as opposed to frequency-driven adjacency in artificial intelligence, and language types.

One could argue that the ultimate underlying structure of language, that which undergirds much of syntax, can be abstracted and reduced to a dual-template scaffolding scheme which governs a dual processing of language: (1) one mode of processing which assigns a 1:1 [sound : meaning] association, establishing the lexical-itemization of vocabulary, (2) one which provides a variable-slot for otherwise abstract, grammatical relations. The former being described here as a 'sister-sister' relation, (often termed in computer-coding lingo as an ABABABA-grammar), the latter as a 'mother-daughter' relation necessary for hierarchical/recursive design. Sister-relations are vertical in nature (a laundry list of sorts) which provides no other syntactic material than what the label of the two terms describe semantically—e.g., the Verb Phrase [VP [V drink] [N milk]] is the result of a mere sister-sister **MERGE** operation which provides the semantics of √DRINK and √MILK. This type of flat production is what we find in early child syntax (e.g., Mommy drink milk, Boy bounce ball) and is also consistent of an early stage where mixed word orders manifest (a stage-1 production approx..18-36 months of age (the so-called 'Lexical stage-1'):

(i) [VP drink milk] [VP bounce ball] No order since both are sister-relations {X, Y : Y, X}
(ii) [VP milk drink] [VP ball bounce]

One of the first instances of hierarchical syntax, an instance of **MOVE**, comes with fixed correct word order within a phrase (English VP: Verb+Object order), such that movement of one of the two merge items dominates the other and becomes labelled as the Head of the Phrase (the so-called Labelling theory). The Head of the VP √DRINK moves up to a higher hierarchical position in order to dominate its complement √MILK:

(iii) [VP [drink [~~drink~~ milk]]

The template for such movement, as we discuss throughout this paper, requires a mother-daughter/hierarchical tree diagram, as drawn below:

Type-A Type-B

(iv) a. Sister-sister {X, Y}

[X Y]

b. Mother-daughter {Z {X, Y}}

[Z [X Y]]

a. Sister-sister {X, Y}

[drink milk]

b. Mother-daughter {Z {X, Y}}

[drink [~~drink~~ milk]][2]

(vi) b. Mother-daughter {Z {X, Y}}

[House [Boat boat]]

'House-Boat' (= a kind of Boat, not kind of House)

[House [~~House~~ boat]] *(see (6) below).*

In fact, one could argue that the very essence of language is of the (b)-type 'mother-daughter' relation which provides recursive syntax. Hence, Language narrowly defined is type-b, with type-a only providing potential itemizations of words into basic flat two-word phrases (without fixed word order).This fact alone opens up question regarding the brain-to-language mapping (How does the brain provide Type-B?, when did humans developed the type-b template, and the question how does this new NARROW (Type-B)

[2] Note it's the movement of the lexical item (creating 'mother-daughter' hierarchy) which enables the phrase to be labelled (as Verb Phrase, etc.). Examples in (vi) such as [boat-house] vs [house-boat] (the former a kind of 'house', the latter a kind of 'boat', pose (ambiguous) problems for young children regarding interpretation. For a full discussion, see papers: https://www.academia.edu/5761528/Lecture_1

definition of language change our understanding of language as BROAD (Type-A) (e.g., early child syntax (which shows only initial Type-A prosaic structures, animal communication).

It seems the human brain/mind is unique in its capacity to move from (i) a *recurrent* mental-processing through to (ii) a *recursive* mental-processing. Some scientists argue that this 'uniquely human-species-specific capacity' has emerged on our evolutionary scene as recently as 40KYA (thousand years ago). While there may be more general-cognitive and learning schemes tethered to such recursive processing (e.g., theory of mind, declarative vs procedural knowledge, etc.), on a pure linguistics footing, this recurrent + recursive progression defines what we find in the two stages of child language syntax—whereby a recurrent stage-1 manifests primary base-lexical stems (as well as the stacking of such bases), while the recursive stage-2 manifests movement-based operations (what was once termed the classic Lexical vs Functional dual stages of child syntax).

Recall, one very simple example of the functional vs lexical distinction can be found in our 'How *do* you Do?' example, where the first functional *do* constitutes an Auxiliary Verb (something like a *light verb* √*do*)—Aux verbs are **'category-based'**, provided merely for an interrogative/question syntax, are non-substantive in nature, and can be readily deleted in spontaneous speech (e.g., How _ you DO?)—while the second DO is a lexical substantive main verb and can't endure deletion (e.g., *How DO you__?). Main verbs are **'Item-based'**. **Items** are base-lexical words such as [*nouns, verbs, adjectives, adverbs*] which carry semantic meaning and can readily be *stacked* on top of each other (in a vertical way) in forming basic lexical phrases (VP drink milk), [NP mommy sock], or in forming compounds [N back-bird], [N chain-smoker] [N boat-house]. (See discussion for 'chain vs cigarette' smoker below). **Categories** are abstract, functional 'Edge'-properties of non-substantive value—they form the Specifier position of X-bar theory and host otherwise moved items displaced from lower down in the syntactic tree and they encode for non-semantic words which are manipulated to derive a movement-based syntax (e.g., *Question* (Aux inversion), *Tense* (subject-verb agreement, affix hopping), *Case* (Subject +Finite verb relation), *Structure* (The => N (DP), fixed Word Order, etc.).

https://www.academia.edu/34403440/Working_Papers_5_Minimalist_Perspectives_on_Child_Syntax_Merge_Over_Move_Movement_Application_in_Child_Syntax

Flat-item bases (lexical items) can be pulled directly from out of the lexicon (vocabulary) and can via *Bricolage* (i.e., a structure-building process) build-up (stack) 'one item on top of another' in forming nonmovement-based lexical phrases: √drink + √milk = [VP drink milk],

√mommy + √sock => [NP mommy sock], √black + √bird => [Adj Black bird]. etc.

These base ↑ items fall along the **Vertical Axis:**

<d>…

<c>,

<b>,

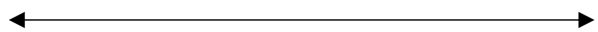
Items: <a>, ↓

Examples of **flat** ‘directly-pulled’ **items** is what we find for **recurrent** (logical-&) operations:

e.g., I need to buy apples (a), bananas (b), carrots (c), doughnuts (d) (where the comma returns after each variable back to the main proposition: I need to buy(_): (return): I need to buy(_),…etc.

Notice, regarding recurrent linear arrays, no inherent structure is required: *Hierarchy* doesn’t arise out of recurrent arrays. For example, the string read like a backwards laundry list: ‘I need to buy: c, a, d, b…’ works just as nicely. (Another example of ‘Logic-&’ having no hierarchical word order is the structure: ‘I love [Mom & Dad] = I love [Dad & Mom]. Such combined items are referred to as *Merge-based*.

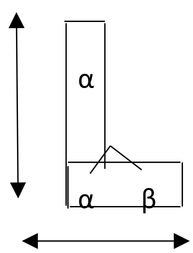
While the **Horizontal Axis** allows for movement (displacement) and the ‘spreading of rules’ which renders a **categorical** syntax:



Categories: <d, <c, <b, <a>>>>…

This two-prong axis is what we find regarding the two stages of child syntax:

Merge



Move

This dual pathway is expressed as Merge {a, b} which gives you an unordered SET, while Move provides a hierarchical PAIR: < α, <α, b>>.

(See Appendix B for vertical/horizontal brain mapping).

(* Note. Phonology straddles the flat vertical axis (combine: <phoneme + phoneme> is only 1-Deminsional in processing), Syntax and the horizontal spreading of Rules (X+Y=Z) maps onto at least 2-Deminsions).

