

## Five Notes on Recursive Syntax:

### *The Dual Mechanism Model, Problems of projection, Proto-language, Recursive implementation in AI, and the Brain.*

*Joseph Galasso (California State University, Northridge~Linguistics Dept.)*

These *five notes* have very much in common with one another. First of all, the whole notion of 'label of projection' seems to conflate a portmanteau of syntactic *mechanisms* and *features*, one of which is to determine exactly which, out of a host of possible syntactic operations, is singularly required in order to label a phrase.

The notion of *labeling* a phrasal projection (e.g., VP, DP, AdjP) has become a central question with regards to the minimalist program (MP). Secondly, once recognizing which of the mechanisms are defined for labeling, it becomes clear that the notion of syntactic *Movement/Move* (as a recursive property) immediately gets implicated as the essential property of the labeling process (anti-symmetry). As addressed herein these five notes, this unique *recursive* property is found not only to be the engine behind movement and labeling of a phrase as such, but, furthermore, when defined as that quintessential ingredient to human language, *Move* comes to be considered as the one core component which would be crucially required for any approximate attempt at Artificial Intelligence (AI)—that is, if the reconstructing of a *near* 'human-like' mode of processing is what is being sought.

*Move* can also be looked upon in its subsequent absence. Namely, we find evidence that very young children's syntax (Radford & Galasso) parallels what we find of adult *pidgin* language, as well as what we might speculate of a theoretical *Proto-language* (Bickerton). Defining *Move* as the sole and unique property of human language (Fitch, Hauser, Chomsky), we go on to present a theoretical 'brain-to-language' corollary—via a Dual Mechanism Model (DMM)<sup>1</sup> (Pinker,

---

<sup>1</sup> **DMM, a working definition.** The DMM was first proposed to address processing distinctions held in the brain which showed demarcation between regular rule-based inflectional morphology, as contrastive with memory-based irregular and derivational morphology. Our DMM proposal here extends this to a dual processing treatment of the brain between frequency sensitivity of nodes/items/lexemes vs. non-frequency determination of symbolic values—the former being local in nature (equivalent to adjacent neuro-cells firing together), the latter to recursive operations of variables performed over non-adjacent distance. The DMM upholds a dual processing: (i) a *memory* component of the brain (traditionally assigned to the temporal lobe region of the brain (Wernicke's area) which is reinforced by classic behaviorist means, while (ii) also strapping-on an added stipulation that human-brain processing is rule-based in nature (classically attributed to Broca's area). Recursive syntax and Movement is hypothesized to be exclusively a Broca-region processing, as shown in

Ullman) that allows us to bundle together this simple fact about Move and target what we believe constitutes a well-defined working model of human language. Such an articulated model would not only have to address aspects of AI (and problems thereof found within that enterprise), but would also have to jibe with evidence gathered in the **Broca Aphasia** literature (Grodzinsky) which, among other things, demonstrates what happens when Move is altogether absent in the underlying syntactic processing of adult speakers.

In other words, we call upon a well-articulated DMM to do more than what its initial authors set out to do in the 1990s—which was to mainly show how regular inflectional morphology (*walk>walked*) is processed differently from irregular morphology (*go>went*), etc—the former being *recursive/rule-based* and NOT sensitive to frequency of the data, while the latter is inherently *associative-based* and indeed very [+frequency] sensitive. Our current treatment of the DMM is much more ambitious as it attempts to pull together the data found in these five notes in such a way as to speak to: (1) an emergent DMM (maturational child syntax), (2) as situated in the brain (lateralization), (3) replete with its proper and distinct modes of processing (rule-based), and (4) whereby not only inflectional morphology gets called into question, but so too does the very nature of recursion itself.

This dual aspect of the DMM has led cognitive scientists along with developmental linguists to theorize on how the DMM might be called upon, today, to deliver a brain-to-language mapping that is consistent with what we have learned of recursive syntax over the span of a half century of discover, a vastly rich span which has made up the Generative Grammar Enterprise (GE).

<>

What we attempt here in these *five notes* is to weave a unifying treatment of recursive movement in such a way as to capture some of the critical aspects of each area of concern. For example, with regards to *labeling* (problems of projection), a DMM theory stipulates that in order to break with a sisterhood relation (i.e., two or more flat items held in adjacency, with no order), displacement/movement must ensue in order to break the non-linear directionality, so as to create a hierarchical, (c-command) projection. It is owing to this recursive displacement that a Head of a phrase is defined along with its Complement. Hence, only via Movement do we find syntax gaining ground on two critical fronts.

---

studies of Broca's aphasics who lack movement operation as found in syntax (e.g., Grodzinsky). (See Note just above, § 9 [59-61]).

*Movement* provides the two-fold ability for:

- (i) Specified word order within a phrase, as well as,
- (ii) The ability to label what triggers as the head of a phrase.

The *DMM*, (most famously promoted by Pinker 1999, and Ullman 2012, among others) demonstrates how recursive movement can be a pivotal, defining benchmark not only for the defining of a dual-stage of child syntactic development (mapping onto the classic lexical vs. functional split), but also for devising theoretical accounts for observed pidgin-language phenomena, (with a boost that it might also go far in speculating what a putative Proto-language, as a way-station to a fully emergent language, might have looked like for our very early ancestors).

A well-articulated DMM also serves as a way to promote a dual processing (maturational) model of the brain itself—of course, with the later-onset of Broca’s area being eventually assigned the lion-share of movement applicability. So, if the DMM holds as a workable theory of syntactic development, it takes only a natural extension to implicate the DMM with what we have long realized concerning Proto-Language—viz., that a proto-language most probably lacked recursion.

By bundling some of the finer aspects of these five notes, and by seeking to justify just how a human-like processing system works, we conclude that there must be at least **three requirements** for human syntax:

### **Three requirements for human language:<sup>2</sup>**

Any recursive algorithm must have:

- (i) A set of primitives,
- (ii) A way of combining these primitives to form new, complex structures,
- (iii) A way of determining that the arrangement of these primitives matters.

---

<sup>2</sup> See Marcus 2001, chapter 7 for a summary.

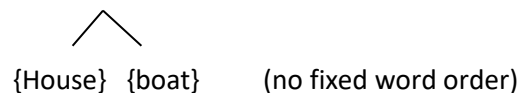
Let's take each in turn:

1. A **pair** of primitives might be for human language simple *words*, i.e., the lexical item. For AI modeling, **primitives** would define the local nodes of a connectionist network. For a model of the brain, primitives would define the *cell* (cell-transport) level. We come to consider these primitives as 'low-level', which are confined to local/adjacent configurations.

In the 'house-boat' example (cf. §Note 4, [2]), the primitives would be the (unordered) *pair* of words: {house, boat}. (Once order is determined, we could speak of the primitives as a **set** [house, boat]: {}-brackets label unordered pairs, while []-brackets label a set). In terms of language, such primitives would be selected from an **array** of lexical items, as found in the lexicon. And that is the extent of such primitives: *nodes, items, cells*, etc.

2. 'A way of combining' these primitives in a meaningful way would mean a particular selection process from the above step-1, and nothing more. In terms of **syntax**, we could claim that before syntax can emerge, there must be the selection out of the *lexical array* of at least two or more primitives (words).

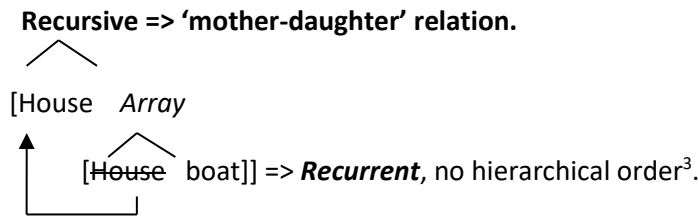
**Array => recurrent/sister-relation**



(Note, within a mere array, there is no way to break the symmetrical ambiguity: for instance, is the above array composed of the pair a 'house-boat' (= a kind of boat) or is it a 'boat-house' (= a kind of house)? In order to derive meaning, e.g., a 'house-boat' is 'a kind of boat', we must look to the recursive structure of syntax found in 3 below).

3. 'A way of determining' that the (asymmetrical) arrangement of these primitives matters. This speaks to the crucial requirement of hierarchy. Hence, our above example (in 2) of the array 'house-boat' (now defining 'boat' as the Head of its phrase/compound), takes on recursive/hierarchical properties and brings about true human syntax. We now know that it is a 'kind of boat'. (See recursive structure below).

Consider the recursive tree below:



We find displacement/recursion here of the [a [a, b]] type: [house [house, boat]] (showing movement of [a] out of [a [a, b]], where 'house' now positions higher up in the tree as the **Mother** of the **Daughter**-relation 'boat', as understood in hierarchical **C-command** relationships. In other words, now, in this newly created C-command relation, we find that the item 'house' has raised into a higher position where it can c-command 'boat'—viz., 'what was once a symmetrical/bi-directional *sister-sister relation* has become an asymmetrical/unidirectional *mother-daughter relation*.

(For C-command, see §4, [12]). Hence, a 'house-boat' is 'a kind of boat'.

Finally, the notion of Artificial Intelligence (AI) has caught the imagination of many theoretical linguists who are more than adamant that in order to fully capture how the human brain-mind thinks and processes information, it is crucial that a recursive operating system be employed.

*Regarding AI, what we want out of a well-functioning operating system (that yields human-like syntax) is the ability to represent relations between local nodes and distant variables.*

One is reminded of the single mechanism model (SMM) days which sought to represent both local and distant aspects of processing in one fell swoop. These were behavioral models (B.F. Skinner) through and through. Such modeling is what remains of current attempts to construct SMM-connectionism.

While some important results have come out of these attempts to model AI (without the three aforementioned requirements), the AI/human-like processing ceiling has never been broken. It has not even been closely approached. One very interesting property which is unique to the DMM brain-language corollary is that it allows for growth (maturation) to take place, matching a **biological basis** for language. There is almost unanimous agreement that Broca's area doesn't come on line until about age 2.5 years of age (hence, a biologically determined,

---

<sup>3</sup> See Note 4 for distinction between recurrent {a, b, c} vs. recursive structure {a {b c}}.

maturational hypothesis). Any language processing which would predate Broca's appearance would have to be exclusively based on a more prosaic SMM, where frequency reins king. This is exactly what we find of e.g., the early lexical-spurt stage, referred to as the **lexical stage** (around 18 months), where frequency of expressions may rematerialize as idiomatic chunks. Early utterances of the latter stage-two **functional stage** (a DMM-stage) are characterized by over-regularizations of the 'drawed', 'goed' sort (as classically observed by Berko).

The critical problem with an SMM is that all aspects of processing are essentially the same (and so any differentiated aspect of processing could not be prone to growth): nodes either trigger based on frequency of stimulation of the environment or they don't. Also, inherent to an SMM is that no *ad hoc* requirement for an innate architecture is needed. Those espousing for a SMM once suggested this to be closer-in-modeling to what we actually find of the brain—when they say 'it is only cells firing' and that 'there can be nothing else' (nothing else innate or otherwise). But it is here that the 'poverty of stimulus' argument is called upon in our theory, and where some innate architecture becomes required ('innateness' reasserts as the biological *null hypothesis*).

<>

Note-1 discusses the theory behind the so-called *Dual Mechanism Model (DMM)*, Note-2 *Problems of projection*, Note-3 *Proto-language*, Notes 4-5 *Recursive implementation in AI and the Brain*.

## Note 1

### **A Note on the Dual Mechanism Model: Language acquisition vs. learning and the Bell-shape curve.**

In this first brief note (one of five), I'd like to reflect on how the *Dual Mechanism Model* (DMM), as compared to a *Single Mechanism Model* (SMM), might inform our more narrow discussion of *Artificial Intelligence* (AI) (discussed in Note 4), as well as inform our larger-scope discussions surrounding the 'nature of language & design' more generally. The description of our methods here will be based on the following dichotomies:

#### **[1] DMM vs. SMM**

(i) Whereas an SMM is solely reliant on brute-force associations which are inherently tethered to overt **Learning**—a frequency endeavor [**+Freq**], where frequency of item-based learning belongs on the vertical mode of processing (to be presented and discussed below). Such item-based learning could be thought of as 'structure-independent' since its focus is solely on the isolated item in question and not on the context of overall structure surrounding the item.

(ii) Whereas a DMM is abstract and rule-based which is inherently tethered to tacit, covert **Acquisition**—a [**-Freq**] endeavor which doesn't rely on a one-one association of item, but rather can be both (i) item-based and (ii) **categorical** in nature, where **structure-dependency** is observant of category over item. Hence a DMM mode—a mode which is both 'item-based' when called upon (e.g. such as lexical learning, irregular formation over rule-based regulars, etc.) and 'categorical-based' when called upon to engage in the manipulation of symbols—is in a unique position to deliver the kind of 'learning curve' which is consistent with what we find of native language acquisition (to be presented and discussed below).

#### **(T)heory.**

(T). Perhaps the sole property of what makes us uniquely human (i.e., the ability to use language) amounts to little more than the sensitivity to remove ourselves from the myopic **item**, and to place ourselves at a perspective, just a step away from the item, and to become sensitive to **structure dependency**. In this sense, T (the ultimate theory of what it means to be human) is that 'taking a step away' from the frequency of an item and seeing how the item

sits in an overall structure. (This process of seeing ‘item plus structure’ will be what makes up T of a Dual Mechanism Model (DMM) as advanced herein these five notes, and what was considered as the core property of language discussed in the four-sentences portion of this text).

It goes without saying that items (lexical words) are quintessential *learned entities* (environmental), they are +frequency-sensitive [+Freq] and carry a classic portmanteaux of features which are typically associated with concrete and conceptual meaning (e.g., Nouns, Verbs, Adjectives). But structure is an altogether different entity. Structure is promoted not by frequency of learning since it is upheld by *categorical processes* which may strip an item away from frequency and place it into a variable standing within the structure. To see what I mean by this *stripping* of the item, let’s consider an example that was presented in Sentence #4.

Consider the two bracketed *Items* (I) (say, as ‘phrasal-chunks’) as presented in the *Structure* (S):

(I) (i) [that is] (a two-word item\*)

(S) (a) I wonder what [that is] up there.

(b) I wonder what \*[that’s] up there.

\*(Word-item here as defined by phonological dominant stress—viz., a single word is represented by a single dominant stress pattern— if two dominant stresses, then two words, etc. Note how the word-item ‘spaghetti’ would have the stress pattern of ‘weak-strong-weak’ with the middle stress being dominant. The item [*that is*] holds two dominant stresses, (hence, a two-word item), as we hear when we clap out the two words, while [that’s] holds only one stress, (hence a one-word item).

Let’s restate the analysis we presented earlier in Sentence #4 below:

The base-generated structure first looks something like:

[2] Sentence # 4 (restated)

I wonder [\_\_\_ [that [VP *is what*]]] up there.



In [2], the Wh-object ‘what’ begins as the object/complement of the verb ‘is’ forming a Verb Phrase (VP), and then gets displaced by moving above ‘that’ in the surface phonology (PF) yielding the derived structure. But if we take a closer look, we see that after such movement of ‘what’ out of the [VP ‘is-what’] phrase, the VP survives only as a head [VP is  $\emptyset$ ] (i.e., the Head (H) ‘is’ survives without its complement object ‘what’). Thus, the phrase is said to ‘partially project’. But partial-phrase projections are indeed allowed in natural languages given that the H still remains (in situ) within the constituent phrase. Hence, we get the licit structure in (a) (as compared to the illicit vacuous/empty VP in (b):

a. I wonder [what<sub>j</sub> [that [VP is \_\_\_<sub>j</sub> ]]] up there? (A licit structure/ oK)

b. \*I wonder [what<sub>j</sub> [that’s<sub>k</sub> [VP \_\_\_<sub>k</sub>\_\_\_<sub>j</sub> ]]] up there? (An illicit structure / Not ok)

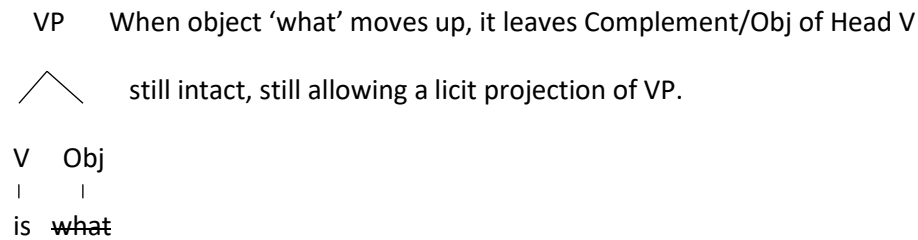
But movement, even partial movement, does have an effect: note how the H ‘is’ must remain phonologically intact as an H of the VP and can’t become a (phonologically attached) **clitic** clinging to the adjacent ‘that’, as in the one-word item [that’s] (whereby there is a reduction now of only one dominant stress). In other words, at least one of the two lexical items within a phrase (P) (in this case, within the VP) must be pronounced (must be phonologically projected). Hence, as we see, when both items [is] as well as [what] move out of the VP –‘What’ moving into a Spec of a higher P along with the item [is] moving out of its head (H) position of the P and forming itself as a clitic piggy-backing onto the item [that] of the higher P—we see the end result that the VP becomes vacuous (completely empty) and so the structure cannot survive (it becomes ungrammatical).

Let’s restate below some points on *move* from just prior discussions.

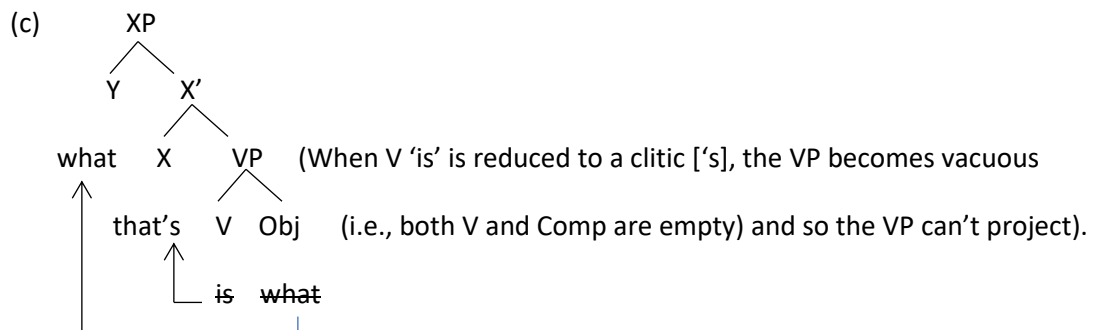
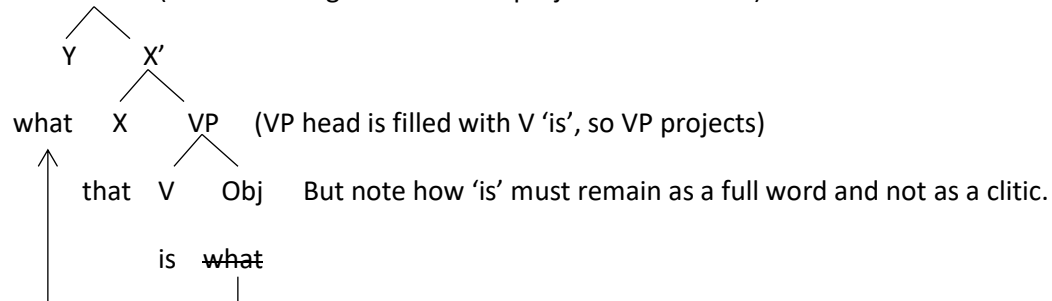
**Moved-based** Hence, \*[[that]’s] is an illicit structure found in (b) (asterisk\* marks ungrammaticality), while **Merge-based** of the two words [that] [is] is the only licit structure. It seems simultaneous movement of both head ‘is’ along with its complement ‘what’ of the [VP is-what] renders the verb phrase vacuous (i.e., phrases can’t be both without a head and without its complement at the same time). In this sense, MOVE-based \*[[that]’s] is barred and only Merge-based (of the two items) [that] [is] is allowed to project—the former (move) being affixal in nature, the latter (merge) lexical). This ‘Merge vs. Move’ treatment is similar to what we find with the distinction between (merge-based) **Derivational** vs. (move-based) **Inflectional** morphology, where the latter is an *affix* process, and where the former is a *word-forming* process. (For a similar treatment of ‘Merge vs. Move’ in child language acquisition, see Galasso 2016).