A more formal way of expressing this dichotomy is by the mathematical expression:

$\mathbf{(AB)^n}$ This yields a logical-& flat **recurrent array**: [AB], [ABAB], [ABABAB], [ABABABAB]…

So-called ‘ABABABA’-Grammars.

$\mathbf{(A^n)\ (B^n)}$ This yields: [A[A[AB]B]B] (a **recursive, embedded structure**) (i.e., Russian Nesting Dolls).

Note below how a recursive processing must keep a record (index) for each matrix pair $[A_i[A_k[AB_j]B_k]B_i]$.

It's this kind of index mapping (trace-theory) which relies on movement analogies in syntax. The fact that very young children lack movement (lack inflectional morphology) can be accounted for by the neurological maturation and late onset of Broca's area of the brain—the area responsible for movement.

Recursive Indexing. An example of how recursive indexing works syntactically can be seen in the following examples which require trace-operations. Consider the 'look-ahead/trace' index features which requires the [+/- Definiteness] feature on a Determiner (*A, The*) to select its proper verb (*are/is*) matrix. Ask yourself which of the two verbs correctly applies to the matrix Determiner:

a. A number of students is/are dropping.
b. The number of students are/is dropping.

You quickly see that part of your inherent knowledge of English grammar relies on a recursive syntax of the $[A_i[A_k[AB_j]B_k]B_i]$ type: where indexing of an element must be able to **move** as a *PROBE* across an array of lexical items to link up with its *GOAL* relation (Probe-Goal Union). In other words, adjacency is a not a governing principle for syntax of the [AB], [ABAB]-type. Rather, movement, even at times 'distant-movement' is required in order to achieve proper syntactic values. Note the distance between words that such indexing must cover—clearly, trace-theory is not an adjacency operation, but rather is reliant on movement at a distance (See Appendix D on how our most advanced Deep-Learning/Artificial Intelligence algorithms—in addition to how young children, and aphasics (below)—suffer catastrophic failures when dealing with such distant operations).

c. **A** number of students are/*is dropping. (plural verb 'are' indexes with [–Def] 'A')

d. **The** number of students *are/**is** dropping. (singular verb 'is' indexes with [+Def] 'The').

Note: Broca's aphasics having similar problems whenever movement is required:

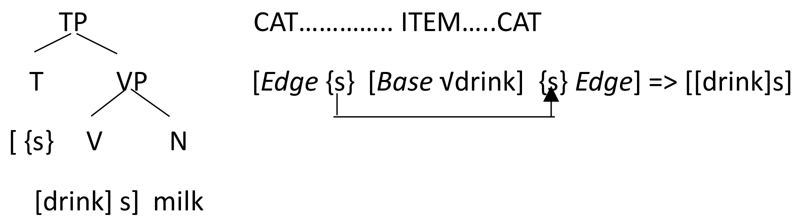
(i) *The girl that the boy chased is tall.* (Who is tall?) Aphasics have a hard time processing such move-required operations. The typical Aphasic (non-correct) reading is that 'The Boy is tall'.

(ii) *The apple that the boy ate is red*. (What is red?) Such questions don't pose problems.
In (i) syntactic recursion is required: [The girl [that the boy chased] is tall]
In (ii) semantics helps since only 'apples' are red, and not 'boys'

The human brain/mind has access to both modes of processing. It is an interesting fact that the onset of child syntax mimics what we might understand to be evolutionary sequences of mental processing found amongst our species (‘ontogeny-recapitulates-phylogeny’). There’s a strong consensus the any putative form of a proto-language early-on in our ancestral evolution would have most likely been confined to flat item-based stacking of the $(AB)^n$ recurrent type.

In such recursive structure, movement of an item into a CATegory position is in operation. We shall call such category positions the **Edge** of a base-word: [Edge [Base] Edge]: noting that it is the **Edge position** which can host movement (affix) elements (e.g., (i) the Tense Phrase [T {s} [V speak] s] or (ii) ITEMs (e.g., (ii) WH-movement and Auxiliary inversions of ITEMs ‘Where’ and ‘DID’ respectively):

(i) [TP Mary [T {s} [VP drink]s] [N milk]]



(ii) [Where did [John ~~did~~ buy his car ~~Where~~]]?

(* Note: the sentence ‘How do you do?’ has the same structure [How do [you ~~do~~ do ~~how~~?]].

The following examples show this two-stage processing:

(i) <u>(Vertical) Recurrent Lexical Stage-1</u> (@18-30 months) where only base-lexical words/items are employed in a flat vertical processing (potentially without a fixed word order),

(ii) <u>(Horizontal) Recursive Functional Stage-2</u> (30+months): a stage where movement operations, the spreading of rules, affix hopping etc, are performed along the horizontal axis.

Examples of Recurrent versus Recursive syntax.

[1] **Small Clause**

Mommy ….‘Him do it’ (24 month-old child utterance)

Note how the small clause is without movement when considered in its full syntactic context:

[I saw [him do it]]…

where ‘I saw’ is the finite main clause (where both Tense (Past), and Case (Nominative “I”) are in operation).

When we consider the **edge position** [*Edge* [*base*]] of the **base-**utterance 'him do it' shown as… [*Edge* I saw [*Base* him do it]]…

…then we can make-out how there is no movement regarding the small clause base-words [him] [do] [it]. (The verb phrase here in fact may be processed idiomatically as a single, unsegmented item [doit], as instance of so-called 'chunking' in early child syntax).

If we were to move the base words into an open edge-position pretraining to the functional Finite Phrase, then we would yield the finite clause [He does it] which incorporates Tense (present affix {s}), Case (Nominative 'He'): A Finiteness-Effect (Tense, Subject-Verb Agreement, Case, along with fixed word order) seems to be the differential between a recurrent stage-1 and recursive stage-2 (whereby stage-1 suffers from a lack of Tense and Case). One other way to put this is that **stage-1 lacks movement** (viz., the lack of INFLectional morphology is a hallmark of early child syntax).

[Recursive……[Recurrent]]

[Edge {+Finite} [Base {-Finite}]]

[2] **Root vs Synthetic Compounds**.

Notice that compounding too can entertain recursive movement operations, as witnessed in root vs synthetic compounds. Consider the differences below between [+Move]/Synthetic vs [-Move]/Root compounding:

Root Compounds: 'Chain-smoker'

(i) *[Chain-[smoker of chains]] (*recursive). => (Not correct structure).

Note how a 'chain-smoker' is NOT a 'smoker of chains'.

The correct structure is recurrent (not recursive) and only operates on non-moved based items:

[Base + Base]

(ii) [Chain-smoker]

Synthetic Compounds: 'Cigarette-smoker'

(iii) [Cigarette [smoker of cigarettes]]

The above synthetic compound has a recursive structure which employs both the base/Items (words) as well as the Edge (move-based slots): [Edge [Base]]:

[*Edge* cigarette [*Base* smoker (of) ~~cigarrete~~]]

Note: This is the same underlying processing we find with the distinction between 'wine-bottle' vs 'bottle of wine': [bottle of [wine ~~bottle~~]]. Children at the recurrent stage -1 only have access to 'wine -bottle' modes of flat item-based processing. It is not until the recursive stage-2 that we find movement regarding the possessive {of}. Hence, stage-1 = Base, stage-2 = Edge (where Base = lexical/ITEM and Edge = Functional/CATegory).