### [3] Progression of structure

(a) 'is-what' = VP (Verb Phrase)



(b) XP (XP marks a higher functional projection above VP)



But what I want to suggest here for our theory (T) having to do with the following *Five Notes* below—including, and perhaps most importantly, our discussions to come regarding artificial intelligence (AI)—is that this **sensitivity of structure over item** sits as a **core property of language**.

Let's play this out below:

Imagine asking any native speaker of English if the two items below are properly formed (Sentence #4):

- (i) [that is],
- (ii) [that's].

Fine, all native speakers will say both are equally proper in their form. And if there was any preference between the two, the preference would certainly go to the item which is most frequent in the input (i.e., the item most usually heard in the speech environment): that would be item (ii) [that's]. My guess would be that the frequency-count between the two versions could be as high as 'a-hundred-to-one' (if not exceedingly more) when measured in spontaneous speech. That is a perfect example of how the processing of an 'item' is [+Freq]-sensitive: clearly, we hear more numerous examples of [that's] than we do of [that is]. (I am treating the two phrases as *items* here, fragments of constituent structure we hear in the input).

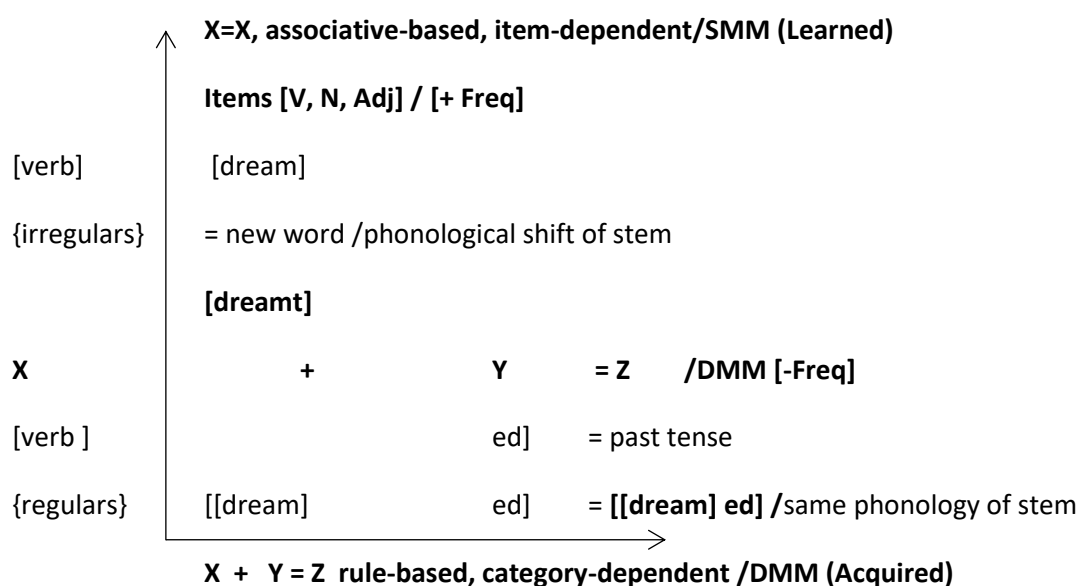
But now reconsider the structure of Sentence #4 (restated in [7] below) and how the isolated item now becomes a rather peripheral feature of the overall structure. Now consider this: when those same people who were earlier asked about the two items *in isolation* are now presented with the same two, but now embedded *within a structure*, all of a sudden the aforementioned preference of [that's] not only becomes the non-preferred item, but, even more egregious, it becomes altogether ungrammatical in its usage (i.e., [that's] it can't be pronounced within Sentence #4).

Here is a perfect example of how +Frequency of **item** is trumped by [-Freq] of **structure**—'Item vs. Structure'. I say [-Freq] of *structure* because structure is not the kind of construct which carries that portmanteaux of features (semantic) which can be readily processed via a brute-force memorization scheme. (Structure is rather category-dependent, syntactic in nature (not semantic), and works on and across variables). In fact, most speakers don't know, nor can they conceptualize, what it is that allows them to tacitly know if one construct is grammatically correct over another (not unless, of course, one is a linguist who works in syntax). Rather, syntactic structure is notorious abstract and hidden, away from the mundane processes of learning a list of items. (Syntax is not simply a list of items gathered which make-up a lexicon).

In linguistic theory, much is made about *linguistic intuition* and to question from where such grammatical intuitions come. One very interesting way to talk about differences between intuition (which seem to arise in a natural way), versus a kind of learned methodology (which relies on declarative understanding of what makes a sentence grammatical) is to overlap a statistical methodology to the range of competency found for such +/-grammatical judgments. The methodology I have in mind here is the classic statistical averaging found across a given

demographic range ( $\gamma$ ), which measures a competency of a given skill  $\langle x \rangle$ . The test is to see whether one finds the classic *bell-shape* curve (a universal staple behind any measurement of a learned skill), or if one finds the so-called *right-wall* (which portrays a biological endeavor of acquisition over learning). This dual outcome of **learned** (bell-shape curve) vs. **acquisition** (right-wall) might suggest how a *template scaffolding* (overlapping linguistic theory) could serve to illustrate our DMM:

#### [4] Template scaffolding overlaps onto linguistic theory\*



Whereas **items** extent **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** [ $\alpha$ -furniture [x, y, z]]).

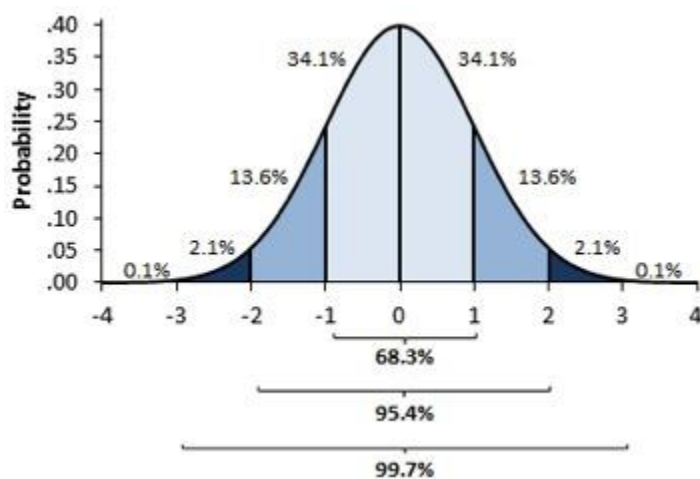
\*(Consider such words which share semantically close stems but where the stems shift phonologically: e.g., [N glass]-[V glaze], [N grass]-[V graze] /s/ > /z/, [N bath]-[V bathe] /θ/ > /ð/, plus vowel shift of /æ/ > /é/. Also note how irregulars such as *dream-dreamt*, *keep-kept*, *knell-knelt*, *dive-dove* 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.).

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.

[5] **Bell-shape curve** (Google© 'free-to-use' image).

#### Competency of a Learned Skill <X> / (L2)

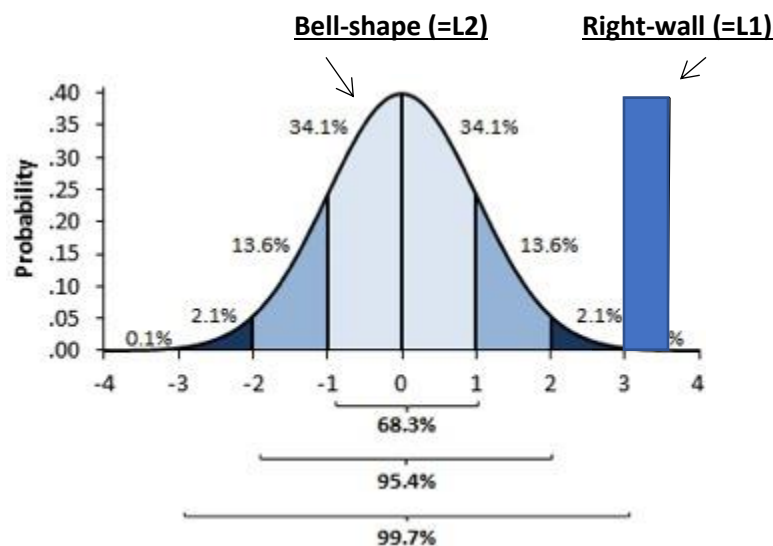


Whenever statistical averages of a competency of a certain Skill <x> are spread across a given demographic <y>, what one finds is a very consistent (probabilistic) average. This average, shown above as 34.1% / 34.1% (= 68.3%) on both sloping sides of the bell, indicates the average skill set of <x>. The largest subset of 'people studies' shows an average-level of competency for skill x, and this is consistent across all skills looked at. (Note: The stableness of this bell-shape ratio comes close to what we know of the Fibonacci 'golden ratio'). The scale here -4 to +4 could be

understood as *extreme incompetence* at the 'left-wall' (-4), while the **right-wall** (+4) shows a very *rare mastery*<sup>4</sup>.

What is so intriguing about the so-called 'right-wall' when it comes to a learned skill is that its extreme high-level of mastery mimics what we know of any biological-bases which governs learning. In fact, it's not overt 'learning' at all, but rather a state of **biologically determined** acquisition. It is in this sense that the terms 'learned' versus 'acquisition' makes its way into the L1 vs. L2 literature— namely, L1 (*pre-critical-period* native first language) is biologically determined and so does not suffer the competency spread found of bell-shape learning, while L2 (*post-critical-period* second language learning) shows bell-shape statistics.

[6] **Bell-shape for language 'learning' vs. Right-wall for 'Biological basis' language acquisition.** (Google© 'free-to-use' image).



<sup>4</sup> (See Stephen Jay Gould's *Full House: The spread of Excellence from Plato to Darwin* (1996) for discussion of the 'right-wall').

What we know of *biological-based* competency distributions is that they show mastery of the acquired endeavor at mastery levels only found at the extreme right-wall—viz., of what would be considered ‘the very rare extreme mastery level’ of 0.1%. What right-wall mastery shows is that amongst the general population (of all biologically healthy individuals) the statistical anomaly of, say, 0.1% actually becomes the normal average. The difference here is that *learned* vs. *biologically determined* accrue very different processing costs—namely, learning a skill is a *general problem solving* skill, cognitive in manner and follows all the classical IQ-dynamic hallmarks of ‘learning’ (e.g., asserting oneself in such a learning environments, note-taking skills, preparation, mnemonic devices for memorization, etc., and other strategies for learning such as motivation, aptitude, as well as some physiological factors which might determine the rate and success of the attempted skill). On the other hand, *biologically-determined* acquisition accrues no cost—it comes for *free* (as part of human endowment).

So, it becomes interesting to us that the right-wall of competency only shows-up across a demographic when a biologically-based behavior is measured. This becomes important when we begin to measure linguistic intuition for first language (L1) as compared to second language (L2). Recall, that for our **four sentences**, the ability to process such recursive structure embedded in these sentence types takes on a right-wall grammatical intuition and acceptability (for L1). For instance, recall another example of the grammatical intuition that came when L1 speakers were asked ‘can eagles that fly swim?’ (so, what are we asking that eagles can do?). Recall, the L1 reply was 99.7% consistent across the board that what was being asked was ‘if eagles can *swim*’ and not *fly*. Such is a right-wall distribution on a par with any other biologically-determined processing.

It rather seems the kind of knowledge native L1 speakers bring to their L1 performance has little, if any, connection to IQ problem-solving skill capacity, motivation, or the like. In fact, it has been repeatedly shown that even low IQ children, who otherwise may suffer from general learning handicaps, seem to have their language competency completely intact and unaffected. (Even some severely mentally retarded children show little impact on their L1 language acquisition). This may be precisely because L1 is indeed innate acquisition (= biology), and not learning. Also, it becomes interesting that when the ‘*eagles sentence*’ is presented to L1 speakers, but just visually shown to them, the innate recursive processing is not immediately made apparent to them (many students initially stumble on which is the right answer). Perhaps, this is because reading is a ‘learned’ processing (unlike speech) and so it doesn’t necessarily map onto the internal language mode of processing. Interesting, once the L1 speaker says the sentence out-loud, and hears the construction via speech, the hidden-internal recursive mechanism becomes activated and immediately the L1 speaker instinctively knows that we are asking if ‘eagles can swim’ and not ‘fly’ (again, despite the surface-level phonology that indicates the first, closest verb as ‘fly’).

(Note how when sequential bell-shape curves spiral out and get spread out over a time span (evolution)—and then when something emerges along the way as some constraint or human barrier, or upper ceiling—that what we find is the horizontal spread of the bells becomes smaller and smaller (in longitude) until such a time that a right-wall develops (latitude). In a sense, the right wall is a natural outcome of a collapse of space and time, as some consequence of human capacity to statistical convergence).

### **L1 is biologically-determined (right wall) vs. L2 is learned (bell-shape)**

Recall, that regarding the four language modes (speaking, listening, reading, writing), only the former two are natural and biologically determined which bring on the right-wall distribution of competency. The latter two are artificial skills, hence, their bell-shape curve of competency. These latter two modes, which are culture-bound and must be practiced, rely on a kind of ‘frequency-effect’ for its level of competency. (Such ‘frequency-effect’ bases of learning are altogether reliant on memorization, among other cognitive strategies).

Let’s keep this dual distinction in mind when we come to discuss the intuitive grammatical judgment of Sentence 4a vs. 4b of Sentence #4 (restated here in [7]):

[7] ‘I wonder [what [that is]]... up there?’

- (i) [what [that is]]\_\_
- (ii) [what \*[that’s]]\_\_

The judgment is even more fascinating given that fact that (ii) [that’s] is abundantly more prevalent in the frequency-data as compared to [that is]. Still, despite the higher frequency of [that’s] as found in the Bell-shape data, the ruling against frequency and rather for structure (even when the structure goes against the frequency) suggest that a very different kind of an operating system (OS), using Artificial Intelligence (AI) terms, is being employed. (The theoretical linguist reminds us that language is indeed *structure dependent*, not frequency dependent).

The best way to test this is by simply asking a native English speaker: Which of the two utterances do you prefer, (i) that is, or (ii) that’s...? The latter is overwhelmingly approved above the former, perhaps for reasons that are not at all syntactic in nature, such as simplicity, ease of speech, economy, etc. In any event, the fact that (i) is close to 100% judged as the only possible structure (for sentence #4), a fact which flies in the face of a statistical/frequency-based analysis, suggests that a larger, and rather hidden deep-state structure is active below what we find at the surface level phonology. Couple this with the notion that close to 100% of native speakers come



to the same conclusion argues against a learned, bell-curve response and rather speaks to how such a deep structure, in Chomsky terms, is indeed a biological determined a right-wall. 'Language is biology through and through'...and may not be something that can be learned, as if learning to play the piano. This demarcation of *natural acquisition* (pre-critical period), as found in child first language acquisition, as compared to what we know of (post-critical period) *artificial* second language **learning** allows us to see the bell-shape curve for what it really is—a probabilistic *informatique* of math and statistics which we reach when observing competency of a non-innate learned skill.

## Note 2

---

### A Note on Chomsky's (2013) *Lingua* paper 'Problems of projection'

<page 44, between examples (17) and (20)>

[1]. *The most general case of lack of label is successive-cyclic movement.* [2]. *The intermediate steps are of the form  $\{\alpha XP, YP\}$ , where  $XP$  can be for example a *wh*-phrase with  $YP$  a *CP*.* [3]. *The syntactic object  $\alpha$  cannot be labeled, but it must be interpreted, if only for theta-marking.* [4]. *If  $XP$  raises, then  $\alpha$  will be labeled  $Y$ , as required.* [5]. *Therefore  $XP$  must raise, and successive cyclic movement is forced*].

[1]. [*The most general case of lack of label is successive-cyclic movement*].

### Labeling

Starting out with the theory-internal assumption that labeling comes out of a movement operation, which as a result forces **Dynamic antisymmetry** (DA) (cf. Moro 2000, Ott 2011), then, for example, two lexical items within a **set** {...} necessarily come without order. Unordered members (X, Y) make-up the set {X, Y}. So the set {X, Y} (where both are heads) retain their c-command sisterhood status and thus don't derive order. Or where {XP, YP} (where both are phrases) behave similarly without hierarchy, and don't derive order<sup>5</sup>. This has been sharpened by Chomsky's analysis that the theory internal and *ad hoc* configuration of Spec-Head no longer holds, and that the basic configuration is reduced to a {Head-Head} merge relation, whereby one of the two terms {H} must be complete (i.e., where its phi-features\* are already drawn from the lexicon, such as with nouns or verbs) (Chomsky BEA 2001 ms. p. 12).<sup>6</sup>

The interest here lies in the fact that one of the two Hs must be replete with features so that it may serve as a Probe in a **probe-goal relation**. In a sense, the traditional Spec-Head has been reduced to a probe-goal relation whereby a symmetrically and unordered merged set {H, H} then

---

<sup>5</sup> Adjunction processes of {XP, YP} which seem to show asymmetry (thought which still must force an interpretation) have been discussed in the literature (e.g., see BEA p.18).

<sup>6</sup> BEA 'Beyond Explanatory Adequacy' ms. *MIT working papers*. May 2001. \*Interpretable n, v-roots act as 'completed' Heads while uninterpretable Case and Phi features act as probes.

becomes an asymmetrical (ordered) pair  $\langle H, XP \rangle$ <sup>7</sup> and where the modified H (Head) seeks out a goal within XP. The question is how does this asymmetry establish itself?

[1a] Turning to the recent incarnation of the minimalist program (MP) (Chomsky 1995), perhaps the most essential property which has come out is the notion that all *syntactic objects* (SO) are the result of a (re)combinatory operation called **Merge**, whereby  $SO = \{X, Y\}$  is the mere result of an unordered binary merge of two items:  $SO =$  the unordered members of  $(X, Y)$ . In this ‘first-instance’ merge, SO renders two lexical items  $(X, Y)$  as an unordered set  $\{X, Y\}$  (where  $\{X, Y\} = \{Y, X\}$ ). Call this **External merge** (EM) since X remains independent of Y (they are not part of one another), and as defined by set  $\{\dots\}$  as being unordered, there is no intrinsic order to  $(X, Y)$ .

But this is not ‘Language’ as there is yet no hierarchy/order within the set—it still being a primitive membership composed of nothing other than ‘flat’ symmetric sister relations (in logic, what we call ‘logical-and’  $\wedge$ , as in the  $\wedge$ -expression: ‘I need to buy a and b and c and d’ where  $\text{set}\{a, b, c, d\}$  are unordered). Hence, this ‘first-instance’ EM of the set  $\{X, Y\}$  doesn’t necessarily give us syntax (but merely a string of items, a lexicon)—viz., there is no hierarchy of the kind required of syntax.

[1b] At the very minimum, what we need as a prerequisite for syntax is an ordered pair  $\langle X, Y \rangle$  (where X precedes Y). This renders the unordered {set} now as an ordered  $\langle \text{pair} \rangle$  (see fn. 40). Specifically, the problem with first-instance merge (EM) is that SO is yet to be labeled: viz., *there is a labeling problem*. ‘Labeling of  $\{H, XP\}$  requires that the **Head** (H) not be of the form  $\langle X, Y \rangle$ ’ (Chomsky, *Lingua* paper p. 46). In other words, SO cannot emerge as a singleton instance of an arrangement of two heads  $\langle X, Y \rangle$ . Some form of **Labeling** must ensure. In order for SO to be labeled, a ‘second-instance’ merge operation must take place creating **Displacement** (hence, labeling comes via displacement). While displacement yields a hierarchical expression, it does so in a unique way as specified by the theory: ‘Labeling differs from other notions in that it is not virtually detectable by direct inspection of expressions’ (Chomsky, *Lingua* 37). In other words, labeling via a movement operation is theory-internal.

---

<sup>7</sup> Using MP terminology, a **membership set**  $\{X, Y\}$  comes freely unordered, while a **pair**  $\langle X, Y \rangle$  is necessarily ordered. When movement/raising is shown of an element within a set (DA), we can speak of an ordered **recursive set** as  $\{X \{X, Y\}\}$ . (See Galasso 2016 for full account of Merge as related to the development of child syntax).

So, a theory-internal operation is needed that takes us beyond a mere grouping of items of a (first) **membership set**  $\{X, Y\}$  (say, of two equal heads) to a (second) **recursive set**  $\{X, \{X, Y\}\}$  (or pair  $\langle X, Y \rangle$ ) which establishes an ordered syntax—noting that  $\{X\}$  of  $\{X, Y\}$  raises and gets displaced from its original membership set, this creating hierarchy. Call this displacement operation **Internal merge** (IM) of **H** since one of the two Heads (H) displaced must leave a **Copy** of itself behind. This forms a recursive structure  $[SO = \{X_i, \{X_i, Y\}\}]^8$ . At this point, the SO can be labeled by the raising of the H  $\{X\}$  of the **Phrase** (P)  $[XP \{X \{X, Y\}\}]$ . Ordering as seen within sets  $\{X, Y\}$  has now emerged as an ordered pair  $\langle X, Y \rangle$  (x comes first, then y).

[2] [*The intermediate steps are of the form  $\{\alpha XP, YP\}$ , where XP can be for example a wh-phrase with YP a CP*].

In this phase of derivation,  $\{\alpha XP, YP\}$  again are sisters, and properly make-up the equivalent membership set of  $(XP, YP)$  with no order. Both are potential (independent) projections which must then get later defined based on the actual (spell-out) projections of their contained lexical features. So, to suppose that XP is a ‘wh-phrase’ and that YP is a CP (which houses the wh-phrase) is tantamount to saying the ‘projection’ is one thing, and ‘that which projects’ is another—though the two are intertwined as specified by what type of P can project what kind of X (where X = H(ead), and XP is P(hrase) headed by X. Phrases are projections of Heads as Heads are bundle of features). For example, Chomsky addresses the fact wh-expressions can also remain *in-situ* without raising (and therefore may in fact not constitute CP), as in the example: *They thought JFK was assassinated in which Texas city?* (so-called ‘quiz-show’ structures) (p 44).

But the problem here is about projection of both [wh-expression & CP] when they form part of a raised constituency, as is found within an intermediate step along successive cycling: viz., if both  $\{XP, YP\}$  are sisters of a set at some intermediate step in the derivation, then how does labeling generate necessary projection? How does the SO get labeled? In order for a P to project, it must be defined by a set of features specific to its H. So, a potential symmetry of  $[[[wh-Q]...],$

---

<sup>8</sup> Chomsky tends to distinguish the site of the labeling of a syntactic object (SO) as either being placed at some point after movement (IM) (as with successive-cycling movement), or at the place of a merge-based/base-generated point of the derivation (EM)—with the former (IM) XP bearing the subscript  $\{\beta\}$ , and the latter (EM)  $\{\alpha\}$ . E.g., where raising of XP is involved and where a copy of itself is found in the lower structure, the use of  $\beta$  is employed  $\{\beta XP, YP\} = [XP [\text{copula } [\{\beta XP, YP\}]]]$ , as opposed to  $[\alpha N \text{ TP}]$  where NP is the product of simple merge. Take  $\{\alpha... \beta\}$  to form a chain of  $[SO \{\alpha... \beta\}]$  (or  $\{\gamma.. \{\alpha.. \{\beta\}\}\}$  that spans three domains). For example,  $\{\alpha NP_i, \{T, \{\beta NP_i, \{v, VP\}\}\}\}$ , the highest copy/instance of the NP is in the domain of the entire SO that is labeled  $\alpha$ , since every instance of the NP is within this domain. The lower NP instance, however, is not in the domain of  $\beta$  since not every instance of this NP is within this domain.

& [[CP]...] = {XP, YP} cannot stand. One of the two Ps must raise in order to form DA (as discussed citing Moro, Ott). One question here is why shouldn't YP raise if both {XP, YP} are equal-distant sisters. The raising conditions behind DA should allow either of the two terms to raise. But it seems only XP (the wh—phrase) raises. (Chomsky expresses this concern in his footnote 36: 'One may ask why YP doesn't raise'). Perhaps the notion of 'computation atom' expressed within the lexical item (LI) has a determining factor behind why the XP (=wh-phrase) of the set {XP, YP} raises leaving {YP} (=CP) without recourse of IM. The CP may be a vacuous phrase (without a head) unlike the wh-expression which is headed—creating an inherent DA within the set even before potential IM takes place for labeling of SO.

Chomsky cites Moro here regarding copular/Small Clause (SC) constructions—e.g., [be [lightning, the cause of fire]] where one of the two terms of the SC must raise so that the SO can be labeled. Hence, DA comes out of a need to label the SO projection.

A very nice example of this is what we find in otherwise symmetric {X, Y} configurations. For example, consider the SO to be of the two lexical items, the membership set: {boat, house}. As it stands, the SO cannot be labeled. What is needed is **raising** thus providing a syntactic hierarchy via a DA-set. (Raising thus renders an unordered membership set into a recursive set). As Moro claims, one of the two terms must raise. If {boat} raises from {X, Y}, we get {boat, {boat, house}} = {<sub>β</sub>X, {X, Y}} where the lower X copy is invisible to pronunciation, it being a discontinuous element. Hence, the SO gets labeled and projects as [boat [~~boat~~ house]] (and where hierarchical syntax is generated that allows us to interpret 'boat-house' as a kind of 'house' and not a kind of 'boat'). Such N-compound structures are likened to Adjectival Phrases where the H of the P labels the projection—e.g., [AdjP [Adj black] [N bird]] whereby 'black' is the Head of the phrase, so that the interpretation is that a 'black-bird' is a kind of 'bird' and not a kind of 'black'.

[3] *The syntactic object  $\alpha$  cannot be labeled, but it must be interpreted, if only for theta-marking.*

[4] *If XP raises, then  $\alpha$  will be labeled Y, as required*

One of the questions regarding 'performed operations on syntax' is essentially 'how does the algorithm operation work? (Chomsky refers to the operation as a labeling algorithm (LA). The problem of labeling is inherent in language, bottom- up vs. top-down solutions have been jousted from the very conception of the problem: viz., (i) in order to label a word, it must be embedded in a top-down structure (language is structure-dependent), (ii) in order to generate a structure, there must be a well-defined status of word. (The catch-22 is similar to what we find in our

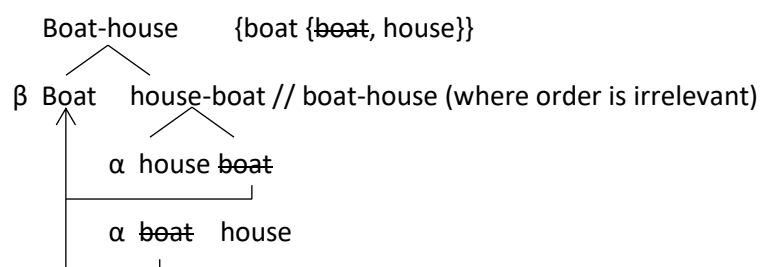
approximate understanding of the way genetics works: proteins create amino acids and amino acids create proteins, etc.).

Here is how the problem is stated. A labeling algorithm (LA) must scan a syntactic object (SO) and provide a label for one of the two (within a binary selection). So, suppose  $SO = \{H, XP\}$  ( $H$  = head,  $XP$  = non-head) is rendered via the **external merge** (EM) of two items. Then LA will select  $H$  as the label. In other words, the feature specificity for labeling is already encoded in the pair, and projection is straightforward since the  $H$  is visible to the LA—e.g.,  $H$  = verb, and  $XP$  = noun complement,  $[VP [V, NP]]$ . But what if LA scans two sisters  $\{X, Y\}$ , or  $\{XP, YP\}$ , where both terms are non-heads? Here, the search is ambiguous. It is in this case, when the label  $\{H\}$  is not made available in the search, that the MOVE-property of raising (**internal merge** (IM)) must establish an hierarchical configuration, breaking symmetry. So, consider  $\{XP, YP\}$  as a mere membership set. In order to break with symmetric sister-relations, (since both reside within a flat structure), the raising of one of the two terms must proceed, yielding  $\{XP, \{XP, YP\}\} \Rightarrow [XP [\cancel{XP}, YP]]$ . A pair is created with derived syntax (order)  $\langle XP, YP \rangle$ .

### Antisymmetry in Compounds

Consider  $\{X, Y\}$  as two lexical items: e.g., the twin-noun formation of  $[N, N]$  leading to the compound  $\{\text{boat}, \text{house}\}$   $[N \text{ boat-house}]$ . In order for LA to label one of the two SOs as the (adjective-like) **modifier** (M) (in marking the compound ‘boat-house’ vs. ‘house-boat’) one of the two SOs must raise in order to break the symmetric reading. In other words, the labeling  $\{N, N\}$  would force one of the two  $N$ s to raise thus allowing the  $M$  to project, and as a result, allow the  $H$  to be labeled accordingly:  $\{N [\cancel{N}, N]\}$ , where the moved item acts in an adjectival  $M$  capacity and where the residual unmoved item becomes  $H$  by default. As we see, what is different about lexical items/*compounds* as compared to English *Phrase* types is that Heads of an **English Phrase** are **Head initial** e.g.,  $[VP V [\cancel{V}, N]]$ , whereas Heads of English lexical **compounds** are **Head-final** (as opposed to Spanish compounds where they are **Head-initial**—e.g., *comi-ratas* (eater-rat) (= rat-eater). In the examples below regarding ‘boat-house’ what needs to be selected is not  $H$ , but rather the modifier ( $M$ ) of  $H$ . The item which stays in place, not a result of MOVE, will thus be labeled as Head. The item which raises will become the Modifier of  $H$ .

[5] boat-house (a kind of house)



Within the flat structure [5α], order is symmetric and irrelevant. MOVE (IM) (raising) is a unique displacement property which allows for recursive structures found in [5β] above, thus providing a mechanism for labeling the head of a phrase. A ‘boat-house’ is a kind of ‘house’ (and not a kind of ‘boat’), where order becomes crucial to the compound’s reading.

*Where X modifies the head*—This same formation is exactly what happens in Adjective Phrase (AdjP) structures: So, compound antisymmetry looks something like the following {X {X, Y}} where {X} gets labeled as M, and as a result, where in-situ Y takes on a head status. In the N-compound ‘house-boat’, the H of the compound is the N ‘boat’ (= a kind of boat).

[6] red car.

Where the two lexical items {X, Y} [X, Y] = [red, car]. In this formation, no word order is realized since both items are sisters and are thus ambiguous in their reading. It is not until some raising takes place via IM that order can be assumed: [red, car] > [red [~~red~~, car]] = [AdjP red [~~red~~, car]]<sup>9</sup>.

### **Compounds: Root vs. Synthetic (cf. §1 [15]).**

[7] a. coffee-maker = (maker of coffee) => [coffee-[maker of ~~coffee~~]]

cigarette smoker = (smoker of cigarettes)

=> [cigarette-[smoker of ~~cigarettes~~]]

b. chain-smoker (not \*smoker of chains)=> \*[chain-[smoker of ~~chains~~]]

Where examples in [7a] (referred to as **Synthetic compounds**) require *double merge* via displacement: [coffee-[maker of ~~coffee~~]].

And where example [7b] (referred to as **Root compounds**) is the result of a *single merge*: [chain-smoker]

In [7b], the adjectival root is not derived via displacement (notwithstanding the fact that some first-order displacement had to have been initially carried out in order to derive the order {chain, smoker}, since the two items must derive an ordered pair: we don’t equally get \*[smoker,

---

<sup>9</sup> We may freely substitute {}-brace brackets with []-square brackets here as the latter is typically used in syntactic tree diagramming e.g., [VP [V, N]].

chain}. Hence, root compounds begin their journey with at least a first-instance merge of a recursive set, but they do not go beyond that in forming further displacement up the tree.<sup>10</sup>

In sum, it does seem that at least some amount of movement is involved with the formation of all types of compounds. For one, the placement of the Head of the compound follows similar parameterization constraints which are found in syntax (e.g., the Head Parameter [+/- Head initial]). As well, the fact as shown above that *synthetic compounds* are the result of MOVE (which contrasts with *root compounds* which are not derived via MOVE) suggests that displacement in language is of a morphosyntactic nature.

(See web-link no. 27. for movement in compounds).

(See [https://www.academia.edu/34403441/Working\\_Papers\\_4](https://www.academia.edu/34403441/Working_Papers_4)).

### ***Dynamic Antisymmetry***

Following a version of Andrea Moro's dynamic antisymmetry (DA) (as presented in Chomsky's 2013 *Lingua* paper), let's consider that we have reached a point in the derivation where [XP, YP] have already been formulated, leaving us with only two Heads {if, how} to contend with, both having identical labeling. What DA stipulates is that in order for one of the terms [XP, YP] to be labeled and associated with the right head, one of the two [XP, YP] terms must raise.

So, if Syntactic Object (SO) = {XP, YP} where both Ps are identical in terms of labeling (both are equal phrases), in order for labeling>projection to be properly headed, one of the two terms (X, Y) would have to raise, thus breaking a symmetrical flat-sister relation.

Consider the [copular-small clause] structure below, where the small clause is of the form [XP, YP] (cited in Chomsky 2013 p. 43 (taken from Andrea Moro's work on *dynamic antisymmetry*)):

[copular (H) [small clause {XP, YP}]] , SO = [XP, YP]

---

<sup>10</sup> But this is based upon a theory-internal stipulation which states that 'everything must move at least once in order to become visible as part of labeling'. Accepting this, there still remains the basic assumption that root-compounds are indeed base-generated and are not derived via movement. In Galasso (2016, p. 79), drawing on the '*everything must move at least once*' stipulation, this first-instance of move—as shown in root compounds, as well as all phrases – is referred to as **first-merge (Move-1)**. A reference to **second-merge (Move-2)** then is made for subsequent second-level raisings which yields syntactic compounds.



[8] [be [lightning, the cause of the fire]]

It seems the structure above would yield an ambiguous result if search were applied as is. In other words, in order to label SO (small clause), one of the two terms must raise, breaking its symmetric/sister relation. In this projection, the copular 'Be' triggers one of the two terms of the small clause to raise from out of the identical [XP, YP] labeling:

- [8']    a. [XP lightning *is* [XP lightning, YP the cause of the fire]]  
         b. [YP the cause of the fire *is* [XP lightning, ~~YP the cause of the fire~~]]

Another salient example to this structure would be the even more stark case of how one of the two terms in an otherwise identical [XP, YP] has to raise.

Consider the structure below:

- [9]    a. *If you keep eating ice-cream, **how** can you lose weight?*  
         b. How can you lose weight if you keep eating ice-cream?

The two heads (H) involved in the labeling of [XP, YP] are [H {how, if}]. Where XP, YP are given:

[XP you keep eating, YP can you lose weight]]

Noting here how both clauses {XP, YP} can be inverted as long as they remain headed by their proper **polarity-item** head (H)—in this case, the H 'if' triggers the [XP you keep eating ice-cream], while the H 'how' triggers the [YP can you lose weight]. Here's how antisymmetric raising might work in such examples (which show an inverted [YP, XP] for illustrative purposes only, since the order of XP, YP is irrelevant within its flat sister-relation [X, Y]):

- [10] [[XP you keep eating] [YP can you lose weight]] (where both terms [XP, YP] are identical in label).

Notice their *symmetric* quality: (how can you lose weight if you keep eating?  
> if you keep eating, how can you lose weight?)

Now, what must happen to break this symmetry is that the head must select one of the two terms [XP, YP] to raise/MOVE, thus allowing for the proper labeling of the phrase—where the two Hs are {H {if}} and {H {how}}.

(a) [**if, how** [XP you keep eating, YP can you lose weight]]

(a') [**if** [XP you keep eating **how** ~~[XP you keep eating~~, YP can you lose weight]]]

(b) [**how, if** [XP you keep eating, YP can you lose weight]]

(b') [**how** [can you lose weight] **if** [XP you keep eating, ~~YP can you lose weight~~]].

Of course, we quickly notice that the order of the Heads {how, if} too is irrelevant, with the same projections that follow:

(c) [**how, if** [XP you keep eating, YP can you lose weight]]

(c') [**how** [YP can you lose weight] **if** [XP you keep eating, ~~YP can you lose weight~~]]

So, it appears based on our discussions of dynamic antisymmetry DA so far, that it is the head (H) which defines and projects all syntactic objects (SO). All relevant information about an SO, a lexical expression, will be provided by a single designate element within it—call it a ‘computational atom’, which defines the H. So, what is the triggering mechanism involved? It seems the H ‘if’ has encoded in its lexical head a collection of *computational atoms* (i.e., features) that requires ‘if’ as a polarity item to trigger raising of only the declarative [XP you keep eating]=> [if [XP you keep eating]], while preventing the projection of \*[if [can you lose?]] (where only the H interrogative item ‘how’ can trigger an auxiliary inversion derived from the lower matrix phrase [how [YP can you...?]] ([\_ \_ [you can how?]]).

Connected to this, consider how one might parse an already formulated expression such as the now infamous Pink Floyd saying (song ‘Another brick in the wall’ from the album ‘The Wall’). Imagine there are two ways to say the saying, of course, depending on which H you select first:

*<If you don't eat your meat, you can't have any pudding! How can you have any pudding if you don't eat your meat?>*

Again, notice how each polarity expression Head (if, how) selects its own complement phrase via antisymmetric raising. Consider how DA might go about determining the sequence of the Pink Floyd saying. Recall, at this point, the two Hs in question are symmetric (due to their sister relations) and either one or the other could be selected to start the complex sentence: {H, H}, or {if, how // how, if}.

So, the question we turn to now is to ask: What is the mechanism involved whereby the H selects its appropriate XP?

### ***Head as Computational Atoms***

What Chomsky suggests here is that each specific H carries a ‘collection of atoms’, minimal elements of the H which enter into a computation—‘relevant information about SO, a lexical item, will be provided by a single designated element within it, call it a ‘computation atom’ which defines H. Hence, a lexicon is a class of atoms to be computed (p. 41). Let’s consider that the two heads {H, H} [if, how] carry their own special properties of atoms which engage in a computation. Suppose the H {how} carries an anaphoric copy of itself which must be found somewhere within the search of [YP], as in [How...copy]. So, one such atomic feature of {how} can be expressed as follows:

[11] {how<sub>j</sub>....copy<sub>j</sub>} => [how.....[if.....how]]

(along with other traditional features which go into the projection of the wh-element {how}, such as {+interrogative}, {+ aux inversion}, {+polarity express}, {+adverbial/manner}, {CP-projection}, etc.).

Consider below the search mechanism for {how<sub>j</sub>....copy<sub>j</sub>} => [how.....[if.....how]]:

[12] {**how**}

[if, **how**<sub>j</sub>] search => [XP [you ~~can~~ have any pudding ~~how~~<sub>j</sub>]]

How => [XP can [you \_\_ have any pudding\_\_ ]]

How => \*[YP you don’t eat you meat]

**\*how** you don’t eat your meat **if** you can have any pudding.

The H {how} must select appropriate XP {how<sub>j</sub>....copy<sub>j</sub>}.

Consider now {if}

[13] {if}

[if, how] search=> [YP you don't eat your meat]

If => [YP you don't eat your meat]

\*If => \*[XP can you ~~can~~ have any pudding ~~how~~<sub>j</sub>]]

In this case, the H {if} can't select an XP which contains the anaphoric copy {how} found within the scope of search, therefore, the XP becomes what is referred to as *discontinuous* and is not visible to labeling, discontinuous due to it already being either selected by H 'how', or by the fact that the most prominent feature [copy] can't enter into an Agreement (AGR) checking relation of {how<sub>j</sub>....copy<sub>j</sub>}.

So, what we are supposing is that since the H {if} carries no such anaphoric copy, but rather only features associated with H of C (e.g, the polarity condition of (if> then), as well as CP-projection), then part of the LA search becomes modified in order to secure SO labeling. What we can say is that while part of the formal checking requirements of {how} must search for a copy of itself, the computational atoms for H {if} require no such copy. It is in this sense that DA might be applied:

{if, how} [XP, YP] (where [XP, YP] are sister relations in the same sense that Moro talks about his small clause formations).