[*Edge* Bottle of [*Base* wine ~~bottle~~]]

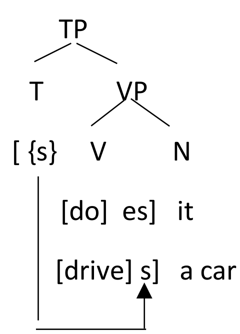
(Note: Examples of *functional* words are the *determiner* class: *A/The, This/That, Each, Some, Every, ALL,* etc. etc., all of which structurally introduce a Noun: D, N = DP. In this way, Determiners mirror what the Auxiliary does for the Verb: both are seen as CATegory-based and serve similar functions as they both instigate movement and drive the motivation of AGReement).

[3] **Tense**

This same edge-feature shows up with Tense (affix-hopping) of Present affix {s}, Past {ed} etc.

Recurrent Stage-1: Him do it. Daddy drive car. Yesterday, I play ball.

Recursive Stage-2: He does it. Daddy drives a car. Yesterday, I played ball.



[CAT………[ITEM]……..CAT]

[edge {s} [Base √do] [edge {s}] (spelled out as {es}, 'does')

TP is move-based/recursive: Tense {s} affix lowering [[drive]s].

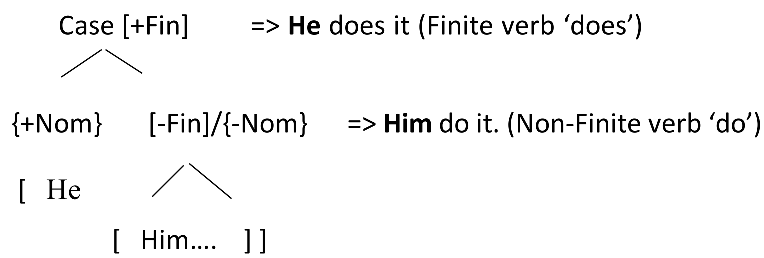
VP is merge-based/recurrent: combine *[drive, car*]

*(Again, notice how ITEMs equate to Base-lexical words, and CATegories equate to move-based Edge-operations. Also, note how the final /s/ as pronounced in 'fix' /fIks/ is an ABABABA recurrent grammar, part and parcel of the lexical (item), while the final /s/ in 'speaks' /spiks/ is recursive (category): e.g., [fIks] vs [[spik]s]. See Galasso, **Note 4**).

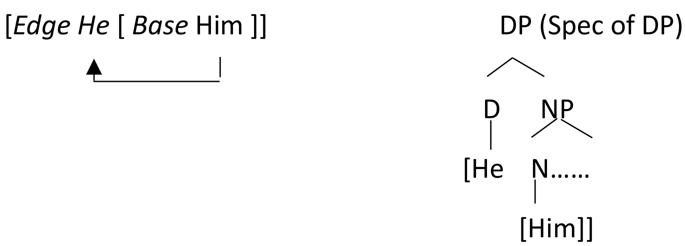
[4] **Case**

Case, being an instance of INFLectional morphology likewise shows recursion. Consider below the developmental progression from (i) **default** Accusative [-NOMinative] Case to (ii) (recursive) Structural [+NOM] Case in child syntax:

(i) Him do it => He does it. [{+Nom} He [{-Nom} [Him]]. For the full form 'He does it', NOM Case 'He' is a reflex of a Finite-Effect relation to the Finite verb 'does'.



Note the movement from ITEM-based 'Him' to CAT-based 'He' found at the Edge:



(* Default Case: Who wants ice-cream? Me! (not I!) Note how the default Case in English gets spelled-out in the Accusative [-NOM] form (unlike in Spanish where [+NOM] is the default (Yo!). You did it? Him! (not He!), etc. Spec of DP is a recursive functional-Phrase position in assigning Case—noting movement from N to D.

[5] **Possessives**

Too, an instance of INFLectional morphology, consider early child POSSessives below (noting that Possessives are a property of Case):

Recurrent Stage-1: Him car. Boy bike. Mommy sock. Me toy.

Recursive Stage-2: His car. Boy's bike. Mommy's sock. My toy.

(i) Him do it => He does it. [{+Nom} He [{-Nom} [Him]].

DP Case [+POSS] => **His** car (Possessive): {+Nominative} inflects POSSessive {his}

{+Nom} N+N {-Poss} => **Him** car (Non-Possessive) (considered {-Nom/-Poss}

[His

[Him car]]

(ii) Mommy sock => Mommy's sock

DP Case [+POSS] => **Mommy's** sock (Possessive) {+Nom} inflects {'s}

{'s} N+N {-Poss} => **Mommy** sock (Child N+N equivalent of Possessive)

[Mommy 's

[~~Mommy~~ sock]]

<u>Note the dual-stage processing between stage-1 recurrent [..] vs stage-2 recursive [..[..]..].</u>

Stage-2: [Mommy ['s ~~[mommy~~ sock]]] (noting move-displacement of Noun 'mommy'.

Stage-1: => [Mommy sock]

Stage-2: [My [~~Me~~ toy]] (showing move-displacement of 'Me' up to DP for Case)

Stage-1: => [Me toy]

Stage-2: [He does [~~him~~ do it]]

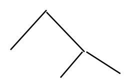
Stage-1: => [Him do it]

[6] **Word Order**

The most intriguing aspect of variable word order found in early syntax has to do with the reading, interpretation of otherwise ambiguous compounds. For instance, in experiment working with very young children, (between @2-4 years of age), studies confirm that when presented with picture tasks (such as a 'boat-house', vs 'house-boat', 'tree-house' vs 'house-tree', etc.) young children have difficulty, and mainly guess, at what the labelling Head of the compound is: (Heads of Compounds show-up rightward, such that a 'house-boat' is a *kind of boat* (and not **a kind of house*). So, for instance, when shown a picture of a 'boat-house', they are unsure if the correct reading is 'a kind of house', or a 'kind of boat'. In order to derive such word order via hierarchy, some amount of move-based recursion is required, and it appears that very young children lack movement. Consider below where a fixed word order is derived via movement: {α, {α, b}}:

(i) [house [~~house~~ boat]] => a kind of boat.

(ii) [boat [house ~~boat~~]] => a kind of house.



[house [~~house~~ boat]]

(where 'boat' remains Head of Phrase: a kind of 'boat', and movement of 'house' instigates adjectival reading).

In an exclusive Merge-based Recurrent stage-1, we also find early word order problems related to examples *cup-coffee, coffee-cup,* noting that the stage-1 possessive expression for *'cup of coffee'* has yet to emerge, thus pinning both elements in competition with one another {coffee, cup // cup, coffee}. The adult version of 'Coffee-cup' in this treatment arises out of the move-based 'cup of coffee' form: (iii) [cup of [coffee ~~cup~~]]

For word order discussion, see paper: https://www.academia.edu/5421727/Why_Move

Afterword: Notes and Conclusion.

See Sentence 3b below (taken from 'The Four-Sentences', Galasso) as a further example of recursive design, tracing progressively the decades of the Chomskyan Framework. See Note C and Fig. A for Brain-Language mapping along the two axes.

Note A. The 'Four Sentences'

The *Four Sentences*—(used here as a pedagogical device*)* serving as postmarks of theory throughout the decades of the second half of the last century—each illustrates, decade by decade beginning with sentence#1 (Chomsky 1956), through to sentence #4 (1980's-style analyses of movement and empty categories): note how the progression of recursive-grammar-[[ ]] came to be understood as undergirding syntactic structure.

Sentence-1: 'Can eagles that fly swim?

v1 v2

i. [Can eagles that fly swim?]