Notice again how the order doesn't apply to sequence of {if, how}. Since the XP, YP is already formulated (as part of knowing the Pink Floyd phrase), the only selection that needs to be made is which H you select in order to trigger the appropriate XP/YP raising:

[14] a. [If, \_ how [XP can you have any pudding [YP you don't eat your meat,]]

b. [If [YP you don't eat your meat,] how [XP can you have any pudding  
~~[YP you don't eat your meat]]~~

In sum: H {How} searches for a copy of itself, finds it in XP, then proceeds to raise XP. {if} doesn't raise XP because the copy of {how} would go unchecked. Hence {if} is rather forced to raise the other sister relation. It is in this case that {if} is less restricted than {how}, and where {how} must restrict itself to search out an anaphoric

copy within the matrix clause. In other words, {how} requires less search ambiguity over {if}. Heads {if, then} (cited below) however would seem to be more on a par with each other (since neither contain a copy of itself in XP). What then drives a particular labeling perhaps has to do with logic.

[15] What of the Hs {if, then}?

The [if...then] checking sequence would work as follows:

e.g., **If** you eat your meat, **then** you can have pudding.

But notice how the two heads can't seem to be reordered:

\*then you can have pudding, if you eat your meat.

This would mean that the feature attributed to the H {if} requires it to be an antecedent (in first position) to any anaphoric expression [if...then] => [if..[then]], where the two Heads are not symmetrical/identical in nature (they are not sisters). This is different from what we saw with the Hs {if, how} where both were symmetric sister relations [if, how] (though where {how} contained an anaphoric feature {+copy}).

e.g., [if, [then]] [XP you eat your meat, YP you can have pudding]

It may be understood that the H {then} of the {if, {then}} sequence searches for the prominent element {can} within XP, where {can} contains a conditional element related to the lexical item {then}.

If X, then you can Y: {if X, then Y}. => logical structure

It was this same kind of underlying thematic/logical structure which motivates Move to take place on a Semantic/Thematic level. Recalling that Move/displacement up until now has had two motivations:

(i) syntactic (S), and (ii) phonological (PF).

Well, it may be the case that Move also caters to two sub-types of S structure: (i) syntactic (surface-level, PF) and (ii) thematic/semantic (underlying logical form, LF).

[16] ***Reasons for Move/displacement:***

- (i) Phonological (PF)
- (ii) Syntax (S)
  - a. Syntactic (CP)
  - b. Semantic/Thematic (vP)\*

---

\*Note. See the end section of this note ‘Family of Merge’ for a brief discussion of the **Duality of Semantics**. Following Miyagawa (2000), we consider the two Phases of **CP**, and the light verb **vP** to be the phases which serve this duality of semantics—viz., with the highest *functional* projection of CP dealing within the probe-goal relation of AGReement, and the lower *lexical* projection of vP dealing within the probe-goal relation Case. We’ll come to consider Case as being somewhat lexical/semantic in nature, presumably loosely associated with theta marking, while we’ll consider AGR as being quintessential formal in nature. (Recall that in a phase-base theory (Chomsky 2008), TP is not considered a phase. But we may entertain the notion that a T-feature [T] may either serve as a featural {F} or Affixal {Af} adjunct which can adjoin to either TP or CP (along with affix lowering onto the V), and that as T(ense)/TP is not phasal in nature, the T-feature itself may be free to *percolate* up or down the syntactic tree). We’ll propose, loosely following e.g., Miyagawa (2010) and Radford (2009, 2016) in a number of respects, that the duality of semantics, as pegged to Chomsky’s notion of Phase-theory (where only CP & vP are considered phases), that the light verb **vP assigns Case** (Case being lexical semantic in nature) and **CP assigns AGR**.

---

Let’s consider below this duality with respect to ‘reasons for MOVE’: semantic or syntactic?

***Semantic Move.***

Consider Chomsky’s (2013) remarks below:

- [17]    a. Which books did John read => S/PF
- b. ‘For which books X, John reads books X’ => semantic/thematic.

One of the reasons for Move/displacement here seems to be to get the semantic roles correctly assigned. For example, here the XP {which books} has two semantic roles: it receives its role as object of {read} (as in the expression 'read books'), and it also serves as a distinct interrogative operator, binding the variable in the object position, so that the interpretation is something like 'for which books X, John reads books X'. Showing base-generated structure, then movement, the structure looks like this:

- b. ['which books did [John read ~~which books~~]]?

In addition to two types of Move: *Internal & External Merge*, Chomsky identifies two types of merge: *Copy & Repetition*. Consider the **repetition-structure** of the phrase 'What hit what'?

[18] What hit what?

It seems that the lexical item 'what' takes on two independent arguments {X, Y}, they are repetitions of the PF spell-out, but both are independent: consider the argument structure:

- a. [HIT [x, y]] = X hit Y...e.g., 'Boy hit ball'

Compare (a) above to the **copy-structure** below:

[19] What was hit (what)?                      (John hit what?)

In [19], 'what' is a copy derived by [\_\_ was hit what] (= ~~What~~ was hit what?) with Internal Merge (IM) raising of 'what' to the surface subject position. Consider the underlying structure:

- a. The ball was hit  
i. \_\_ hit the ball. (John hit the ball).

In terms of IM, the *raising of a copy* seems to be a **syntactic effect**, whereas *raising of repetition* seems to be a **semantic effect**. This same semantic/thematic vs. syntax split regarding merge can be seen in:

- b. Which books did John read?

Here in (b), ‘which’ takes on two independent semantic roles (via repetition):

- (i) Object of ‘read’
- (ii) Interrogative operator

For ‘Which books X’, ‘John read books X’.

A further complication might actually be suggested that Copy (IM) serves a two-fold operation:

- [20] (i) can serve **semantics** when dealing with **Case** (argument/thematic marking), as shown in (b) above, and/or when dealing with a light verb (vP) projection..., or
- (i) It can also serve **syntax** when dealing with **Agreement**, when dealing with a CP projection.

Hence, what we get out of Internal Merge (IM) is a potential **double-merge** operation of (i) semantics, followed by (ii) syntax.

Copy as IM:

- (i) Semantic (vP): Case
- (ii) Syntax (CP): AGReement

In sum, what the above dual treatment of merge (IM) details is that while Repetition is triggered by a single-merge (exclusively for semantics), Copy may trigger double-merge (both for semantics (vP-Case), and for syntax (CP-AGR)).

(See Miyagawa (2010) for such an account of a Case/AGR double-merge. In fact Miyagawa goes even further and suggests that the entirety of Move is based solely on AGR).

### ***Syntactic Move.***

Let’s follow-up on Move as having a syntactic effect. The quintessential notion of syntactic move has to do with Agreement (AGR) (following Miyagawa here)—which is a movement-based operation triggered for no other apparent reason other than to check off a formal AGR feature. This is what we find in most instances where the higher non-argument position of CP is involved.



### **AGR-based Move.**

T(ense) only carries the set of (person, number) AGR features in a clause (XP) where T is selected by C of a C(omplementizer Phrase) (Radford 2009, p. 397).

What this means is that the Head C (as part of its computational atoms) carries and 'hands-over' AGR features from H of C to H of T.

- [21] (a) I am hoping [CP [for] [TP him to win]] [-Tense/-Case]  
(b) I am hoping [CP [C that] [TP he wins]] [+Tense/+Case]

Let's flesh such a treatment out in terms of DA and H computational atoms as discussed above.

Suppose the two Heads are {H, {for, that}}, and the {XP, YP} are [TP {him to win}] and [YP {he wins}].


What the search reveals in labeling the SO is that the H {for}, carries no Tense/AGR features (from C) and so only the default non-finite verbs along with the default accusative case get selected. In (b), it is the well-formed H of CP {that} which selects and labels for AGR (Tense/Nominative Case). What we have regarding the labeling of XP in terms of the H is precisely the same type of selection as discussed in our treatment of dynamic asymmetry:

- [22] {for, that} [XP him to win, YP he wins]


Both {XP, YP} are identical. So in order to label the SO, one of the two terms must be modified by raising.

[for, that] [~~him to win~~, he wins]

(i) [for [him to win]] => search [for him to win] [XP ~~him to win~~, (YP he wins)]



(ii) [that [he wins]] => search [that he wins] [XP (him to win), YP ~~he wins~~]



H {for} carries computational/atomic features [-T, -Case] => [-AGR]

H {that} carries [+T, +Case] => [+AGR].

## Dynamic Antisymmetry & Recursiveness in Possessives {‘s}, {of}

### John’s book vs. Book of John’s

Consider the structure {John {Poss, book}}, [John [Poss, book]]. Here John [J] can raise either of the two items [X, Y] (X Poss, Y book). If J selects (via search) {Poss}, leaving {book} in place, we get the underlying structure [John [‘s book]] => [John’s [\_ book]] where clitic {‘s} raises and attaches to the stem (a result of DA movement due to PF considerations, (as presented above in §1 [2], but see §4 [10] where PF clitics don’t apply).

If, however, J selects to raise {book}, leaving {Poss} in place (so that it remains frozen in-situ and therefore can’t become a clitic), we then get [book of [John [‘s, ~~book~~]]. The double possessive elements found in expressions such as *I am a friend of John’s* can be traced back to how the H John selects either *Poss* or *Book* to raise, following similar steps as laid out in our discussion of Dynamic Antisymmetry which shows DA merge as [H [X, Y]].

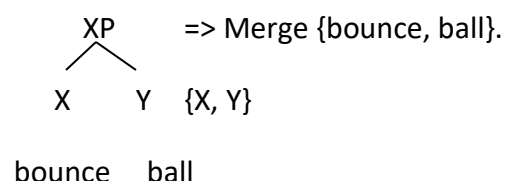
### Dynamic Antisymmetry of Internal Merge: Closing Remarks

Various approaches have pinned DA to models which stipulate that ‘every element must move at least once’ in order to be visible to a labeling algorithm (LA). This reduces to meaning that every right-branching structure must end in a trace (See Kayne (1994). Hence, complements (object DPs) must vacate their base-generated positions. Added to this movement stipulation is that subjects (DP) too must move out of VP, coupled with other language-specific notions determining whether or not a verb must raise to H of T, (or if T affix lowering down to H of V is required, as in English). All in all, the notion of movement, as it entails all the aforementioned operations, is without doubt the crucial mechanism at play in determining fully-fledged human language capacity.

### A Note: Family of Merge

As part of our closing remarks, let’s recap how the step-sequences of Merge (both internal and external) come together to form a fully articulated syntactic tree, particularly focusing on **Case** and **Agreement**, (or semantics & syntactic, which make-up the so-called **duality of semantics**):

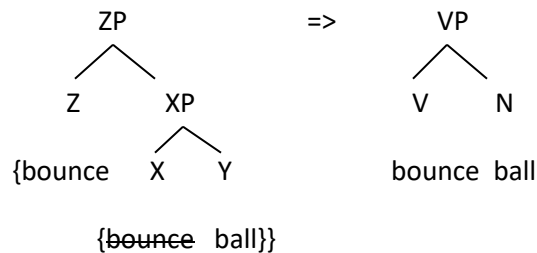
(1) Step-1: External Merge (EM)



At step-1, there is no hierarchical word order. XP could yield either 'ball bounce//bounce ball'. Step-1 is exclusively semantic, whereby two items have simply been pulled from out of the lexicon.

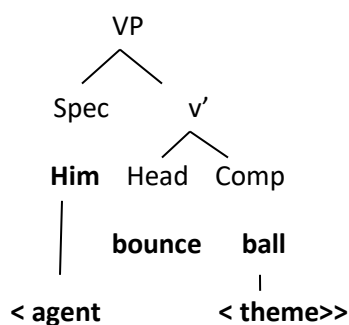
In order to create a hierarchical syntactic phrase, Move is required (our step-2):

(2) Step-2: Internal Merge (IM):



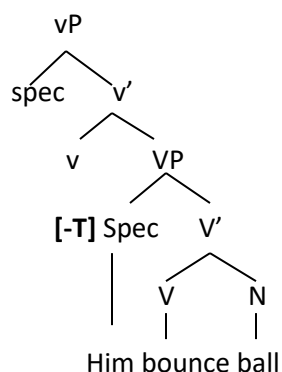
At step-2, we get Internal Merge (IM) (with a non-pronounced copied movement of 'bounce') which now allows order via dynamic asymmetry (DA). It is here that we arrive at a properly formed Spec-Head-Comp. This is a prosaic lexical/semantic projection still without Case, Tense or Agreement.

(2') Spec-Head-Comp configuration (Theta-marking):



Theta marking is a semantic assignment.

(3) Step-3: Light Verb (vP)



The [-T] tense feature may assign the sort of tense

we find with [-Finite] imperatives, subjunctives.

Note no Case, AGR at this step.

At step-3 above, what this next merge projection allows is an added available Spec position for the subject of the lower VP to raise into. Subject/Spec of VP, once inserted into Spec vP, gets assigned Case [+Nominative]. Light verbs can assign Case, since vP is semantic in nature and Case is a residual effect of semantics. (vP is said to straddle both lexical and functional projections. Unlike VP which can only map a theta-grid, the light verb vP projection takes the derivation one step further in securing Case. While light verbs such as *make*, *do*, project alongside the main verb e.g., 'John *makes* roll the ball' (= John  $\emptyset$  rolls the ball), such light verbs, when stranded within vP, can't project Tense). Light verb formations arguable give us the antiquated **subjunctive** mood (which particularly contrasts with the **indicative**) whereby *nominative Case is assigned without finite Tense*—e.g., the subjunctive sentence: '[I'd suggest that **he study** for the exams]]') and not (\*'He studies...').

(We'd suggest that subjunctives have a fully functioning vP with a somewhat defective TP, whereby case gets assigned by the above [featural T-feature] within T, but where the [affixal-T] feature of the Head is unspecified.)<sup>11</sup>

But before Case can be assigned, it must be handed-over by a [+T] feature (Case is marked via Tense—[+Finite] tense marks [+Nom] case, [-Fin] tense marks [-Nom] case (see below).

### Case marking via Tense:

- (i) Finite tense assigns [+Nominative] case:

e.g., I think [he walks often] : He => walks

- (ii) Non-Finite tense assigns [-Nom]/Accusative case:

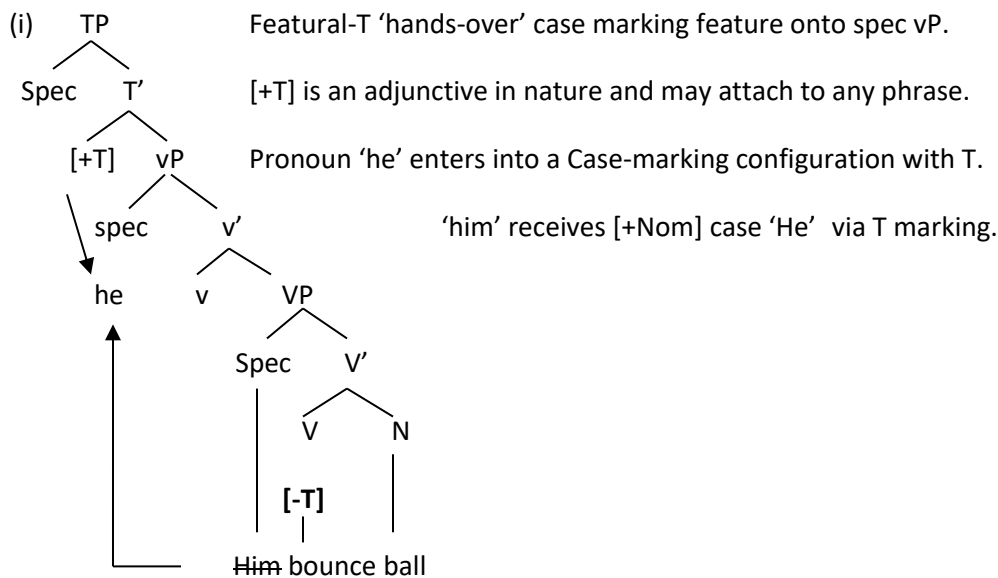
e.g., I saw [him walk often]: Him => walk

Since vP can't mark for [+Tense], in order for vP to assign case, a TP must project whose Head projects finite tense. Consider the next step-4 below:

---

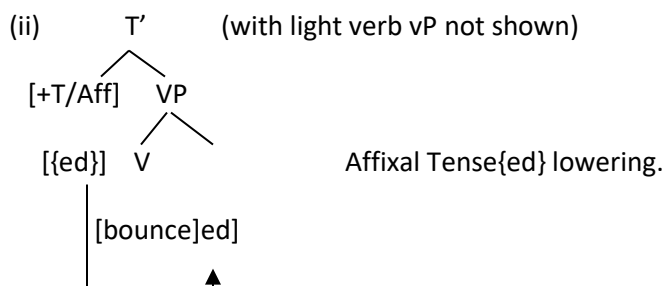
<sup>11</sup> One distinction that can be made between featural vs. affixal is that featural may encode tense as incorporated in the verb stem itself, perhaps as in French, (or English auxiliary and irregular verbs), whereas the affixal-feature is a decomposition of stem+affix.

#### (4) Step-4: Tense Phrase



#### Tense treelet structure: (past tense).

When Tense projects [+Past] {ed}, along with Case marking, we now find affixal-T lowering, as [Stem + [affix]].



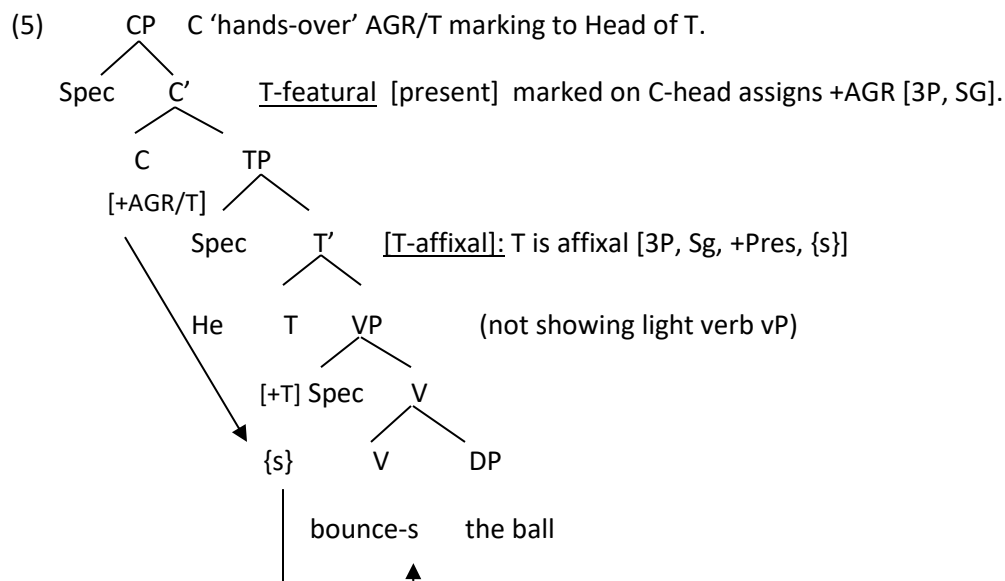
Once Case and Tense are projected via vP, TP respectively, the question now turns to the more formal projection of AGREEMENT (since the two lexical/semantic assignments have both been satisfied). Following some of the remarks made in Radford (2009) about AGR being assigned via C-head of CP, as well as his more recent remarks that C-head may in fact house Tense (with the T-feature assigning AGR down to TP), we similarly suggest the extension, that, like what we find of **wh-subject constructs** (e.g., 'Who found it?'), all declarative TP projections are in fact fully

extended CP projections. By proposing that CP assigns AGR, coupled with vP which assigns Case, we arrive at the **duality of semantics** with the added feature that a **phase-base** theory (MP) also incorporates these two phases: CP & vP—with (i) CP being formal/syntactic in nature, and (ii) vP being theta/semantic in nature.

The duality this way is captured by the lexical vP vs. functional CP split, with Case being a residual lexical phenomena and AGR being quintessential formal. In fact, Miyagawa (2000) goes even further by speculating that it is AGR which is solely responsible for MOVE.

Let's consider the last step showing a full CP-articulated structure:

'He bounces the ball' (showing 3P, singular {s} (AGR)):



**Subject Agreement** (Radford 2016, p. 338):

When T agrees with its Subject/Spec:

- (i) The person/number features of the subject are copied onto the Aux/affix T. (We are adding that the AGR projection is instigated by C).
- (ii) The tense feature [T] on T is copied onto the subject.

With this treatment, CP 'Wh-subject' constructs of the 'Who bounces it?' type (where the wh-word 'who' must raise to spec of CP) mimic what we find with declarative TPs ('He bounces it'): both are full CP projections since, in this analysis, the AGR of the verb e.g., 'bounces' (of 'He bounces it') can only be assigned via 'hand-over' of C-head [+AGR/T] to T.

### Featural vs. Affixal T.

The assumptions made above is that T(ense) comes in two forms: (i) featural and (ii) affixal. When featural, since T is adjunctive in nature, theoretical considerations could be made that T can be found adjoined to any appropriate phrase (not just TP). (Radford (2016, p. 338) goes further to suggest that even DP/Nominals (Nouns) can theoretically take-on a tense feature). Tense has been problematic in the MP literature for quite some time: for instance, it seems to be a [+Interpretable] feature, thus semantic in nature, while, at the same time it seems to be implicated both in Case and Agreement. And Chomsky considers Tense not to be a phase.

What this may call for is the notion that T, when functioning as [+featural], a featural-T can percolate up and down the tree without impunity (when inserted at the **edge** of any phrase) and enter into **probe-goal** relations of various kinds (as determined by the nature/feature specificity of the host head). For instance:

- (i) when T-featural is adjoined to C, it seems to be implicated in securing AGREement of the subject-verb accord,
- (ii) when T-featural is adjoined to T, it delivers two functions:
  - a. it can deliver Case (as found with light verb vP),
  - b. it can deliver a non-affixal Tense (such as with bare verb stems, auxiliary do>did, have>had, is>was, as well as irregular verbs go>went, etc).
- (iii) when T-affixal: {ed}, {s} [+/-past] affix lowers onto the verb stem.

### **Note 3**

#### **A Note on ‘Proto-language’: A merge-based theory of language acquisition—Case, Agreement and Word Order Revisited**

*·Language = Recursion, which is ‘recently evolved and unique to our species’—Hauser et al. 2002, Chomsky 2010.*

*·If there is no recursion, there can be no language. What we are left in its stead is a (Merge-based) broad ‘beads-on-a string’ sound-to-meaning recurrent function, serial sequenced, combinatory non-conservative and devoid of the unique properties of recursion which make human speech special. It may be ‘labeling’ (see Epstein et al.)—the breaking of ‘combinatory serial sequencing’ found among sister-relations—that constitutes the true definition of language since in order to label a phrase one must employ a recursive structure—JG.*

*·If Continuity is allowed to run freely, in all aspects in respect to biology, and is therefore the null hypothesis, then what we may be talking about is a ‘function’ that matures over time, and not the ‘inherent design’ (UG) which underwrites the function, since, given strong continuity claims, the design has always been there from the very beginning. It may be that the (Move-based) function ‘Recursion’ may mature over time, in incremental intervals, leading to stages of child language acquisition, and in the manifesting of pidgin language. But when all is said and done, strong continuity claims don’t necessary span across other species or even intermediate phases of our own species. In fact, strong evidence suggest the contrary—that the unique recursive property found specific to our own species, early Homo Sapiens (Cro-Magnon) has in fact no other antecedent that can be retraced past a date of approximately 60kya—JG.*

#### **[1] Introduction**

Before advancing theories about the nature of protolanguage, it would seem that what we now know of the ‘brain-to-language’ corollary would help inform our understanding of critical issues on the topic. Along with an ‘ontogeny-to-phylogeny’ trajectory, perhaps indicative of how the stages of early child language seem to unfold and mirror what we know of language evolution in general, the best heuristic tool we have to solving the puzzle of language emergence, growth, and mastery is two-fold in nature, which approximates answers to the questions: (1) what type of linguistic processing seems to be unique only to our species



(species-specific)?, and (2) what levels or areas of neuro-cortical substrates seem to underwrite these unique processes? The former question is perhaps best articulated in the Hauser, Chomsky and Fitch (HCF) paper which first appeared in the journal *Science* in 2002. The latter has been addressed in multiple sources, the first of which drew my attention was the Fisher, Marcus review which first appeared in 2006 in the journal *Nature Reviews Genetics*, and others including most recently in Larson et al. (2010) *The Evolution of Human Language* (see both Lieberman and Stromswold chapters).

- [2] What seems to be the locus to the question surrounding the nature of protolanguage hinges on our understanding of, first, how we should go about exactly defining ‘Language’—not say language with a small ‘l’, (such as *French, English, Japanese*), but rather language with a capitol “L” (what we mean of *Language in principle* (question 1 above)). What has come out of the second half of the last century, in terms of our Chomskyan framework, is an attempt to demonstrate, *empirically*, the quite distinct notion of what had henceforth been assumed *a priori* by theory-internal devices—namely, that ‘*No syntactic principle or processing applies directly to words, or to superficial word ordering*’ (Piattelli-Palmarini 2010, p. 151)<sup>12</sup>. Rather, what the Chomskyan view grants us is a language borne of a *categorical nature*, abstract and seemingly defiant of communicative functions and unrelated in critical ways towards any strict interphase with the environment.
- [3] It seems theory had to move away from traditional ‘word-based’ constituencies (such as Noun Phrase, Verb Phrase (NP, VP)) and move towards more abstract constituencies dealing with inherent features of the H(ead) of a word, along with a H’s relationship to other Heads. (For instance, the INFlectional Phrase (or IP) came out of this tension between head of words versus heads of features (see fn. 1)). A cursory look at the chronological record of the generative grammar enterprise takes us from early 1950s T-makers of transformational grammar (TG), to recursive phrase structure grammars (PSG), to X-bar theory (of the principles & parameters framework (P&P) of the 1980-90s which delivered a ‘Spec-Head-Comp’ configuration (the holy grail of the ‘Spec-Head’ relation), only to be overturn most

---

<sup>12</sup> Hence, linguistic theory had to move away from a traditional ‘word-based’ Phrase-constituencies (VP, TP, CP) to more abstract ‘feature-constituency’ (Distributed Morphology of an INFlectional Phrase (IP) (Halle, Marantz). Most recently, ‘phrase’ has been replaced with ‘phase’ (an alignment which maps onto the so-called ‘duality of semantics’ (vP, CP)—a move in keeping with what had been articulated by prior notions of scope, c-command, Head-to-Head/Comp movement, dynamic anti-symmetry (Moro), as well as probe-goal relations: all of which have become central, abstract tenets of the theory.

recently within the minimalist program (MP) by the prosaic ‘Head to Head’ relation whereby the simple ‘Merge’ of two Heads is now the driving force behind all syntactic operations. (See ‘Note-2’ for full discussion).

- [4] Much of our discussion related to the ‘four-sentences’ section of this book is deliberate in showing just how rather byzantine constraints on abstract syntactic structure defy what would otherwise be intuitively expected of a simplistic means of functional communication: e.g., why should the clitic formation of [that’s] in ‘sentence no 4’ (found in our ‘four-sentences’ analyses)—a clitic formation that is legitimately pervasive otherwise—not be allowed<sup>13</sup>?

Sentence-4: ‘I wonder what \*that’s/that is up there’

Clearly, either form [*that’s*] vs. [*that*] [*is*] should share in the ‘equal status’ of plainly being able to communicate the simple proposition; however, the clitic [that’s] in this syntactic structure found in sentence no. 4 is ungrammatical. (What we could say is that while [that’s] communicative value is plus [+Com], its syntactic value is minus [-Syn], demonstrating that there is disassociation between (formal) syntax and (functional) communication. (See Piattelli-Palmarini (ibid) for other such data and analyses).

- [5] For example, it is the internal categorial structure of the ‘H(ead)-features of the word’ which is now seen as projecting the outer phrase constituency: e.g., for the lexical item V (verb), it may be the categorial features related to T (tense), a ‘finiteness effect’, which determines its syntactic valence of how it might select for a determiner (Subject, Object, Case). The dual probe-goal phases of CP & *vP*, as currently understood within the Minimalist Program (MP), may similarly assign H-features which map onto the so-called **duality of semantics**: where phase/CP is responsible for scope and discourse-related material as well as the functional projection of AGReement (and presumably Tense), and where the phase/*vP* maps onto argument structure (and presumably Case).

- [6] Recently, it has been proposed (HCF) that it may be the sole, unique properties of recursion which is behind the very underwriting of this ‘categorial nature’, and that more specifically,

---

<sup>13</sup> See ‘Four Sentences’ [Sentence #4] for a full analysis.

these features reside as ‘edge-features’ of a phase<sup>14</sup>. If we assume that Chomsky (in particular) and HCF (more generally) are right, within the linguistics context, namely, that ‘language = recursion’ (as defined in his terms by a **Language Faculty narrow** (LFn), then our question becomes: What types of neuro-substrates serve recursion, which would be separate from other cognitive/motor control function? A second follow-up question, relevant to the question at hand, would be what a proto-language might look like stripped of this narrow language faculty, where a language may only show evidence of linear sequence [A], [B], [C]..., a **recurrent** but not a **recursive** structure, as found in [A [B [C]]]...<sup>15</sup>

### Broca’s area/Wernicke’s area revisited

- [7] Thought it still makes for a nice pedagogical device, we now realize it is quite over-simplistic to talk about a compartmentalized ‘seat of language’ in this way that straddles the classic Broca-Wernicke divide. Although it continues to feel natural in wanting to map etiology of language-specific diseases to specific cortical regions of the brain—e.g., how Parkinson’s disease (PD) presents differently from Alzheimer’s disease (AD), or how with the Autism-spectrum Williams’ syndrome (WS) suffers from unique processing deficits distinct from Asperger’s syndrome (AS), etc.), and, furthermore, how these distinctions might in fact show up in specific areas of the brain (Broca (= PD, WS)) vs. Temporal-lobe (AD, AS) respectively)—what we have rather discovered is that the surface areas which we call Broca’s area (BA) & Wernicke’s area (WA) are merely terminus levels found on the cortex. (The notion of terminus nodes which surface on the outer-cortex was presented as early as 1885 when Lichtheim analyzed interconnected neuro-pathways between BA & WA). If BA & WA are just termini which are *post hoc* defined merely by the gathering place where certain types of neuro-

---

<sup>14</sup> In a [Spec [Head-Comp]] configuration, the so-called ‘edge’ would be the Spec position, away from the core inner working of the phrase/phase. Spec is often defined as an ‘elsewhere category’ which allows for MOVE to take place (whether the Spec is serving as a host for the moved item, or is instigating the move in the first place in accordance to a Probe-Goal relation).

<sup>15</sup> Recall, a linear recurrent model would show a potential two-word utterance as [[drink] [water]] without the necessary syntactic/recursive properties which would allow for a full expression behind the notion of *someone drinking*. In other words, a flat combinatory sequence would only yield two items in isolation [x], [y]. What is lacking is the recursive syntax of: [drink [~~drink~~ water]], which shows MOVE allowing for a hierarchical expression. (See §[22] ‘A Summary of Labeling and how “Merge vs. Move” affects Word Order’. Also see ‘Note 2’ for fuller discussion of ‘dynamic antisymmetry’).

bundles that gather together fire together (under specific tasks, language tasks, etc.), then, the more critical question is: What actually underwrites such specific neuro-bundles?

- [8] In other words, what we must reconsider is the possibility that perhaps it is not the cortex at all that is doing the underwriting of the neuro processing (not BA, WA), but rather the processing is being guided by more robust and underlying subcortical-clusters which precisely bundle and target specific areas of cortical mapping. In other words, if BA and WA do subserve specific types of language tasks (as classically assumed), they do so due to their mapping of subcortical-neural-circuit (SNC) triggering. The best case scenario here for such SNC processing is what we have learned over the past twenty years regarding the functions of the **basal ganglia** (a group of structures found deep within the cerebral hemispheres, which includes the relay-connectivity of the *Putamen* and *Thalamus*, both working in tandem which form a cortical ‘feed-back’ loop). Recent studies have now shown (see Cummings 1993 for review, as cited in Leiberma’s, p. 167 found in Larson et al. (2010) that distinct regions of the frontal cortex indeed connect with their basal ganglia and thalamic counterparts, constituting largely segregated basal ganglia-thalamo-cortical (BTC) neuro-circuits.
- [9] The main ‘SNC-processing’ which roughly maps onto BA is **movement** (MOVE), the unique ability (perhaps motor-control related) to displace an item in the surface-level (phonology) to some other place in the underlying (syntactic) structure, *inter alia*. The basal ganglia, with its ‘looping effect’, bringing subcortical neuro-circuitry to percolate up to the surface cortex, seems to be the best-case cerebral candidate to serve the unique phenomena of MOVE, where recursion is required to break with a flat sister-relation otherwise found of surface phenomena (See § ‘Note 2’ below regarding **recursion & dynamic antisymmetry** (DA)—processes which extend otherwise flat sister-relations to having hierarchical status). Recursion has the property which allows cortical mapping of two language-specific tasks (both seemingly BA-related): viz., that of *phonology*, and that of *inflectional morphology*. While phonology as recursive is still hotly debated<sup>16</sup>, (and Chomsky doesn’t appear to be

---

<sup>16</sup> Syllable structure of <onset, <nucleus, code>> might be recursive due to its inherent hierarchical structure. For review, see Schreuder, Gilbers, and Quene’s paper ‘Recursion in Phonology’ *Lingua* 119 (2009). It also bears keeping in mind that MOVE-related diseases such as PD do seem to impact both phonology and syntax, while other studies also suggest that MOVE correlates to mouth movement, planning and articulation of speech as well as syntax. Broca’s aphasia may impact both speech as well as syntax. What we could then say of PD is

swayed by such arguments), to the contrary, inflectional morphology which is defined by movement (displacement) is clearly quintessential recursive in nature.

[10] In this note, I focus only on the displacement properties of recursive syntax found in inflectional (INFL) morphology (as present in morphological **Case** and **Agreement**, both which are INFL-related, and see what proto-language absent such INFL/Recursion might look like (comparing data results to that of pidgin language and/or even Chimp ASL e.g., Nim Chimski, (Terrace 1979)).

### Proto-language and Derek Bickerton

[11] I know of no more passionate advocator for a protolanguage than Derek Bickerton. His and his colleagues' tireless work examining Hawaiian pidgin—as a heuristic model for what linguists should look for towards a proto-language grammar—has brought the once taboo topic to the fore of current linguistic theory and debate. Today, the theoretical notions leading to any understanding of a putative proto-language have suddenly found its underwriter by the larger, and perhaps even more ambitious, interdisciplinary field of *Biolinguistics*. This brief 'Note-1' is in response to some thoughts on what has been laid out in Derek Bickerton's 2014 paper 'Some 'Problems for Biolinguistics' (*Biolinguistics* 8).

Having set-up some discussion regarding the current state of the 'biolinguistics enterprise', and some non-trivial problems pertaining to its research framework, particular to the *Minimalist Program* (MP) (Chomsky 1995), Bickerton goes on to express his long-held views on the nature of a Protolanguage (§4.2)—namely, *pace* the given Chomskyan account, that there should be NO inherent contradiction between the coexistence of the two statements (below):

---

that it affects the basal ganglia along with its SNC-processing leading to the inability to exact MOVE-based recursion, as found both in phonology and syntax.

### Statements:

[12] (i) 'Statement-1': That language is to be properly defined, very narrowly, within the terms of a **Language Faculty-narrow**, an **LFn** which, by definition, *excludes* most of what is typically accepted within the linguistics community (outside MP) as defining what normally constitutes a language (e.g., vocabulary, idiomatic & encyclopedic knowledge (= the lexicon), phonology (syllabic constructions), and some particular aspect of morphology (e.g., derivational processes, compounding, etc.). A layman's classical definition of what constitutes 'language' is intuitively very broad in nature. But Chomsky's definition of a language faculty (LF), to the dismay of many, is exceedingly Narrow (n): That LFn is the sole property of recursion: that language is exhaustively defined by the exclusive and very narrow property of recursion.

(ii) 'Statement-2': That a putative **protolanguage** theoretically exists and could **serve as an intermediate step between a partial language and a full-blown LFn**—viz. an intermediate language phase which would find itself tucked-in between what we know of pidgin languages (an L2 attempt to formulate a rough grammar for functional communicative purposes), and perhaps chimp sign-language and other animal cognitive-scope features (of the type taught to the chimp named Nim Chimpski (Terrace 1979)), along with other communication systems which are not on equal par with LFn, of what Chomsky refers to as **Language faculty-broad (LFb)**—viz., 'broad' factors which *include* the aforementioned lexical-item development sensitive to frequency learning, formulaic expression, and other similar 'frequency-sensitive' morphological word-building processes such as compounding and derivational morphology.

[13] In other words, Bickerton's claim here is that we can accept both statements as true—they are not mutually exclusive:

(i) (LFn) Yes! 'Language-proper' is to be narrowly-defined as pertaining to the sole (and, as it turns out, quite a unique) property of 'recursion', and,

(ii) (LFb) Yes! There too could be a protolanguage (by definition, an LFb) without ‘recursive operations’—a language just shy of maintaining the status of a “full-blown language’ along the language spectrum<sup>17</sup>.

The two claims appear to reflect on larger dichotomy issues. Let’s flesh this out below in the way of the dichotomy debates: ‘form vs. function’, ‘continuity vs. discontinuity’, ‘nature vs. nurture...’

### **The dichotomy debates.**

[14] Taking the former ‘recursive property’ (=syntax/LFn) as a critical aspect of a dichotomy-debate (say, of continuity), one would most certainly claim the emergence and development of recursion (MOVE) to be *discontinuous* in nature from all other non-human primate communicative systems, perhaps accepting Gould’s version of recursion as ‘exaption’ at one end of the spectrum with Chomsky’s single-mutation-event leading to a ‘pop hypothesis’ on the other<sup>18</sup>. In any case, both claims would be consistent with what Gould calls a ‘punctuated equilibrium’ hypothesis—i.e., that recursive language (LFn) emerged in one fell swoop, either exaption from prior material (his ‘spandrels’) or a completely novel structure<sup>19</sup>. The features of the latter (LFb), ‘from prior material’, most certainly would maintain at least some level of *continuity* assumptions, as widely expressed in the language-evolution literature (e.g., Pinker & Bloom (P&B), among others).

P&B may be correct in assuming that there is ‘somewhat’ continuity regarding the articulation mechanism of sound/phonology (i.e., the chimp’s ability for syllabic pant-hoots, and other primate syllable-vocalization capacities—though it must be said that human speech is indeed quite unique and highly specialized due to the lowering of the larynx), as well continuity in what we would find regarding the idiomatic ‘one-to-one’ associative-learning mechanisms behind the mapping of ‘sound/gesture to meaning’ (manuofacial expression), ‘cue-based’ representation (in the ‘here & now’), and other non-formal constructions leading

---

<sup>17</sup> Bickerton has long sought to advance an intermediate stage of language, ‘a proto-language’, as a grammar just shy of maintaining a fully-fledged recursive grammar. What so-called ‘flat- recurrent’ (non-recursive) grammars would not be able to do is creatively generate and parse constructions beyond a preconceived semantic/canonical specificity. See endnote of this section for discussion of recurrent versus recursive grammars.

<sup>18</sup> See Jean Aitchison (1998) for a review of the ‘slow-haul’ vs. ‘pop hypothesis’ in this context.

<sup>19</sup> See Crow (2002) for a sudden ‘genetic mutation’ hypothesis (which would be akin to Gould’s ‘punctuated equilibrium’).

to compounding and even limited syntax (lexical-root phrases such as [NP [N] + [N]] constructs which approximate possessive structures e.g., [NP *daddy book*] (= daddy's book) or prosaic [VP [V] [N]] constructs which approximate Tense/Agreement e.g., [VP *daddy drink water*] (=daddy drinks water), etc.