So, if we are simply scanning strings via a process which only adheres to the 'adjacency-factors' of the string, then we should interpret that we are asking 'can eagles fly?' But let's consider now what sentence1 looks like under a recursive structure:

i. $[_x$ Can eagles $[_y$ that fly $_y]$ swim $_x]$

Now, if we consider the nature of recursive structures (as found with embedded strings), then we can see that indeed the closest verb to the subject [Eagles $_x$] (found within the x constituency, or unit of structure), is in fact [swim $_x$] and not [fly $_y$]. As Chomsky puts it, it rather seems that it is due to some unique design of our human brain (a brain which gives rise to language) that allows us to instantiate immediately upon recognition (an innate recognition) the underlying recursive structure of [[ ]] over a flat structure [ ].This recognition is knowledge not learned in school, nor is it taught to us by our parents at an early age, but rather, comes for 'free' out of the *human design* of language.

Sentence-2: 'Him falled me down' (1960s child language studies)

In considering sentence-2, the item we are interested in here is the **over-regularization** of the verb 'fall' => 'falled' (fell). If we were, again, to take the *naive* flat assumption that all words are memorized, stored and retrieved as holistic chunks, in other words as [falled], then the immediate problem surfaces as to where and how the child ever came across such the word, being that it is not supported by the input. This very question goes to the heart of what Chomsky referred to as the **creativity of language.** Berko's work on child language quickly saw that such errors in fact proved that the child was working under a rule-based design of language, and that at roughly the point where over-regularizations take place in the child stages of acquisition, we find that the over-regularizations align with the acquisition of the rule—viz., [[N] + s], for plural, and [[V] + ed] for past), noting that such 'errors based on rules' supports recursive structure. Hence, what we have here with such errors is a decomposed item of [[stem]+affix] e.g., [[fall]ed] whereby the two parts of the words must be stored in distinct units or constituencies as found in the morphology (stem, inflectional morphology).

Sentence-3a: 'The horse raced past the barn fell'.

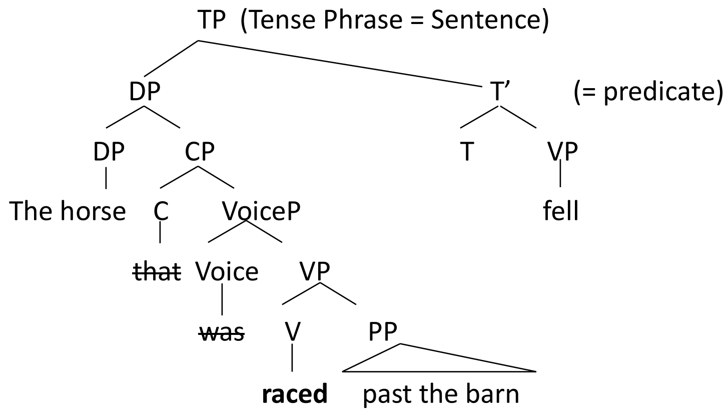
Sentence-3a is also known as a '*garden-path'* sentence. (The classic sentence and its first use is attributed to Tom Bever. In such designed constructs, readers are often lured into parsing (processing) the structure of a given sentence in a certain way, and by doing so is actually led down a wrong syntactic reading of the sentence (i.e., down a 'garden path')—viz., in believing that a grammatical element should follow based on what came prior. In other words, the erroneous assumption is tied to a processing which reads the first verb parsed '*raced*' as a past tense main verb of the matrix subject '*horse*', rather than how it should alternatively be processed, as an embedded passive past-participle of an covert embedded clause (The horse—that was **raced** past the barn—fell). This nice parsing trick shows how the brain seeks to parse and process pieces of syntactic phrase structure in systematic ways, in ways which speak **to phrase-structure rules,** (and in more current theory) **X-bar syntax.**

When the reader first hears and confronts the designed parsing of an initial DP, say 'the horse' (in the above garden path sentence), the DP immediately gets assigned as subject—this is done in concord and under syntactic X-bar theory, assuming that the syntax of the given language is SVO (subject-Verb-Object). Fine, but what this also means is that the following verbal item usually gets assigned as a Tense verb which then, due to phrase-structure rules, determines the Tensed verb to be a matrix predicate of the subject. The phrase structure design would read as follows:

S (sentence) → DP, TP…

But this reading is false. The first Tensed Verb item **raced** does not relate to the predicate of the subject, but rather is part of an embedded structure which should rather be parsed accordingly:

[S [DP The horse] [~~that was~~ raced past the barn] [fell]]



(where Voice P = Voice Phrase for passive voice *was raced*).

Sentences 3b:

(i) [The boy Bill asked to speak to <u>Mary thinks</u> he is smart].

So, Who is doing the 'thinking'?

Assuming a flat ABABABA-grammar […] in (i), then the closest person (argument) who could do the 'thinking' would be Mary (since <u>Mary thinks</u>… is adjacent): <u>Bill</u> might be the second-best choice if you assume the ABABABA-theory that 'closeness counts'). However, this is not how the human recursive-mind processes the structure: the human mind (here, not an ABABABA-grammar), rather processes the sentence recursively, as in (ii) viz., It is 'The boy' who is doing the 'thinking':

(ii) [<u>The boy</u> [bill asked to speak to Mary] <u>thinks</u> he is smart].

[The boy [who Bill asked to speak to Mary] thinks he is smart.]

The 'Who Bill asked' clause is a relative-embedded clause:

[……[Who Bill asked to speak to Mary]…] (begins a different (relative) clause shown with different brackets.

If we assume a flat [ABABABA]-grammar, Mary (or even Bill) would be the 'closest' possible argument (person) doing the 'thinking' (*closeness* in terms of adjacent position, but not in terms of structure). But as it turns out, we utilize without even knowing it, an embedded-recursive grammar <u>where there may be dependencies at a distance.</u>

This is where AI/Deep Learning breaks down, due to such long-distance dependencies (See Appendix D).

Finally, *'wanna' contractions*—perhaps better than any other syntactic phenomenon which illustrates the underlying structure of syntax which incorporates empty categories not heard in the surface-phonology, but which nonetheless impede, psychological, between two syntactic constituencies—demonstrate how adjacency can't be the final verdict in explaining syntax:

For a full discussion on the 'Four Sentences', see:

https://www.academia.edu/43319709/Class_lectures_and_reflections_on_syntax_draft_includes_4_Sentences_

For a popular article in American Scientist, see:

https://www.americanscientist.org/article/the-uniqueness-of-human-recursive-thinking

Wanna Contractions:

(1) Who do you wanna help?

a.	You do want to help who?	(base-structure before movement)
b.	Who do you__ want to help __	(after movement)
i.	~~want to~~	(want-to' is adjacent in underlying structure)
ii.	<u>'wanna'</u>	(OK 'wanna' contraction)

In (1b) above, the 'wanna' contraction is permitted in phonology:

e.g., Who do you *wanna* help?

However, note the syntactic distinction in the underlying structure pertaining to constituents 'want' and 'to' in (2) below (i.e., they are not adjacent):

(2) Who do you want to help you?

a. You do want who to help you? (base-structure before movement)

b. Who do you__ want __ to help you? (after movement)

i. ___want Ø __to help you? ('want' 'to' is NOT adjacent: Ø = empty category) ii. *wanna (* ungrammatical)

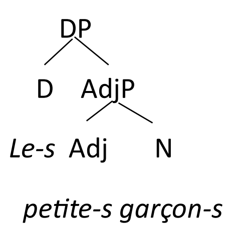
(*Who do you *wanna* help you?

<u>Note B. Comparative Language</u>: A look at grammatical number ('regular' sound-pattern rules which provoke MOVE vs irregular 'broken' patterns which provoke MERGE). Noting how regular sound pattern of N+ {s} instigate Recursive MOVE [[N] s], while irregular broken sound-patterns [ ] which must be memory-based instigate Recurrent MERGE.