But I suppose, for Chomsky, the question is: Can we really get there from here? Can FLb turn into FLn?: Really, can broad-communicative features (attributed to non-humans) as laid out in (HCF) evolve into (human) FLn? (For Chomsky, the answer is an unequivocal NO! and hence another dichotomy debate). Chomsky's now famous analogy lends us to imagine a sudden mutation (or catastrophic event) devoid of any 'bottom-up' Darwinian selective pressure for FLn:

*'We know very little about what happens when 10<sup>10</sup> neurons are crammed into something the size of a basketball...' (Chomsky 1975: 59).*

These open lines of a much longer paragraph on the topic fully commit to a top-down 'form-precedes-function' analysis regarding FLn. Bickerton has carried on with the same theme arguing against any 'non-human to human-language continuity' when he claims that:

*'[T]rue language, via the emergence of syntax, was a "catastrophic event", occurring within the first few generations of Homo sapiens sapiens'. (Bickerton 1995: 69).*

### **Child-to-adult-Continuity.**

[15] Let's remind ourselves that Chomsky believes in 'child-to-adult' continuity (if not in 'function', in 'form') given that language, per se, has potential drop-off way-stations on its way to a full target-grammar projection. So, for early child language utterances, the nature of their errors (functions) is rather epiphenomenal since the underlying grammars (forms) which *underwrites* the syntactic templates must be (if we assume a UG) the same 'all the way up/down' between child and adult. Though, perhaps a better way to view Chomsky's remarks is to suppose that we can still tease apart '*form* from *function*' (yet another dichotomy)<sup>20</sup>. For

---

<sup>20</sup> 'Function-to-form': so, think baseball glove: catcher's-glove is padded due to repeated fast balls ('catch the ball softly!'), outfielder's glove is light due to its having to be held while running for the fly-ball ('catch the ball running!'), first-baseman's glove is extended due to a race between the ball and the bat's-man running to tag first base ('catch the ball quickly!'). This here is 'function defines form' (or function precedes form). But language seems to be the reverse (form precedes function), where the form of the (internal) mental template



instance, assume that Chomsky agrees with the assertion that children first exclusively *function* with Merge (and not Move)—while still maintaining that the *form* of UG is the same as consistent with continuity. Well then, there could be space within such an argument for an emerging grammar. The hypothesis would be that young children (at the low Mean-Length of utterances (MLU) stages) would be *forming* the same UG as their adult counterparts while their *functions* would be immature, following a protracted maturational scheduling of function: UG...(stage-0).....stage-1 (Merge)....stage-2 (Move)...on their way to a full Target language (stage-T). What could be claimed then is that it's the function 'MOVE' which matures and eventually comes on line.

True, the capacity for MOVE was always there (UG), it's just that the hidden processes which map 'form to function' followed a protracted schedule. This is not unlike what we would find for the maturational development, say, of **functional categories**—viz., while their form is intact, as part of UG (DP, TP, CP), their mappings of 'form to function' are delayed (see Galasso 2003). Again, this form-to-function disparity could be one way to reconcile Chomsky's strong stance calling for a non-developmental UG (since the empirical observation is valid: that language, at any given stage of development, never exhibit UG violations, nor do they ever exhibit 'wild grammars').

- [16] Lastly, the above notion that MOVE is maturational-driven (within our Homo species) seems to nicely correlate with what Chomsky himself claims of language evolution (within our Homo species): that 'Every inquiry into the evolution of language must be an inquiry into the evolution of the computational brain machinery capable of carrying out edge-features operations' (Chomsky MIT lecture, July 2005, cited by Piattelli-Palmarini (2010, p. 151)). Recall that what we mean by 'edge-feature' operations are those syntactic operations which can only be handled by the unique recursive property of MOVE. Also recall that there is also a high level of inherent abstract symbolism involved in any MOVE-related/edge-feature operation since such principles of MOVE (i.e., syntax) do not map onto words *per se*, in an iconic 1-to-1 manner (as might be intuitively imaged of language), nor is there any surface word-order mapping (which might be expected of surface phonology). Rather, MOVE inherently requires the mental manipulation of categories (=symbols)—these are categorical

---

seems to shape the potential (external) function of language. For example, an Arabic speaker sticks out his phonological-perception glove to catch an (external) English '/p/-ball' (say, the word /P/olice) (using a phonetic-pitch metaphor to baseball), but (internally) catches it as a '/b/-ball', where /polis/ (police) gets caught as /bolis/. (Arabic has no /p/ phoneme in its phonological inventory, and /p/ vs. /b/ does not make up minimal pairs). See also discussions surrounding the dichotomy between 'functionalism vs. formalism'.

concepts such as Verb, Noun, or constituency structure which breaks with surface word order. (See our 'Four-sentences' analysis for full discussion of recursive constituency).

The implications here is that in order to question the nature of language evolution, and all of its complexity, the first order of business is to address the question of determining when the first evidence of MOVE appears in the early Homo species, and if, as a result of MOVE, other spin-off exaptations (or so-called 'hitch-hiking' free-rider adaptations) can be explained as being *bundled with recursion* (perhaps even neurologically bundled): I am thinking about theory of mind, shared attention, symbolism and displacement which contribute to so-called 'detached representations', altruism, dance, ceremonial practices & taboos, and other perhaps niche motor-control abilities such as tool-making capacities (which demonstrated a mental template for the design of the tool), throwing capacity, so-called 'remote threat', sheltering, cooking of food, etc.

[17] What all of the above features have in common—as a unifying thread which can lead to recursion/MOVE—is the ability to project an item, project oneself, away from an icon, an index, and to become symbolic and categorical, both in nature, in index, and in design. What we currently know—out of all archaic homo species (Homo-Habilis (*Australopithecus africanus*), (early Africa)-Ergaster, (late Asia)-Erectus, Heidelberg, Neanderthal)—is that only Cro-magnon<sup>21</sup> (our early homo sapiens-sapiens ancestors, say at around 40-60KYA) had emerged onto the scene (seemingly top-down) into being *categorical* in nature, gaining a rich symbolic system first drawn from an inner mental language (MOVE), with subsequent bootstrapping to be applied to other non-linguistic, cognitive, motor-control tasks. Once a full-blown symbolic inner-language system emerged (either via a catastrophic mutation or via exaptation), what came with it was all the 'bells and whistles' of being a member of a unique symbolic club, what today we call the 'homo-sapiens-sapiens club'). It is now well recognized that by the time Cro-magnon comes on the scene, having evolved in whatever which way, they came on the scene drenched in symbolism (White 1989). If they evolved at all (bottom-up), what we can say is that they evolved from an earlier time/species of not yet having

---

<sup>21</sup> The brain-size trajectory most certainly would be a major contributing factor with regards to any such evolutionary-based theory for either a 'gradual development' (bottom-up) or 'sudden emergence' (top-down) of FLn. To be considered is a respective brain-size spectrum that would begin at around 450cc with Australopithecus, Erectus at 1000cc, to roughly 1500cc with Neanderthal, followed by a very slight decline in Cro-magnum stabilizing at 1300cc).

recursion, to a later time/species<sup>22</sup> when they have it, or if you prefer, using our current linguistic terminology, from an earlier time of having FLb, to having FLn.

- [18] All of these FLb linguistic factors—which, as suggested above, may have hitch-hiked from categorial symbolism in its purest form (viz., *sensori-motor control, mapping of sound-to-meaning, lexical retrieval, word-building, compounding, and even derivational morphology*)—are all found in the very early stages of child language acquisition (Radford & Galasso 1998, Galasso 2003, 2016), and have antecedents which can be traced back to pidgin grammars (Bickerton), and, to a large degree, even further back to what we have gleaned from non-human primate communication systems (the use of ASL by Nim cited in Terrace). Of course, it goes without saying that notions of any putative “‘somewhat’-continuity’ between human and non-human as it regards cognitive scope, theory of mind, altruistic features, etc., must be taken with a grain of salt—viz., there really is not much continuity to speak of, in these realms, and the very fact that non-human primates lack what would in humans be such simple operations surely present us with the ‘smoking gun’ of discontinuity through and through. (The underlying question to ponder here is whether there might be a ‘singular, unifying mode of processing which underwrites these realms—and, neurologically, might it be related to MOVE?).

In any case, for what it’s worth, this dual acceptance allows for both *continuity* (LFb) and *discontinuity* (FLn) to flow from out of ontogeny and phylogeny trajectories—ontogeny in terms of ‘critical period’ cases in which a proto-language (Bickerton’s ‘bioprogram-hypothesis’) may fare no better than the ‘end result’ of a trajectory of an individual’s growth and plateau of syntax (leading to a pidgin language—in that a pidgin is in many ways discontinuous from a target L1, just as early child language shows discontinuous properties due to their lack of full recursion of syntax).

- [19] In terms of phylogeny, we could assess claims which speak to how ‘language-broad’ evolution might be continuous in nature, with antecedents which harken back to animal cognitive capacities. In other words, Bickerton claims we can find an intermediate phase along these dichotomy-spectrums, in one sense leading to a human-language (immature) capacity which would solely incorporate LFb-features—including *inter alia* a limited lexicon with perhaps a maximum ‘mean length of utterance’ (MLU) count of below 3 (i.e., no more than three words per utterance), along with the complete absence of Inflectional morphology; what we would

---

<sup>22</sup> One possibility implied here is that (early-FLb) *Homo erectus* evolved into (late-FLn) *Cro-magnum*.

expect of pidgin-language capacity. Though it is the latter statement that Chomsky rejects, I, along with Bickerton, see no reason at all, at least conceptually, why there couldn't be a syntactically, albeit robust, LFb-phase of child language (ontogeny) on its way to a fully-fledged LFn, and if so, why this intermediate phase that the child passes through couldn't constitute what we would at least theoretically claim of a protolanguage (phylogeny). In one sense, this kind of argument mimics the old adage 'ontogeny recapitulates phylogeny' (first cited by Ernst Haeckel)<sup>23</sup>.

[20] Chomsky's insistent belief is that there could be, conceptually, no intermediate step shy of a full language; if you have such a step, then it's merely a function of a communicative niche (as expressed above), and that such a deprived system (deprived of recursion) would by its fixed nature need to remain there, as a non-evolving, non-human communicative system (=LFb). This is tantamount to saying that **LFn cannot arise from LFb**—viz., that there can be no continuity between LFb and LFn (not in a phylogenetic way 'evolution', nor in an ontogenetic way 'child maturation' as cited above). I believe (and I assume Bickerton would agree with me here) that Chomsky's assertion is too strong. Chomsky has been quite consistent ever since our reading of the 'Fitch, Hauser, and Chomsky paper' (2005)—on the topic of the nature of LFn and of language evolution—that a definition of 'Language' (a language with a capitol "L") can only be purely based on one essential property, namely the property of **recursion**.

For Chomsky et al. (2005), [language = recursion]. This very narrow definition is perhaps the only way that Chomsky can maintain his long-held notion that language is biologically modular and human species-specific (modular in that it functions like any other organ, e.g., the liver, stomach, lungs) and species-specific in that its operation is uniquely situated in the human brain/mind (presumably Broca's area, a region which seemingly only serves recursive operations such as (inter alia) the planning of articulation leading to mouth movement, and the movement involved with syntax).

---

<sup>23</sup> For example, see Dan Slobin (2004).

## The Minimalist Program (MP) Enterprise: Resolving a dichotomy

[21] But there does seem to be way to reconcile both statements within the MP enterprise. Within MP, there are two types of **movement** (mapping with what one finds regarding the ‘duality of semantics’):

**(i) Local-move** (= Merge, the merging of two Heads) is based on the merging of two items (two Heads (H) within a Phrase (P))—e.g., such as what we would primitively find in H-H compounding (e.g., Adj+N sequences such as [black]-[bird]=> [blackbird]), a simple base in-situ Verb Phrase such as [VP [V bounce] [N ball]], and non-formal sentence constructs such as ‘Me go’ or ‘Him do it’ which show a complete lack of inflectional morphology (a lack of Case and Agreement). All these constructs do show up in impoverished pidgin systems as well as in very early MLU stages of child language, and can be attributed to the kinds of features we find in non-human communicative systems of the sort famously demonstrated by the chimp named Nim (ibid).

But note here that in order to know where the H of the P is, one must involve a second (later) merge operation coming on the heels of the first. In order to reach the VP derivation of the unordered set {V, N}, locate the Head {V} and **label** the P accordingly, the speaker must utilize what is referred to as Internal Merge (IM) (an instance of distant-**Merge/MOVE**), so that the unordered set {bounce, ball}, becomes an ordered pair: syntactically deriving the mere twin lexical items [bounce, ball], to a fully-fledged VP [bounce [~~bounce~~, ball]]. So, we would speculate that at any impoverished mere ‘local-move stage’, we should find instances of mixed word order and lack of inflectional morphologies leading to the absence of Case and Agreement. This is indeed what we find not only of child language (See Radford & Galasso 1998), but also what we find regarding pidgin formations, and finally what the extremely curtailed limits of Nim’s speech range.

**(ii) Distant-merge** (= MOVE) is based on the subsequent move (a second-order move) which, as a result, breaks the flat symmetry of an unordered set and allows the labeling of a Head of the phrase to be defined. In contrast to local merge, distant merge (= move) allows for a portmanteau of features and phenomena, among them the syntactic operation of movement which break base in-situ constructs and allows for the lexical item to percolate up the syntactic tree in order to check-off +Formal features (in current MP terms, as guided by the ‘probe-goal’ relation). Other consequences of MOVE would be the projection of

Case (+/-Nominative), AGReement (Person, Number) as well as Tense, all of which are found in higher phrasal projections above the base-generated VP. Again, any lack of these higher, formal projections would have syntactic consequences. (A brief summary follows. But see Note 2 for full discussion of Merge vs. Move and ‘problems of projection’).

### **A Summary of Labeling and how ‘Merge vs. Move’ affects Word Order.**

- [22] First-order /local merge—the simple assemblies of two lexical items in creating an unordered set, say a Phrase (P) {a, b} out of the two items. Yet, there is no recursion; hence, there can be no labeling of what would constitute the Head (H) of the P. In order to derive H of P, a second-order /distant merge must break with the set in creating an ordered pair  $\{\alpha, \{\alpha, \beta\}\} = P$  (where  $\alpha = H$ ). It is via this second-order merge (which constitutes as a recursive property) that we can derive order within the P—an order which comes about as a result of the ability to label which of the two items is rendered as H.
- [23] Consider, at least theoretically if not empirically<sup>24</sup>, a young child’s inability (at the early mean length of utterance stage (early-MLU) to derive second-order merge labeling, thus being incapable of understanding labeling of H, rendering such otherwise adult unambiguous structures ambiguous: e.g, [*house-boat*] is read and interpreted as a kind of boat (and not as a kind of house). But, if we first examine the base-structure of the two lexical items {house, boat}, there is no way we can glean from a flat, unordered structure what the Head word of the compound [N+N] would be. This problem is in fact what we find in very early instances of

---

<sup>24</sup> See the monograph *From Merge to Move* (Galasso, 2016).

child language<sup>25</sup>. Carol Chomsky<sup>26</sup> first found the lack of recursive operations regarding passive formations—that when young children were faced with (improbable) irreversible passives (e.g., *The ball was kicked by the boy*/\**The boy was kicked by the ball*) they scored quite well. But when children were presented with reversible passives—passive interpretations which must exclusively rely on ‘syntax’, as opposed to irreversible passives which were actually acquired quite early in development since ‘semantics’ can serve to help with the only probable interpretation—the children tested were unable to correctly demonstrate that type of movement necessary for a passive interpretation. In other words, children had a hard time with (e.g., *The man was killed by the lion*/*The lion was killed by the man*) where both readings are probable and reversible.

[24] It is interesting to note here that Grodzinsky (1986, 1990, 1995) similarly finds in Broca’s aphasia subjects an inability to handle ‘distance of movement’ in embedded subject-relative clauses, where (i) **local movement** had an ‘Above chance’ level of acceptance/reading and where (ii) **distance movement** had only a ‘Chance level’—e.g.,

a. ‘The cat<sub>t</sub> [that [ \_\_<sub>t</sub> chased the dog]] was very big.

(local move = Above chance)

c. cat<sub>t</sub> [that [the dog chased \_\_<sub>t</sub>]] was very big.

(distant move = Chance).

---

<sup>25</sup> And as discussed herein, such an inability for labeling would force a flat reading of the two items [drink] + [water] as two separate intonities without the luxury of syntax—viz., an individual with a recursive grammar can reconstruct the two items syntactically, within a VP, such that a proposition can be generated: that ‘someone is drinking/wants to drink/should drink water’, etc. The same two items, recursively, get structured as [VP drink [~~drink~~ water]] where the Verb ‘drink’ now dominates the Noun ‘water’ in a *mother-daughter* hierarchical relation [x [x,y]]. As long as the two items stand in a flat non-recursive, recurrent manner [x, y], all one could glean from the utterances is that ‘drink’ and ‘water’ have been combined, ‘stacked’ in sequence, and where perhaps word order has no bearing on structure. The fact that a person (with full adult syntax) can reconstruct a meaning out of a simple two-word utterance such as ‘drink water!’ suggests that such a bootstrapping relies on a matured mental syntax in supporting the reading. (See Note 2 herein for Dynamic antisymmetry and Problems of projection).

<sup>26</sup> Chomsky, Carol. 1969. *The acquisition of syntax in children from 5 to 10*. Cambridge, MA: MIT Press.

[25] In other words, the greater difficulty in comprehending sentences in situations where syntactic form is not supported by semantic content suggests that the semantic component of grammar may play an important role in the young child's acquisition of syntactic comprehension—the latter ‘semantic-content’ interpretation being a product of local merge, viz., the yielding of the lexical items and how each item plays a thematic role in the sentence. In the case of ‘distance-moved’ (cf. Grodzinsky), it seems that adjacency of linear order (surface phonology) takes prominence over hidden structure at a distance. (Also see Galasso 2016, chapter 8 for treatment of Broca’s aphasia data).

[26] Distant Merge (= Move) has something to say about how we glean a Head for a given phrase. In the case of ‘house-boat’ (‘a kind of boat’, not ‘a kind of house’), in order to derive the head of the Compound (C) (heads are *right-branching* in English Compounds) we must employ second-order distant-merge. Following Moro’s work on dynamic antisymmetry, accordingly, in order to label a H of a P (or C), we first must break with flat/sisterhood relations—of the kind typically associated with ‘logical and’ (e.g., I need to buy: ‘a and b and c and d’ whereby comma-insertion allows for displacement and rearrangement of ‘a-d’ in any order since sister-relations are symmetrical and hold no hierarchical order—and create an antisymmetric hierarchy, such that from out of a sister, first-order/local-merge set  $\{\alpha, \beta\}$  <house//boat, boat//house> (showing symmetry), we derive {house {house, boat}}.

In this second-order ‘Move-based’ structure, notice how the Noun ‘house’ has risen up to a higher functional node within C. It is this movement that breaks flat sister relations and creates, as Moro puts it, dynamic antisymmetry in labeling H of C. For phrases, that is at work. Take for example the VP [VP [{V bounce}, {N ball}]]. In order to derive the H verb ‘bounce’ of the P ‘bounce ball’, (English H of P are left-branching, just the reverse we found with C), there needs to be second-order distant move, such that the H becomes labeled as distinct from its complement: first-order local merge: {bounce, ball} (showing no order) becomes second-order distant merge/move {bounce {bounce, ball}} => [VP {bounce {bounce, ball}}]. It is clear that there are instances in the child language literature where young children cannot yet discern their proper word-order, e.g., a child may utter VP *bounce ball*, or *ball bounce* with identical intentionality (see Galasso, 2001, <https://www.csun.edu/~galasso/worder.pdf>).



***Distant-Merge related structures missing in pidgin and which constitute the basis for Proto-language.***

**How ‘Merge vs. Move’ affects Case Assignment.**

[27] Let’s begin this section on Case with some basic assumptions, some of which are theory-internal:

(1) Case marking is a ‘functional-category enterprise’—viz., a formal projection which requires movement of the case-marked item to raise out of the base-generated VP and insert into a higher functional phrase.

(2) That there are three distinct (and overt) Case markings in Standard English (SE): Nominative on subjects [+Nom] (e.g., *I, he/she, we*), Accusative on objects [-Nom] (e.g., *me, him/her, us*), and Possessive/Genitive when used as prenominal [Poss+N] [+Gen] (e.g., *my, his/her, our*), and when used as pronominal [N-{Poss}] (e.g., *mine, his, hers, ours*). Also [+Gen] is morphemic {‘s}, {of} in examples [Tom’s [house]], and ‘The house [of [Tom] ‘s] ~~house~~’ (*The house of Tom*). The morpheme {to} also serves to case mark [-Nom/Obj] (e.g., *give it to him / \*give it him/ give him it/ \*give him to it*).

(3) The default, base-generated order of Double-arguments is [Indirect Object + Direct Object] [IO, DO]. (For theoretical discussion, see Boeckx (2008).

(4) That Case can’t be doubly marked from a single verb—in this sense, case is of a ‘Probe-Goal’ (PG) relation instigating an upward projection of a targeted item. Once a probe has located its goal and relevant features have been checked, the probe is no longer active in the syntactic derivation.

(5) There are at least two mechanisms for Case marking:

a. via **Structural/Configurational** with a lexical item (in local domains)—so-called **lexical Case**:

i. Verb-complement, PRN => Object [-Nom]

ii. Preposition-complement, PRN => Object [-Nom]

b. via **Morphemic assignment** (probe-goal) with clitics {-‘s}, {-to}, and {-m}

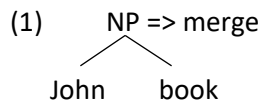
c. Otherwise, via **default** (or **inherent case**).

## A Theory

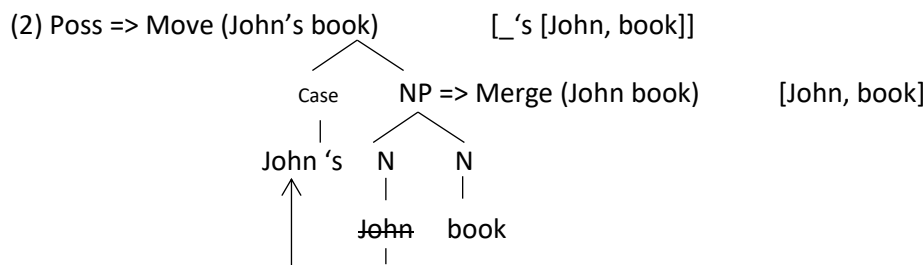
[28] One very persistent characteristic of any putative protolanguage would be its lack of MOVE (movement operations which are motivated by (*inter alia*) **functional features** which make-up a **Probe-Goal** (PG) relation: e.g., Case, AGReement & Tense (and Word Order is most often a result of some movement operation whereby the surface-structure phonology is derived from an underlying hidden structure). Let's just briefly examine how MOVE might correlate to the functional feature of CASE (nominative, accusative, genitive) assignment.

Theory Internal consideration: All functional features/projections must involve movement from out of the base-generated VP/NP, (the VP/NP being a first result and product of simple merge).

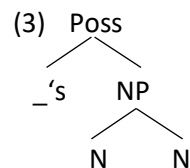
[29] Let's begin with two lexical items (they can be both Heads (H) for the time being). Consider the merging of [[John] [book]]. In this simple [N]+[N] merge operation (absent of any MOVE), the two Hs are considered flat-sequenced, base-generated and thus cannot generate any formal functional features such as Case.



In this instance of merge, genitive/possessive case can't be assigned.



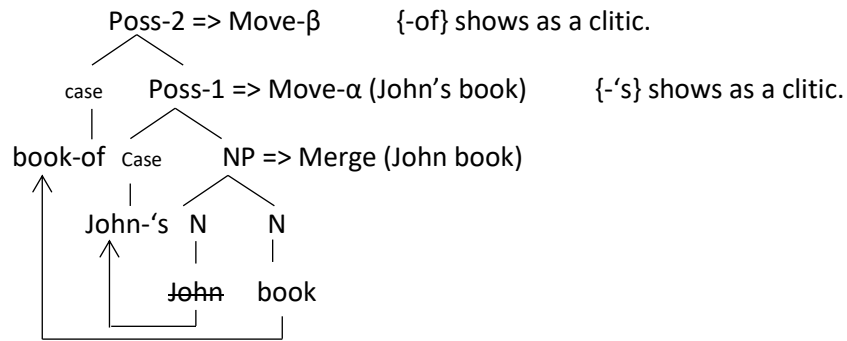
It is rather 'Move', a recursive structure, which generates Genitive/possessives:



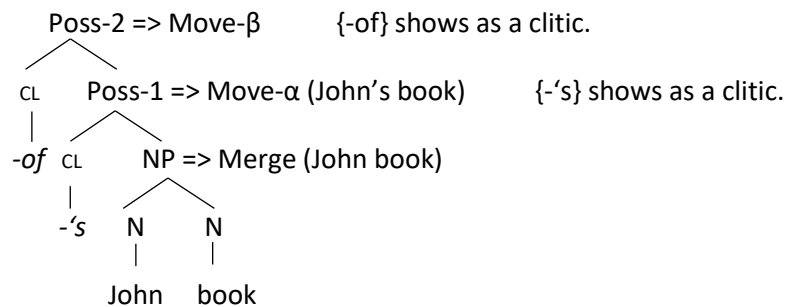
- [30] What we find here, theoretically, is that MOVE is responsible for triggering possessive (Genitive) case marking. One further speculation (see analysis below) is that such functional case marking: {-s}, {-to}, and {-m} function as *clitics* (bound morphemes) which directly insert, perhaps directly pulled from the lexicon, (sometimes merely as a feature, but often as a lexical item itself, as in the case of 'to') which in turn motivate the raising of a lexical host (as specified by the Head information to search-out a PG relation up from a lower position within VP/NP).
- [31] What we know of pidgin syntax, word order is often variable (e.g., Bickerton 1990, also see Galasso 2003, 2016, 2018 for accounts and analyses of early-child mixed word order). Theory internal considerations speculate that at the exclusive merge-level—what we would find of proto-language—no word order can be fixed since both Heads (H) {x, y} or H and Phrase (P) {x, yp} would serve as flat sister relations with no hierarchical dominance. In other words, in brief, fixed word order must be the result of MOVE {x {x, y}}, a recursive property, a property only seen in full-blown human language.<sup>27</sup>
- [32] Given that pidgin, as well as very early child utterances, lack a fixed word order (at the early multi-word stage) (e.g, *me car*, *car me*, (= 'my car'), *mommy sock*, *sock mommy* (= 'mommy's sock'), etc.), this seems to suggest that pidgin, early child language would fall somewhere on the spectrum close to a protolanguage, if what we mean by a proto-language system is that which is devoid of any formal movement operations, and is a system which only employs merge.
- [33] What's also very interesting about the analysis above (and explicitly advanced in Bickerton's 1990 syntax) is that this would explain double-possessive markings found in examples such as [The book of John's], where possessive Case for 'book' seems to be marked twice. Let's see how this might work:
- (a) [John 's [~~John~~ book]]
  - (b) [of [John 's [~~John~~ book]]]
  - (c) [book [of [John 's [~~John~~ book]]]]

---

<sup>27</sup> See web-link no. 28. On Merge vs. Move in child language



[34] So, the **clitic-climbing** is expressed as shown in below, with lexical items raising up to attach to the CLitic (CL) as in a PG relation (in this sense, clitics and/or features of clitics force raising).

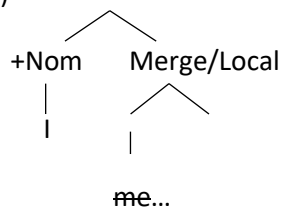


Let's consider further examples of how MOVE & CLitic PG-relation triggers and projects POSSessive case.

In sum, our analysis of how MOVER triggers Pronoun (PRN) Case assignment shows as follows:

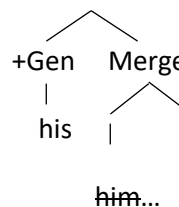
[35] **Case**

(a)



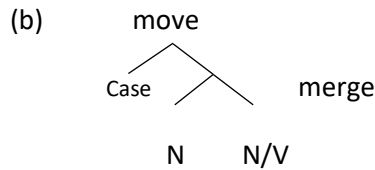
(= I like John) where PRN I is nominative case.

(b)



(=His book) where Genitive Prn his is POSS case.

So, Case marked syntactic tree looks like this:

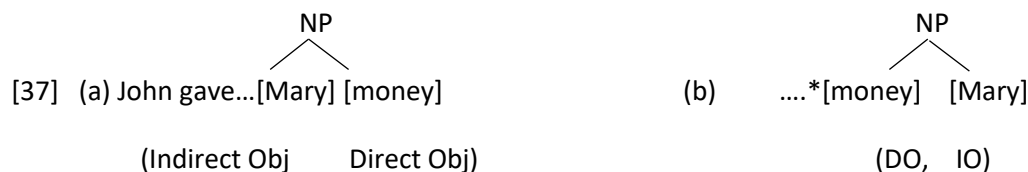


[36] Move-based Case: (Where accusative is default case):

- a. From accusative to nominative via Move: [I [~~me~~ do it]], [He [~~him~~ do it]],
- b. From accusative to genitive via Move: [my [~~me~~-dolly]], [his [~~him~~ car]].

### **Case in Double Object Constructs**

In double-object constructions, when PRNs are employed—which must present overt case marking—e.g., *I, me, my / he, him, his / they, them, their* (Nominative-subject, Accusative-object, Genitive-possessive)—we see how MOVE can trigger Case assignment. Consider the distinction between the two sentences:




[38] a. 'Give [him money/it]!', whereas b. \*'Give [money/it him]!' is \*unacceptable due to there being no 'PG-relation' to enable the Case-marking to be checked-off on the pronoun 'him' (noting that pronouns in English need the checking-off of the overt Case-marking feature, unlike Nouns which require no Case). It can be argued that only in (a) does the PRN *Him* remain in a PG configuration whereby it can receive and check-off the accusative [-Nom]feature structurally via the verb 'give' (cf. [27, (5a,i)]). But nothing hinges on that treatment: otherwise Case is acquired via default. Also note that there seems to be a preference for the structure in 'Give him money' over 'Give him it'\*, suggesting that the PRN 'it' is more sensitive to the right configurational position leading to Case-marking than is its Noun counterpart 'money' (again, since Nouns in SE don't require case-marking). (Below we note that in order to save the derivation found in b., 'him' must raise to be structurally adjacent to the verb, a position that would be forced by a PG-relation in any event).

### ***'Him raising'***

[39] In addition to a potential default setting (where configuration of word order is no matter), another work-around may be to assume that in order to save the ungrammatical derivation in \*(b), movement of the Noun 'money' found in (a) optionally can be employed, allowing for the case-marking morpheme {to} to attach to an appropriate stem—now, the case-marking clitic {to}, attached to the N, serves as the probe of a PG-relation, attracting the goal PRN 'him' to raise in a local/adjacent domain in order to receive Case.

'Give [money-(to) [him money]]!'    Probe-Goal relation.



[40] So, restating what was said in (5), we have three ways in which the PRN 'Him' gets case marked:

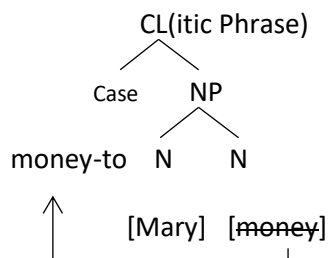
- (1) Structural via the verb {give}:    a. John gave him money (to) ~~him~~. = 'Him' raising
- (2) Morphemic via the clitic {to}:    b. John gave money-to him ~~change~~. = 'money' raising
- (3) Lexical via [Verb + Preposition {to}]:
  - c. John gave (to) him money. = (similar to (a))
  - d. \*John gave money him. = \*(unacceptable)

**Lexical Case** must derive from a [Verb+Prep] configuration: *John [V gave [PP to him]] money*, as separate from [Noun + Clitic], as found in the structure:    *John gave [N/CL money-to [him]]*.

(4) whereby the ungrammatical sentence in d. \**John gave money him* is due to there being no case marking PG-relation for the PRN 'him', it being stranded in a non-configurational manner without local domain to check-off its Case feature (i.e., neither morphemic, structural, nor lexical configuration is available, and a putative ACC default can't be employed within a structural configuration—viz., an accusative 'default status' can only emerge as a result of a non-configurational environment (i.e., 'no structure').

This is exactly what we would expect if Case were a functional projection triggered by MOVE (and an operation lacking at the early stages of child language, pidgin, and certainly not found in non-human primate communication). So, with raising (MOVE) we derive DO-IO and render the sentence:

[41] a. John gave money-to Mary.



With the underlying structure showing PG relation between {to} & Mary, and raising of money to serve as a host to the clitic {to}:

[42] John gave money-to [Mary money].

(Recall, as noted in our **lexical case-marking** treatment above, another work-around involving Case via prepositional {to} is to analyze the Verb Phrase (VP) [give+N] as the right kind of projection that can allow prepositions as its complement, thus allowing case marking to be applied lexically (**lexical case** given that Heads of PP can only assign Accusative [-Nom] Case in SE).

(Again, where {to} now serves as a case-marking clitic, this brings to the number of **morphemic case-marking** clitics to three: *to*, *of*, *'s*).

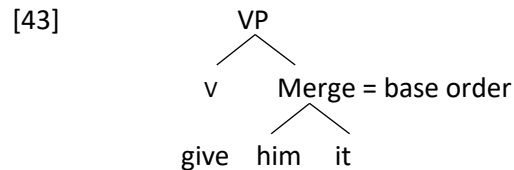
The PRN 'Her' *Mary* must be case marked by clitic {to}, just as in the other cases involving {of} and {'s}.

Note again how we cannot say \**'I want to give money Mary'*. This sentence is unacceptable since neither the proper name 'Mary' along with its object/pronoun counterpart 'her' wouldn't be case marked. But both *I want to give money to Mary*/*I want to give Mary money* are fine.

Note: The sentence *I want to give Mary money* brings our attention to the problem of how *Mary* gets case marked—since no necessary movement has been employed out of base-generated [IO, DO], and lexical {to} Case checking is absent. Well, *Mary* is first of all, not overtly marked, (only Pronouns in SE get overtly marked, perhaps explaining why we can't say \**'I want to give to Mary money'*), and secondly, we have a mechanism easy enough that can take care of Case, that of Structural case via the verb 'give'. [Give [Mary]]

is in the structural local domain where the verb 'give' is allowed to case-mark its complement/object 'Mary/her'. Since lexical case-marking is present, the extra insertion of case-marking {to} would be redundant, and thus ungrammatical as seen in the contrasts between the two structures \*'I want to give to Mary/her money' vs. 'I want to give money to Mary/her').

In summary of this section, a structure such as 'give it to him' derives the following steps:



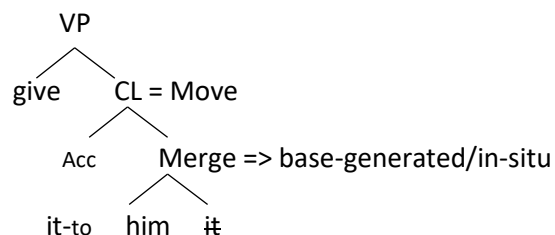
(1) Note how \*'give to him it' is unacceptable, while 'give it to him' is acceptable.

(2) The ProNouns 'him' is case-marked via base-generated/in-situ verb 'give' (or otherwise by default). But the PRN 'it' must also be case-marked (overtly so). So the PRN 'it' must raise to clitic 'to' (morphemic case marking). Note the grammatical contrasts of

(a) \**give it him*. (where PRN 'him' is left stranded without a case-assigning configuration)

(b) *give it to him*. (where PRN 'him' now receives a proper case-assigning configuration via CLitic {to}).

(c) *give him it* => *base-generated order [IO, DO]*.



In the illicit example found in (a), case marking cannot be doubly assigned to PRN 'it' because the dative verb 'give' has (earlier) already assigned case to 'him' via base-generated/in-situ order ('Give him it!'). (Ex. C).



[44] Given that the morpheme {to} now has a dual status (preposition and case marking), let's consider the contrast regarding phonology/stress (schwa reduction) between the following clitic case-marker {to} versus the prepositional {to}—two very different items:

(a) John gave (to) him coins. (base order): [V+PP] => {to} is a Preposition (no schwa reduction)

(i) \*John gave him them, (ii) John gave them to him

(In (a) above, it appears that when the Prep {to} is removed, and case of 'him' can be assigned via the verb, what happens is that the PRN 'them' is then left stranded without a case-assigning configuration. The structure in (ii) corrects this).

(b) John gave coins-to him \_\_ : N+CL => {to} is CLitic case marker (schwa reduction is allowed).

In (a), 'to' is a preposition and no schwa reduction is observed; 'to', if pronounced, must either be pronounced with stress /tú/ or may be deleted (whereby 'him' receives structural case). (Recall, the Noun 'coins' doesn't have to raise to be overtly case marked, and so it may remain in-situ in base-order with default case. Only Pronouns in SE get overtly case marked). In (b), the stress is weakened to a schwa.

(c)	(i) John gave it-to him.	{to}-Clitic
	(ii) *John gave to him it.	{to}-Prep

Recall, that for the example (c) above, 'him' has already been case marked structurally (in base-generated order) via the verb 'give', (cf. [40 (1)] ) (\*and so the prepositional structure in (c, ii) above is redundant for case marking). But the PRN 'It' still must be overtly case marked, so raising to the clitic {to} (as in a PG relation) is forced. Note that {to} in this respect found in (c, i) is a 'case-marking' clitic, and not a preposition, noting the possible phonological/stress 'schwa reduction': e.g., 'John gave it /tə/ him' as opposed to 'John gives to/tú/ him so much money'.

## Agreement

[45] Another formal feature we could consider is Agreement, which contains the dual features of Person, and Number. For instance, if the subject 'He' is [3<sup>rd</sup> person, singular], then the verb 'speaks' must match (so-called subject-verb AGREement): e.g., 'He drives'. We could speculate that the default setting here is the non-affix Infinitive-verb stem of 'speak', so that a child at the very early multi-word stage might say 'Daddy drive' whereby no AGR features is present. Coupled with a default Accusative case (as suggested above), we might expect to find utterances of the example 'Him drive car', and we do. Let's consider some token examples of such utterances dealing with the absence of Case and AGR below, as we draw our attention to what a speculative Proto-language would sound like.

## Note 4

---

### A Note on Artificial Intelligence and the critical recursive implementation:

#### The lagging problem of ‘background knowledge’

*Humans tell themselves stories in order to get themselves to work on this or that. It is almost always the case that these high-level stories are relevant only as motivation and not really relevant to what eventually happens in terms of technical understanding.*

Allen Newell

#### Opening Remarks

Most historians of the *Cognitive Revolution* consider the now historic 1956 MIT *IRE Conference* ‘Transactions on Information Theory’ to be the conceptual origin of the revolution. It was at this conference that three of the most important papers in the emerging field of AI would be read:

- (i) George Miller’s *Human memory and the storage of information* (coupled with an earlier 1955 paper *The magic number seven, plus or minus two: Some limits on our capacity for processing information*).
- (ii) Allen Newell & Herbert Simon’s paper *The logic Theory Machine: A complex Information processing system*.
- (iii) Noam Chomsky’s paper *Three models for the description of language*.

But it would not be long before splits would occur in the very defining of AI. For some, let’s call them the **AI-soft** crowd, despite the ever-growing consensus that the brain really did not function like a computer after all, (as was earlier suggested by the naïve ‘brain is computer’ metaphor of the time), the AI-soft crowd, against the push-back, were content to go their own way and see just how far they could actually push their learning algorithms in solving ‘real-world’ problems (eventually using Bayesian networks). Most early cognitive scientists of this time—while now at least partially acknowledging and accepting the fact that what they were doing was indeed not real ‘human-intelligence’ modeling—would nonetheless remain undeterred from learning about

how to improve upon these non-human-like networks. One AI-soft champion that stands out here would be Frank Rosenblatt and his *Perceptron* model for visual learning (1959-1962).