French Plurals on Nouns:

In French, the great majority of plural Noun-constructs are based on the regular sound-pattern of add {s}: (e.g., *[garçon] > [[garçon]-s]* (not unlike what we find in Spanish, English, even German). However, there are a minority of nouns which must be memorized via broken-sound patterns which require either add {x}, (when nouns stem ends in -eau, -al, -au, -eu: e.g., cheval > [chevaux] (horse > horses), genou > [genoux] (knee > knees), etc. There are a number of examples which would seem to be included in the above irregular patter (e.g., -{al}) which nonetheless take on the default plural {s}: chacal > [[chacal]s] (jackal > jackals), etc. Please note in French that like many European languages, both Adjectives and Nouns must AGRee in number:

Les petites garçons (the little boys), (noting the {s} spread across the entire phrase:



(DP , AdjP: Determiner Phrase, Adjectival Phrase)

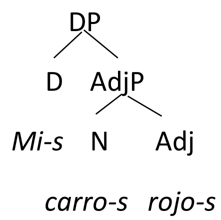
(note word order of Adj + N)

Other examples of the spreading of the plural rule include*:*

Hautes côtes des nuits (= high-s coast-s of

night-s: high coast of nights).

Note similar projections as found in Spanish (noting Spanish word order of N + Adj):



(My-s car-s red-s : my red cars)

Spanish Plurals on Nouns:

The great majority of Spanish nouns require the exclusive addition of {s} for plurals (with the sole exceptions of foreign noun borrowings which must be memorized (e.g., gentleman > gentlemen, *gentlemans).

Spanish Plural: el libro> los libros (the book> the-s book-s), la pluma >la-s pluma-s.

*Note: One interesting note here is the distinction in morphological processing between {Los} (which doesn't have a rule counterpart of {Lo} > {Los} , since [Los] is irregular/memory-based and undecomposed, (not decomposed as *[Lo [s]]), while the regular [La] is a stem with affix {s} addition [La [s]]. Hence, most native Spanish speakers don't even realize that {Los} has its matrix counterpart of La-s (because the singular {el} to plural {Los} has supplanted (highjacked) the regular pattern of {singular stem {La}, to plural affix {La-s}.

German Plurals:

German has five plural suffixes: -(e)n, -{e}, -{er}, -{s}, and zero {Ø}—three of which allow umlauting, an irregular broken phonological sound-pattern (somewhat similar in English to {foot} > {feet}, {goose} > {geese}, etc.):

-{er}: das kind (child) > die kinder (children)

-{(e)n}: die schlange > die schlagen (snakes), die frau > die frauen,

We note that though the German {s} applies to quite a small percentage of nouns, it seems like in English to behave like the default rule: e.g., for names of cars: Golf > Golfs, for names of movies and plays (faust > faustes), as opposed to faust > fäuste (fists) which takes -{e}.

Recall, that in English where there may be umlaut-like irregular plurals, {mouse} > {mice}, English uses the {s} as default: e.g., I need to buy two new 'mouses/*mice' for my computer, I am tired of taking series these administrator 'mickey-mouses/*mice'.

Of course, such defaults surface within the verbal formation of past tense as well, where the English rule is –{ed}: e.g., past of {fly} is {flew} (a broken sound pattern)—but note in baseball lingo, 'Last at bat, he flied out right' where [fly[ed]] takes on the default status –{ed}.

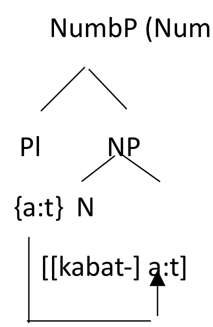
Arabic Plural on Nouns:

In Arabic, there are 31 instances of irregular broken-patterns for plural. Such products of MERGE-based, given their word-based processes are actually produced as itemized lexical items—not unlike how in English the word 'children', despite being produced in phonology as a token 'single word', actually gets processed syntactically as a stem √child + plural rule {ed} in the underlying mental, syntactic procedure. For this reason, children have an easier time which such broken patterns since they only need to abide by the surface-level sound : meaning, and don't need to operate on deeper-level involving a MOVE-based operation employing stem + affix . Of course, this realized distinction between the two modes of operations, surface vs deep, is only tacit knowledge and children as well as adults don't require to make deliberate choices or preferences of the two. For example, in English there are at times two options regarding surface-broken patterns vs regular patterns of plural (e.g., [dreamt] vs [[dream] ed], [dove] vs [[div] ed], [hung] vs [[hang]ed] (though the two words 'hanged' 'hung' have different sematic interpretations), etc., it seems when it comes to mere speed of processing, the irregular broken-pattern (as I**TEM**) wins out. This is due to the underlying processing of decompressing grammar into a lexical item (like what happens with the word 'yesterday' as a grammatical marker for past tense {ed}, or the word ' children' as a plural marker for [child] + s}). However, what the regular pattern provides is productive creativity: i.e., we don't need to have heard the sound:meaning in order to recognize that simply by adding {s} onto the Noun (as **CATEGORY**].

Irregular Broken-pattern:

[Kitaab] (singular) => [kutub] (plural) : (vowel shift)

Regular patters for plural: {u: n} and {a: t}



√[Kabat] (=cupboard) √[[kabat-] a: t] (= cupboards)

√[tufa: ħ] (= apple) √[[tufa: ħ-] a:t] (= apples)

Russian Noun Plural: Add {a}:

город - город-а (eye > eyes)

лес - лес-а (house > houses)

цвет - цвет-а (house > houses)

Irregulars: word change which must be itemized and memorized:

человек - люди (person > people)

ребёнок - дети (child > children). брат - братья (brother > brother)

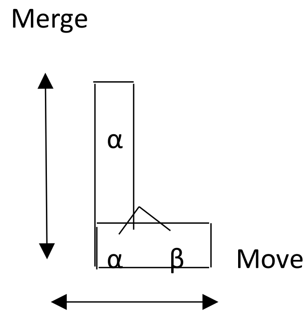
***Other languages*.**

For the most part, East-Asiatic language typologies like Chinese, Japanese, Korean utilize lexical word formations in lieu of inflectional morphology, such as in Japanese ‘Two book’ (where the determiner ‘two’ indicates plural), or in Chinses (where a time-reference word such as ‘yesterday’ would indicate past tense, e.g., ‘Yesterday I visit the museum’). American Sign Language (ASL) might too come down on this side of the spectrum where *word-lexicalization* takes the place of *INFLectional-morphology*. In this sense, the aforementioned language-types fall along the recurrent side of the spectrum, as opposed to our earlier cited Latin languages (Spanish, French, Italian) which come down on the recursive side (at least in the respect of morphology).

*Note that we don’t even address here the many other prolific forms of inflection, such as Subject-Verb Agreement, Verbal Conjugations, Structural Case, Subjunctives, etc. in all of their ‘regular vs irregular’ glory.

<u>Note C: Vertical vs Horizontal Brain Mapping</u>

Note: Recurrent vs Recursive structures project a two-prong axis in parallel to what we find regarding the two stages of child syntax (termed stage-1 MERGE vs stage-2 MOVE):



This dual pathway is expressed as Merge {a, b} which gives you an unordered SET, while Move provides a hierarchical PAIR: < α, <α, b>>.

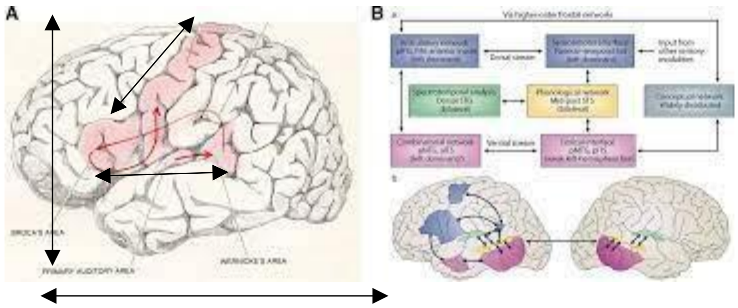
It's quite interesting to view this vertical vs horizontal directionality as a dual pathway to what we see in the brain. It seems rather intriguing to capture this same dual axis when viewing—say, with brain scans, (fMRIs, EEG, and Event related Potentials (ERPs)—that Nouns, Verbs, Adjectives (Lexical Categories) behave in up-and-down mapping directionality, with *vertical pathways* beginning at the Temporal Lobe (TL) and proceeding upward to the motor-strip areas of the brain (the top potions of the mid-brain where cognitive, motor-control, and even aspects of semantics come to bear on thematic/argument structure), as opposed to how horizontal pathways map from Broca's Area (BA) and proceed to the Temporal Lobe and back, forming TL-BA loops.

In fig. A (below), the pink area which slices the brain *vertically* up & down maps lexical semantics with motor-strip, cognitive areas—this is our 'vertical stacking' of items, tethered to more primitive and robust motor-control area (top-mid neocortex) where we find cogno-semantics in accordance with ABABABA-Grammars. This arrow leading upwards takes a lexical item and maps it onto general problem-solving skills associated with motor control etc. (e.g., that a Noun 'book' is something that you can *weight, write, offer* as a gift) but that 'walk' is an all-together different concept with different cognitive-conceptual (problem-solving) requirements. The arrow which moves *laterally* in a *horizontal* way maps a lexical item (TL) to the functional-abstract areas of the brain (BA) which involves syntax.

Also, what we know of Autism spectrum (e.g., Asperger's Syndrome (AS) vs Williams' Syndrome (WS)) seems to project in similar trajectories (with AS falling along the vertical axis, given that AS subjects often prefer lexical retrieval and factual (declarative) information over discourse (procedural) information, and conversely for WS which falls on the horizontal axis. Of course, most famously, the spectrum along these two axes shows us how the two forms of *Aphasia* attack the brain—with *Broca's Aphasia* impacting the horizontal-axis of processing (Rules/Syntax/Recursion) and where *Wernicke's Aphasia* impacts vertical, itemized-lexical retrieval.

Vertical Impact: Lexical retrieval problems, semantic difficulty, Wernicke's Aphasia, Asperger's Syndrome.
Horizonal Impact: Grammar problems, syntactic difficulty, Broca's Aphasia, Williams' Syndrome.

Fig. A



Template scaffolding overlaps onto linguistic theory*

Vertical: Stacking of Items: Lexical Categories (Recurrent): + Frequency Effects

X=X/ MERGE, associative-based, item-dependent/SMM (Learned)

Items [V, N, Adj] / [+ Freq]

[dream] [verb]

{irregulars} = new word /phonological shift of stem

[dreamt]

X + Y = Z /DMM [-Freq]

ed] = past tense [verb]

{regulars} [[dream] ed] = **[[dream] ed]** /same phonology of stem

X + Y = Z / MOVE rule-based, category-dependent /DMM (Acquired)

Horizontal: Rule-based/Functional Categories (Recursive): -Frequency Effects

(DMM refers to the *Dual Mechanism Model* (sometimes referred to as a *Dual Pathway*). It seems +/Frequency sensitivity play a major role is distinguishing the two processes (also see 'Declarative vs Procedural' model, where the Declarative is much more (vertical) +FREQuency sensitive contrary to Procedural which is -Freq and loosely tacit).

*Note whenever there's a sound shift, the lexical item marks as a new word (see *below):

e.g., (N) bath /æ/ > (V) bathe /e/; (N) house /s/ vs (V) house /z/, (irregulars) dream /i/ v dreamt /ɛ/, etc.

Whereas **items** extend **vertically** [x = x], **rules** spread **horizontally** [x + y =z], the former is **recurrent [ ]**, the latter **recursive [[]].** (As discussed in the Preface, this dual distinction makes-up my personal metaphor of **Items** [x-tables, y-chairs, z-nightstands] vs. **category** [α-furniture [x, y, z]].

*(Consider such words which share semantically close stems but where the stems shift phonologically: e.g., [N glass]-[V glaze], [N grass]-[V graze] /s/>/z/, [N bath]-[V bathe] /θ/>/δ/, plus vowel shift of /ǽ/>/é/. Also note how irregulars such as *dream-dreamt, keep-kept, knell-knelt, dive-dove* must contain a similar phonological **sound shift** in order for the lexicon to identify the item as a new word (X=X). (Sound-shifts facilitate memorization of a new item—there is a difference between *grass* and *graze*, one is a noun-item, the other a verb-item). Also note how only a DMM could handle a certain class of words which can be both irregular and regular (both versions being accepted) at the same time: √*dive (dove or dived), √knell (knelled or knelt) √dream (dreamt or dreamed)* etc.). ______

In-note: One very interesting note here is the finding that has come out regarding rote-learned vertical vs rule-based horizontal modes of processing and the corollary that certain forms of autism map onto this dual distinction: viz., research has shown that while Asperger's syndrome seems to heavily depend on memory schemes governing vertical-encyclopedic-based knowledge and processing, this mode of processing is seen in direct opposition to Williams' syndrome which shows an over-emphasis on horizontal, rule-based processing.

______

So, to recap, what our theory above shows (implicating a DMM as compared to an SMM) is that with such high frequency [+Freq] learning, (as with any skill which relies on brute-force memorization), what we get statistically is the bell-shape curve (below). On the other hand, when the competency level seems to reach a mastery competency across 100% of its demography, what we suggest is that such a **right-wall** is consistent with what we find of **biology**. It has long been recognized that **first language** (L1) **acquisition**, as compared to (post-critical period) **second**-**language** (L2) **learning** follows this same trajectory—with L1 biology pegged to right-wall distributions, and L2 learned skills pegged to bell-shape curves.

*A final note on brain-mapping has to do with lexical vs functional grammars. Granted that lexical categories (Nouns, Verbs, Adj, Adv, Prep} are located in Wernicke's area of the brain, we would suppose that functional material, say INFLectional morphology, would never enter into a lexical slot, since both lexical v functional are distinct reflections of unique brain processing: Lexical-to-Wernicke's/Temporal Lobe region vs. Functional-to-Broca's/Frontal Lobe region. The grammatical slip below nicely shows how even errors are systematic and flow from this theoretical processing distinction:

Error: What about taco tonight-s?

Target: What about taco-s tonight?

A lexical functional distinction would slot the words accordingly:

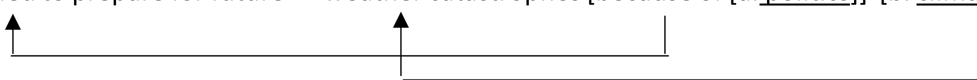
[What] [about] [[taco]s] [tonight]?

Note how the word [tonight] would have a slot free for a functional inflection [[tonight]], such as possessive {'s} [[tonight]'s], or verbal {s} [[tonight]s]: e.g., tonight's party, tonight's going to be fun. Hence, the functional inflectional {s} was misdirected into a functional slot at the end of the sentence, since the slot [[ ]INFL] is an appropriate syntactic-processing slot. However, also note how surface-adjacent {s} and {t} (tacos tonight) could never entertain such errors (unattested @what about tacot sonight?), or likewise, how the *lexical* {t} from 'TACO' could never misdirect as an error and insert at the *functional* end-slot of 'TONIGHT' (unattested @ _ACOS TONIGHT-T).

Note D: Artificial Intelligence (cited from Gary Marcus).

AI-Deep Learning has catastrophic difficulty with distant dependency:

E.g., We failed to prepare for future weather catastrophes [because of [a. politics]] [b. climate change]].



Once the phrase 'climate change' is uttered, its logical 'cause-effect' reference must refer back to 'weather catastrophes' (i.e., 'due to climate change'). The phrase, 'politics' refers back to 'fail' (i.e., 'reasons for failure to prepare'). Natural readings assign long-distant references very easily in human nature language (as part of what we know about the word, termed background/common-sense readings). However, such long-distant dependencies breakdown very rapidly under deep-learning/AI algorithms.

Perhaps the best example I have come across of AI/Deep-Learning catastrophic failure of distant dependency is the following French examples taken via Google translate: (cited in Marcus et al. p. 87):

When Google translates from English to French of the sentence…

English: The electrician whom we called to fix the telephone works on Sundays.

Google improperly translates to…

*French: L'electricien que nous avons appelé pour réparer le telephone (i) *fonctionne le Dimanche.*

Should be (i) travaille *: 'L'electricien travaille le Dimanche'*

The French notion of the English verb 'work' has two aspects: (i) 'fonctionne' which means 'works'/functions properly, and (ii) 'travaille' which means 'labors'. Clearly, the above French sentence means to say 'travaille' *(i.e., The electrician labors/works on Sundays, and not 'properly works/functions on Sunday').* Note the problem: AI/deep-Learning relies on **frequency and adjacency** of the usage [telephone + Verb], and that the (adjacent) noun ‘Telephone’ is most frequently used in **big data** with ‘functions properly’ (or ‘fonctionne’):

> *a. A telephones functions (works) : (Fr. Fonctionne).*
> *b. An electrician labors (works) : (Fr. Travaille).*

The electrician whom we called to fix the [telephone works] on Sundays. (AI Adjacent bias: Recurrent)

[The electrician] whom we called to fix the telephone [works] on Sundays. (Syntax: Recursive long-distance)

Consider one last example, just exercised by me this week using both CHAT-GPT and Perplexity AI:

Prompt your favorite AI platform with the following question:

In the question: ‘The big muscle man who knows the little girl plays with dolls’ : Who is playing with dolls?

So, ‘Who is playing with dolls’? The answers at times are unstable, but the leading AI responses seem to favor ‘The big muscle man’. Such catastrophic errors can be the result of a convergence of both adjacency (since ‘little girls’ and ‘play’ are adjacent strings, and/or pure SEMantics since ‘muscle men’ don’t play with ‘dolls’ while ‘little girls’ do.)

Natural language response: The Recursive Linguistic Mind):

> [The big muscle man [who knows the little girl] plays with dolls].
>
> (A^n B^n): [A[A[AB]B]B]

AI response:

> [The big muscle man who knows the little girl plays with dolls].
>
> (AB^n): [ABABAB]

Afterword

Second Language (shallow Processing). My former PhD-dissertation supervisor Harald Clahsen (University of Essex) (among others) have shown that linear-derived (linear-capture) grammars manifest in L2, a result of a recurrent (non-recursive) algorithm similar to what we have seen with logical-and' procedures. Note that oft-error made by L2 speakers (as a result of ABABABA-style shallow processing):

Did he visited New York? [Did] + [He visited New York]?

This algorithm, a result of linear-derived grammar, has the strategy of combing the well-formed sentence [He visited New York] with the directly inserted (pulled from the lexicon) Yes-No Question operator [Did], thus resulting in the flat (beads-on-a-string' grammar [ ] + [ ]….

> Such errors of multiple finite-verb structures are the result of the lack of movement from the base-generated structure 'He (did) visit New York' => '[Did [he ~~did~~ visit New York]]?' There are other such error types as found in (adult) L2 (post-critical-period) that seem to rely on this erroneous linear-grammar of 'logical-and'. (Cf. Wh-questions: 'Who did he *saw/see yesterday?, etc.

Child Language. One way to capture what we know about early child syntax is the provoke a discussion surrounding the maturation and onset of **Movement-analogies** (and the lack thereof). Given that INFLectional morphology is a hallmark of movement[3], one way to define a stage-1 recurrent grammar is to suggest that stage-1 lacks movement. Hence, out of such a prosaic processing arise the lack of Tense, Case, [+Finiteness], Subject-Verb Agreement, and fixed Word Order. Regarding general maturational factors in the brain, as has been studied (Aphasia, Language Impairment, Child Syntax), a clear picture is emerging that recursiveness is pegged to a healthy and full-fledge onset of Broca's area, (particularly area BA 44).

Finally, the 1990's proved to be an influential period for movement-based analogies of child syntax with the seminal works of (among many others) Ken Wexler (MIT) who claimed for an *Optional Infinitive* stage1, *pace* Radford, Radford & Galasso (1998) who rather argued for an early stage-1 exclusively void of movement. The first decade of this new millennium would see a more subtle movement-based analogy to Chomsky's *Minimalist Program* (1995) by specifying an (External)-Merge vs (Internal)-Merge (= Move) dual processing mode, the former being rather *recurrent* in fundamental ways, the latter quintessential recursive in nature.

---

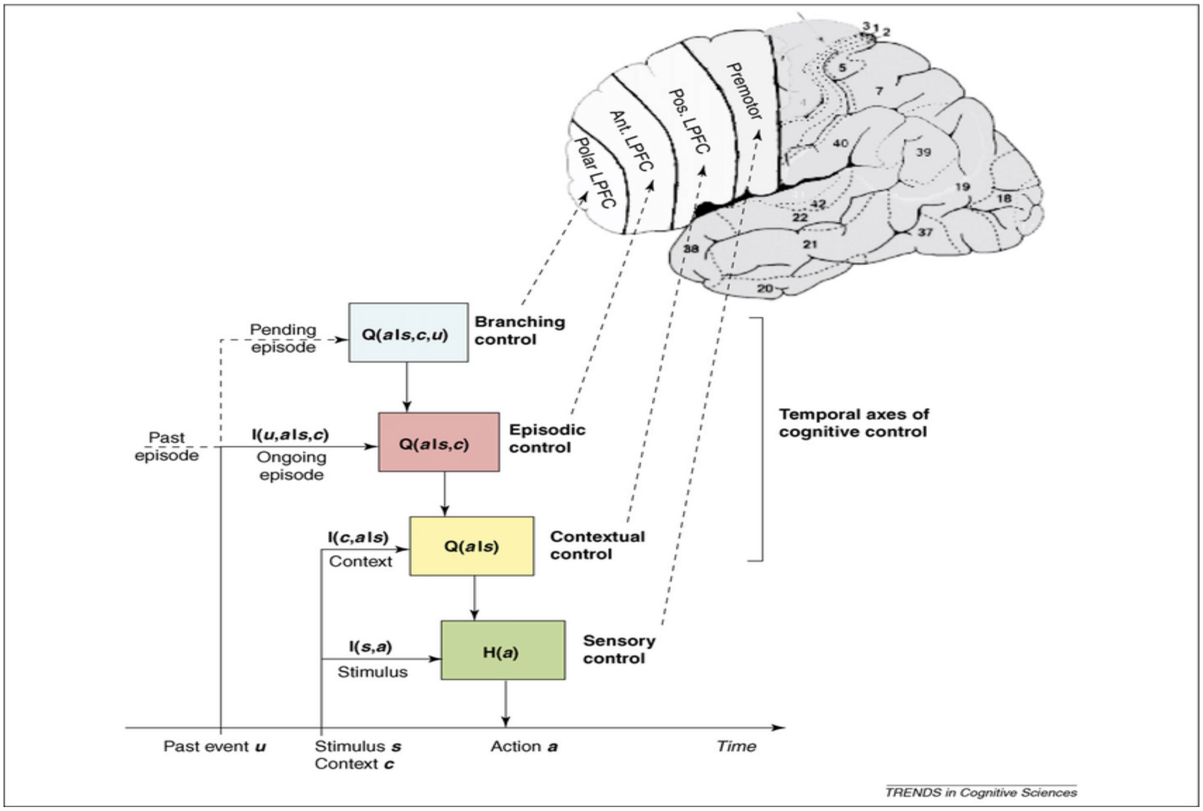
[3] See 'Fascinating vs Celebrating' typologies for full discussion: https://www.academia.edu/75445941/3_Lang_Acq_Movement_Distinctions

Aphasia. I have had one experience working with a global aphasia adult subject (the victim of a stoke) whose only words he could utter were curse words (e.g., *fuck*). It seems this one word, more than any other, was preserved after the stroke. But why? One way to account for this fowl-mouth preservation is to understand how such a word as 'fuck' ticks all the boxes, 'up-and-down' the pathway of vertical mapping. For instance, the word comes replete *with emotions, anger, dissatisfaction, sexual innuendo*, it is a great *intensifier*, it's used for *emphatic* purposes, and it seems to be so pervasive in its lexical mapping that it can be used as a Verb, Noun, Adjective…The strengthening of such a word due to its multi-map overlapping virtues seems to have saved this word from oblivion. So, let's hear it for the 'F' word…the most ubiquitous word in the English Language!

For a fine paper on Cognitive $(AB)^n$ **Merge-base** (Recurrent) (ABABABA-Grammar) computations related to S&R stimuli vs. more abstract-level $A)^n$ $(B)^n$ **Move-based (Recursive)** computations, showing related cortical regions, see Koechlin's 2007 paper:
https://www.researchgate.net/publication/6353749_An_Information_Theoretical_Approach_to_Prefrontal_Executive_Function
DOI:10.1016/j.tics.2007.04.005

The paper nicely shows the cascading cline from (i) *Sensory control* (S&R) (motor-cortex part of the brain), to (ii) *Contextual control* (which overrides Sensory), to (iii) *Episodic*, and finally to (iv) *Branching* (the term ('Branching' itself denotes multiple movement (tree-diagram) analogies). Whereas the first sensory control triggers ABABABA-recurrent neural networks and computations, the latter more abstract three controls trigger recursive movement operations.

> Abstract: The prefrontal cortex subserves executive control--that is, the ability to select actions or thoughts in relation to internal goals. Here, we propose a theory that draws upon concepts from information theory to describe the architecture of executive control in the lateral prefrontal cortex. Supported by evidence from brain imaging in human subjects, the model proposes that action selection is guided by hierarchically ordered control signals, processed in a network of brain regions organized along the anterior-posterior axis of the lateral prefrontal cortex. The theory clarifies how executive control can operate as a unitary function, despite the requirement that information be integrated across multiple distinct, functionally specialized prefrontal regions. (Image: Koechlin 2007)



For a good discussion of the brain-language mapping as presented here, see 'The Neuroscience of Language' (F. Pulvermũller, 2002, Cambridge University Press). Steven Pinker's classic 1999 book 'Words & Rules' (Basic Books) first presents a formal treatment of this *Dual Mechanism Model*. Gary Marcus in his 2001 book 'The Algebraic Mind' (MIT Press) shows us how non-recursive ABABABA-grammars ultimately disappoint our over-hyped promises behind Artificial Intelligence (AI)—and how AI is simply reduced to being a turbo-charged probabilistic calculator of sorts. Most recently, Gary Marcus, equally antagonistic, has come out against all the hype behind GPT-3 super computers designed for language translation. See link below:

https://www.theguardian.com/technology/2022/aug/07/siri-or-skynet-how-to-separate-artificial-intelligence-factfrom-fiction

For review as German {s} as default, see Marcus et al.

https://www.researchgate.net/publication/222460691_German_Inflection_The_Exception_That_Proves_the_Rule

For a full discussion of how recurrent equates to ITEMS vs Recursive to CATEGORIES, see Note-4 below):

https://www.academia.edu/71813057/Note_on_Movement_Distinctions_based_on_Inflectional_vs_Deri

vational_Morphology

For a review of AI, see **Galasso Note 4**:

https://www.academia.edu/39578937/Note_4_A_Note_on_Artificial_Intelligence_and_the_critical_recursive_implementation_The_lagging_problem_of_background_knowledge_1

On AI, see book ‘Rebooting AI’ by Gary Marcus and Ernest Davis (Pantheon 2019).

See ‘Speaking Brains’ on Recursive neuro-networks and Basal ganglia Grammar: https://www.csun.edu/~galasso/speaking%20brains.pdf

For review on MERGE over MOVE, see:
https://www.academia.edu/5761528/Lecture_1

https://www.academia.edu/34403440/Working_Papers_5_Minimalist_Perspectives_on_Child_Syntax_Merge_Over_Move_Movement_Applica tion_in_Child_Syntax

For recursion in language, see Hauser, Chomsky, Fitch (2002):
https://www.science.org/doi/10.1126/science.298.5598.1569

____

Endnote. Marr’s Recursive Neurons.

The late (and sorely missed) neuroscientist David Marr, I Believe, was the first to ever proposal that neuronal connections may take-on recursive-loop functions. The closest analog to recursiveness, at that time, happening at such micro-levels, probably related to what we thought was happening with mirror neurons, whereby interlocking bidirectional neuron firings might mirror each other in forming feedback loops. The notion that the transition loops of GFPs (gestalt frame potential) from (i) short-term to (ii) long-term, back to (iii) short-term memory retrievals, which indeed adds noise to the neuron, nonetheless may not necessarily mean that the neuron is completely eliminated, as was at that time sometimes suggested. Rather, it is currently thought that given a recursive neuron connection, amplifications may ripple through the neuron-bundles via dendrite-axion pathways in quite dynamic ways, allowing memory-traces (pentimento) of the original stimuli (first time-step GFP) to remain, while also incorporated additional overlapping information, (sometimes referred to as ‘noise’), rendering embedded feedback loops. Marr went on further in suggesting that neural networks can exhibit pure recursive functions having to do with reverberating circuits. Marr suggested that while synapses become excited, information packages will pass along the chain of neurons, with the last neuron in the chain be amplified or attenuated. Marr stated: ‘However, the output neuron also feeds back the same output message back to another neuron, which then loops the information back to the penultimate neuron in the chain’. In Reverberating or oscillating circuits, the incoming signal travels through a chain of neurons, each of which makes collateral synapses with neurons in a previous part of the pathway. It is this ‘collateral’ aspect which suggests the potential for recursive reverberation. Notice how input impulses run parallel, reaching the output cell at different GFP time-steps. In parallel processing, the input travels along several different pathways to be integrated in different GFPs timesteps and regions. It is believed that it is this parallel processing which allows human neurons to take on recursive functions. Marr had always speculated that it was the *pyrimidine neuron* which held the key to such recursive looping. Coupling this with what we now know today of astrocyte-basal ganglia-thalamus function, perhaps there is a path forward in bringing AI/Transformer systems even closer to natural language. While current state-of-the-art AI/Transformers may not be there yet—and there are still plenty of cognitive scientists and linguists willing to bet the house that we will ‘never get there from here’ (and I count myself amongst them)—nonetheless, we seem to be approaching a *pivotal point* in realizing a possible convergence, bringing AI/Transformers on par with our understanding of the neuro-circuity behind a brain-to-language corollary. Below, we see Marr's original scheme for a recursive 'reverberating-circuit' neuron.