The other side of the split quickly emerged contesting that the brain is not just composed of neurons firing (viz., that the human brain is indeed much more than the sum of its parts). The rallying-cry would be that the brain is not at all data-driven, but that an inner *a priori* blue-print encodes how humans see and interpret the environmental data around them. For the hard-AI crowd, the human brain/mind cut (not necessarily promoting Descartes's dualism, but dualistic nonetheless), was said to 'boot-strap' a **Theory of Mind** (via a 'brain-to-mind' bootstrapping), and that such a theory was, *inter alia*, intuitive-based, (even superstitiously so), full of imaginary concepts (at times unreal)<sup>28</sup> and symbolic & categorical (rule-base)<sup>29</sup> in nature—all of which were viewed as being uniquely un-tethered to actual environmental stimuli & data<sup>30</sup>. This disconnect between (i) an inner computational algorithm and (ii) the environmental data would foster completely different predications and would run completely counter to any symmetrical algorithmic language of  $X=x$ , where weighted probabilistic outcomes supersede all else. This latter group which countered AI-soft might rather be called **AI-hard**, where any naïve attempt to model the brain by simple recurrent, Bayesian networks were pooh-poohed as being a simple product-calculation of patterns found in the data, and nothing more. This side of the debate was spawned by the likes of Marvin Minsky (see Minsky vs. Rosenblatt debates):

*'Where Rosenblatt would argue that his neural networks could do almost anything, and Minsky would counter that they could do little...'*

One essential aspect of the hard-AI crowd was how they viewed human language as human reasoning. The most recent contribution here comes from Judea Pearl (a UCLA cognitive scientist who some espouse as the one of the founders of second-generation AI). Pearl argues that Minsky's skepticism can be completely understood now in the sense that all Bayesian networks can do is achieve a probabilistic outcome as based on **symmetrical** language, of the sort  $X=x$ , as understood within (B.F. Skinner's) Behavioral theory of **Association**. Pearl shows how this is the symmetric language of algebra: If  $X$  tells us about  $Y$ , the  $Y$  tells us about  $X$  (a one-to-one relation). But the human brain/mind does not reason like this: e.g., 'rain may cause mud, but mud doesn't cause rain'. Pearl rather is animate that the only kind of complete AI that could deliver a real-like human thought-process (AI-hard, or AI-complete) must be based upon **asymmetrical** language

---

<sup>28</sup> See S. Toulmin's (1961) great book on the topic of a 'model-based' vs 'model-blind' dichotomy (attested in the rival approaches between Babylonian and Greek science).

<sup>29</sup> See Pinker, (1999) 'Words & Rules' Theory.

<sup>30</sup> A 'Poverty of stimulus' argument originally proposed by Chomsky. (See Chomsky 2002 pp. 5-6, 8 for review).

of the sort: X tells us something about Y, but Y doesn't tell us anything about X (a one-to-many relation). Such a calculus for asymmetrical language is non-algebraic (perhaps rather geometrical) and has only been conceived as a written language quite recently (over that last two decades). In short, what is needed is recursive mathematics of the kind Pearl describes. Pearl has worked up such a model for how to achieve such asymmetrical processing.<sup>31</sup>

- (i) Associative model:  $P(y|x)$  (typical activity = *seeing* on an iconic 1-1 relation).

Theory: Behaviorism, symmetrical language, probabilistic.  $\{\alpha, \beta\}$

- (ii) Intervention model:  $P(y/\text{do}(x), z)$  (typical activity = *doing* of a 1-many relation).

Theory: categorical, symbolic, non-probabilistic  $\{\alpha, \beta\}$ .

The former *recurrent* model tells us: What is this? (as based on frequency of data).

The latter *recursive* model tells us: What if? (as based on causal reasoning).

The above dual distinction of '**recurrent** vs. **recursive** structure' is what will be more fully expanded upon in this note on AI—a distinction which draws out a critical contrast between:

- **associative**/means (with the flat structure of  $[x,y]$ ), as compared to
- intervention/**recursive** means (with the hierarchical structure of  $[x[x,y]]$ ).

Pearl is one of today's most animate cognitive scientists who have come out strongly against any claim that mere association, as defined by the relation of naked data (which entirely relies on pure input-data of a brute statistical kind), can ever be an **Operating System** (OS) which delivers real human-like thought processing. For Pearl, a **recursive** and **causal** reasoning algorithm, at the very least, is required of such an OS.

Finally, as things stand today regarding AI, most cognitive scientists still find themselves folded somewhere along the Rosenblatt/Minsky cline: for example, considering current events, such a spectrum might look like this— (a first group) those faithful to self-driving cars are the children of Rosenblatt, while (a second group) those who entertain a healthy skepticism of its

---

<sup>31</sup> See J. Pearl (2018) *The Book of Why: The new science of cause and effect*.

human-like driving ability are children of Minsky... (See Gary Marcus and further discussions below).

## Overview

More than 20 years ago, it was already being claimed in the emerging AI world that, following a kind of Gestalt psychology, any putative ‘computing to human-behavior’ process could not simply rely on analyzing things into their atomic parts or logical symbols, and then to expect ‘symbol-manipulation’ processings of those parts to deliver any logical meaning. It turned out that real human cognition was much more fluid than that, as human behavior produced (at the very least epiphenomal/peripheral) noise within its own signal (viz., human cognition is not just the sum of its material atomic parts, even if we were to understand what those parts actually are, which, at the moment, is very far indeed beyond our current reckoning). Thus, human behavior cannot survive any material reductionism as would be required via AI. Two very hard problems, ‘flexibility’ and ‘context-sensitive’ procedures must surely be implemented in any viable AI/deep-learning algorithm.

As part of several ongoing experiments (chief among them today’s experiments for autonomous self-driving vehicles), what the ensuing thesis seems to be—which continuously emerges and resettles within both AI-advocate and skeptic camps—is that what we want from a strong AI program is not just to ask *what* the input is (as in ‘what’s the source of the encyclopedic entry?, etc.’), but also to ask *how* that source/input is delivered into the algorithm. While this latter source of ‘*how over what*’ is a property often over-looked by those building AI code, at least to my mind, it is a quintessential property most critical in implementation if we wish AI-deep learning to come closest to simulating true human behavior. In order for AI to avoid AI winter<sup>32</sup>, theory of mind & background-knowledge-based learning capacities must be implemented in the system: deep-learning computers must learn not just *what*, but also *how* its source-input is being delivered into its system.

Artificial intelligence (AI) implementations have exponentially grown over the second half of the last century. However, little can be said about a real ‘qualitative shift’ since first-generation computing. Today’s AI scientists are seemingly still grappling with problems—some of which are

---

<sup>32</sup> ‘AI-winter’ is a term used by AI skeptics (cognitive scientists who have long-held believed that AI will not (anytime soon) be able to simulate true human behavior. AI-winter also has an economic aspect to it since much of the current U.S. economy has priced-in the stock market billions of dollars of future speculation based on AI promises.

quite basic, if not altogether primitive—problems stemming from the fact that AI doesn't know how to get 'from here to there' in simulating human-like cognition. Real advances have rather come in a quantitative manner, with speed, size of memory, multilayer connections (so-called 'perceptrons'), etc., but little if any headway has come in a substantially qualitative manner.

Today, as was the case back in the 1950s, most AI-coding operations assume a 'catastrophic-decision-based' platform where all its recurrent operating system (OS) can do is statistically *average* weighted input [A] with weighted input [B], whereby the weighed and statistically-average product is the mere sum of its parts [A, B], etc. In this linear and recurrent manner, [A], [B] have been simply combined with the two being averaged to produce  $[(A), (B)=[A, B]]$ . So-called 'simple recurrent networks' (SRNs) with an OS based on input-outputs with recurrent loops cannot remember past individual inputs since all information has been consequentially merged. The OS can't recall past inputs more than one single computational time-step (CTS) away: they hold no history. (This computational amnesia will become a point of interest to us if we wish to simulate human-like cognition, such as intuition and background- knowledge). But while they can average, they then can't properly adjust, it seems, given their tight restrictions of 'locality' (another point of interest for us).

This is precisely how *recurrent* OS differs with *recursive* OS's. In human-cognitive processing—which rely on non-local variables, as well as recursive multichannel processing—inputs, even of the [A, B]-type, resend via feedback loops which may go against the grain of averaged distributional weights and frequency effects. Such loops have an uncanny ability to remember a CTS many steps away. In other words, the information is preserved throughout the computational history, thus allowing not just for averages to accumulate across different pathways (say, from point- $\alpha$  to point- $\beta$  given parallel processes), but that such accumulations can be the result of a historical record, where previous input (lost in SRNs) can be recalled to add an adjustment across variable pathways. This in essence is how human learning works: where (de)learning can be established based on past errors and/or processing glitches.

Now, while this narrative runs for earlier SRNs (of the simple, linear and recurrent type found in G-1 (see below for history), the same processing essentially is intact for more up-to-date complex recurrent networks (CRNs). Whereas SRNs (with no more than one vertical stack of inputs) had notorious trouble generalizing to new and novel items, CRNs, now allowing for more than one hidden unit (having multiple hidden units), could multi-channel in the sense that feedback loops could maintain some history over many CTS. Given this added feature, however, the final actionable processing by the very nature of its OS, had to give priority to local constraints.

A case in point is where we currently find ourselves regarding autonomous (self)-driving vehicles (ADV). These operating platforms are still 'catastrophic-decision-based'—they

essentially are supped-up SRNs which rely on averages of an input stored over time, but where the actionable final input is merely the best calculation that has been added over, say, thousands/millions of generations (CTS) (data-driven experiences). Each decision is allied to a specific point in the spectrum of a spread-out data field without so-called *gradual decay* of past information which might be preserved in any calculus. Irrespective of these multi hidden units, the final input has no memory of how those exchanges took place from point  $\alpha$  to  $\beta$  between various generations. Why it should matter becomes a point of contention for us regarding how we believe human-cognitive learning takes place.

At the very earliest conception of AI, most scientists assumed a ‘brain-to-computer’ analogy with the rationale that since brains are essentially composed of neurons (along with a patch-work of neuro-nets) than what AI first needed to do was begin an OS that mimicked low-level neuron processing. Starting ‘small’ is always the best path forward whenever beginning a new scientific enterprise—whether it is with first studying the earthworm for genetic material before inquiring about the human genome, to analyzing young children speech to understand the properties and structure of adult language. So, given this logical strategy, the first AI OSs attempted to mimic lower-level neuron firing (with the Hebbian cliché of ‘what fires together wires together’). SRNs with their local neuro nets seemed to logically fit the model. In the historical context of behaviorism, the model gained acceptance.

### **A very brief AI history**

**(G-1)** The very first pioneers of Artificial Intelligence (AI) (Generation-1 (G-1)) were actually mathematicians and logicians (intellectual philosophers of the Rationalist persuasion). In 1955, Allen Newell and Herbert Simon created the **Logic Theory** approach to AI (at The Rand Corporation think-tank in Santa Monica, California). They held a rationalist philosophical view (from Descartes, Leibniz, to Kant, Husserl). This was traditional to their fields at the time—namely, that the ‘body-brain’ cut was indeed real (following from a Cartesian principle) which meant that there remained a ‘disembodied-mind view’ which saw that knowledge could only arise out of abstract symbol-manipulation: that knowledge begins in the mind, and that the world is then copied outside from the inner mind). (The neo-Cartesian philosophy of the 17<sup>th</sup> century saw its resurgence come from the Chomskyan paradigm (**nativism**) of the 1950s, based out of MIT). Abstraction was assumed since, in their rationalist framework, that was the only way the mind could possibly work.

In this view, inner-subjective symbols and rules (brain/mind) necessarily linked to the objective things in the outside world to give meaning. Hence, any sense of AI-meaning could only arise via symbols and/or categories. (Recall, the neo-Cartesian effort of Chomsky came out of



studies of human language, syntactic analyses and child language acquisition). One could claim that G-1 logical-theorists really didn't take the word 'artificial' in the term 'artificial intelligence' seriously, since what they were in fact grappling with was 'real' intelligence of the human-logic form. In the end, while this 'cognitive-revolution' version of AI enhanced our understanding of symbol manipulation of logical forms (the 'brain as calculator' metaphor, the brain as 'rule-based computer', or the brain as a 'digital operator', etc.), it ultimately would fail since it avoided the *hardware* problem of how the brain is really structured. It would later be rightly claimed that the brain is not really structured in abstract symbols (nor is it a computer), but rather the brain/mind is comprised of billions of neuro-net connections (more in-line with features/properties of connectionism).

So, the hardware question became a critical one: Do we wish to simulate what the brain is actually doing at the lowest levels of processing (at the neuro-network connection), or do we wish to just stay within abstract theory (of symbols and rules) and see how far we could get? Well, the end result was that Newell and Simon really didn't get too far with symbols and rules (outside of basic tasks), and so the next generation would more seriously have to reconsider the hardware problem and develop neuro-net connections. This hardware problem was what led to the second generation (G-2) school called 'connectionism' of Rosenblatt (1957).

Another factor in G-1 was the (false) premise that 'digital-seriality' was necessary for human behavior/thought (so-called 'serial processing'). Otherwise, it was thought that under a human 'analog- parallel' mode of processing (a best-scenario) one part of your human cognition could simply cancel out the other, or, perhaps (a worst-scenario), one part could not stop what you are thinking in the other part. However, upon closer examination, this serial/digital approach to human thought also turned out to be wrong: (the human mind/brain is an analog machine (not digital), a parallel-processing machine (not serial) (as observed by Patricia Churchland).

But human parallel processing turns out to be a very large problem indeed—a problem still yet to be completely overcome by our fastest computers in current AI. (Problems such as *Common-sense*, *background-knowledge* and *intuition* are very unique human-specific *procedural* activities and seem to defy serial *declarative* processing—they are very likely to be the murky residual result of 'parallel' processes between two or more modes of input/output channeling at any given time). Hence, any attempt at AI could never simply be about 'data collection' (declarative facts), whether with G-1 linear symbol manipulation or G-2 neuro-net connectionism (the former dealing in abstract logic, the latter in simple recurrent networks (SRN). Rather, what future AI would have to do—well beyond the earlier prosaic modes of programming—would be to take 'data collectively' (procedurally) over a spread of multi-layer processing levels.

The trick, as we will discuss below, is that each time-step of the 'data collective' (*pace* 'data collection') must be self-preserving and remain distinct—that is, the data can never be

compromised at any of the intermediate steps as a result of, say, when one datum  $\{\alpha\}$  is combined with another datum  $\{\beta\}$  and then averaged together as a new single result  $\{\gamma\}$ , thus losing each datum's individual and unique value. This problem in fact was what plagued the two aforementioned earlier modes of AI, and, moving forward, is what continues to plague AI/deep learning today.

**(G-2)** But in the beginning of G-2, early networks were very limited in the sense that they were 'vertically processed' (stacked on top of one another) and had no more than two layers of stacking. These units were processed 'vertically' as linear units without so-called feedback loops. In other words, the strict associative (**behaviorism**) of a one-one input-output algorithm was all that was assumed. Frank Rosenblatt (1957) (at Cornell University, Cognitive Systems Research Program) was the first to attempt a **connectionist theory** (of binary classifiers) towards human behavior (vision) in this way. His *perceptron*—an attempt of a first computer program system that could learn new skills by associative trial and error—used a type of G-2 neural network that simulated human-thought processes. Of course, while G-2 was an advance from G-1 symbols, the problem was that the G-2 neuro-network was much too limited both in memory and in speed, and as a result ultimately would fail.

The classic Minsky and Papert paper (again, a product of MIT) quickly shot-down any assumption that human cognitive learning could be simply mapped in such a stark binary and linear manner (again the brain is analog, not digital). Also, by this time (1956) Chomsky's first important paper had already come out (seemingly in support of the doomed G-1 model) defending the skeptics of linear behaviorism in his paper *Three models for the description of language*. Chomsky, though seemingly in support of the 'backwards-looking' G-1 symbol manipulation theory, was correct and 'forward-looking' in calling out that such a simple associative AI theory would only ever be able, at best, to map top-down input already installed in the program to tags matching the output. AI, under such limitations, would never simulate human cognition.

But, advances would still be made on the system: e.g., such top-down encyclopedic tagging (with knowledge already inputted from the top) would ultimately pave the way for so-called 'micro-worlds' of expert AI-systems (closed-worlds where only specific, relevant material to the task could be drawn upon and where the so-called background/commonsense problem that humans innately enjoy (seemingly without step-processing) wouldn't be required). In the main, such simply associative learning was attempted by most connectionist programs of the day with little success. For about twenty years thereafter, AI was both frustrated and stagnant: many of the basic problems we thought we could solve turned out to be ridiculously out of reach: (such

problems may be so-called ‘hopeful monsters’<sup>33</sup>, requiring a completely new way of bootstrapping our understanding. A ‘saltation theory’ is required as in the notion that the evolution of a species might not come from incremental-gradual adaption over a long period of time, but rather might ‘jump-up’ (‘salto’, Latin for ‘jump’) all at once and rather quickly via an ‘exaption’ from micro-changes either in the environment (via *neo-Lamarckian inheritance*) or in the genome itself (via *cryptogenetics*).

For instance, consider one such aspect of the ‘memory and storage’ problem related to encyclopedic micro-worlds: a seemingly trivial task for human behavior, but a major problem for AI seems to be the case, even at a basic associative level, that if one were indeed capable of successfully down-loading all human-encyclopedic knowledge into a connectionism system, there would still exist the problem of ‘how’ one would be able to access the right material needed at any one time (at the right time). Humans get the ‘accessing problem’ right away. It’s called procedural-access of declarative-knowledge. We don’t even ask the question of ‘how’ to access ‘what’ ‘when’. It’s just part of our background, common-sense knowledge.

Well, it turns out that even if we get the ‘memory & storage’ problem right, we still have to deal with the other side of the equation, the proper ‘access & retrieval’ problem. Hubert Dreyfus (while at the Rand Corporation, contemporary with Newell and Simon) was once quoted (talking about encyclopedic machines at that time) as saying (my rephrasing in italics) ‘The problem *may not be* how you can get the vast amount of knowledge into that super-big memory (he doubts that you can), but even if you did, how could you possibly access the part that you needed in any given situation’.

So, for G-2, some of the problems they were already grappling with was somehow related to this commonsense/background knowledge we all have, apparently inherent in our design of the brain/mind, with spin-off attributes leading to subjectivism and theory of mind. The next generation G-3 would either have to deal with this hopeful monster in new and creative ways, or else sweep it under the carpet as some imposter artifact, as some unique feature outside the peripheral realm of AI, and somehow not implicated in the grammar coding of future expert systems. In other words for G-3, the macro-world of human commonsense-background

---

<sup>33</sup> Richard Goldschmidt was the first scientist to use the term ‘hopeful monster whereby it was thought that ‘small gradual changes’ could NOT cause a change between the (tiny) microevolution-level and (large) macro level. Hence, something quick and large would have to be the source, a so-called ‘saltation theory’ (or quick jump, as in Steven J. Gould’s ‘punctuated equilibrium’). In his book *The Material Basis of Evolution* (1940), Goldschmidt wrote ‘the change from species to species is not a change involving more and more additional (small) atomistic changes, but rather a complete (large) change of the primary pattern or reaction system into a new one, which afterwards may then produce intraspecific variation by micromutation’.

knowledge will either be an impediment to AI (perhaps provoking leading AI-skeptics of the day), or it simply won't factor in to the AI equation.

**G-3** Then a breakthrough came in the 1980s (largely from work happening at the University of California, at San Diego (UCSD)) by the likes of Geoffrey Hinton and James McClelland, David Zipser (all three who contributed to the seminal book *Parallel Distributed Processing (PDP)*), Paul and Patricia Churchland, then later Jeff Elman—all of whom in one way or another made major contributions to work in 'parallel distributive processors'. Most crucially, and different from earlier associative-connectionists models (of Rosenblatt), these new and immensely more powerful PDPs were of a very new kind of processor—whereby so-called 'hidden units' (hidden layers) and 'back propagation' (feedback loops) were inherently encoded into the architecture and design of these superfast computers.

Such advanced work at UCSD rekindled old and often heated debates in the 1990s between strong AI-advocates such as Daniel Dennett, James McClelland, and Jeff Elman (of the Peter Norvig persuasion)<sup>34</sup> versus AI-skeptics (of the Noam Chomsky persuasion) such as John Searle, Jerry Fodor, and Gary Marcus<sup>35</sup>. I personally can recall the debates between the latter-two counterparts (Elman v Marcus) in the 1990s (while I was a doctoral student with Harald Clahsen (University of Essex)). From the tenor and passion of their arguments at the time, one got a pretty good sense of the direction to be taken and all the problems to be had facing future AI as it would pave its way towards the 21<sup>st</sup> century.<sup>36</sup>

The 'PDP-promise' included a framework which was closest to an artificial neural network found in human cognition. The model stressed the parallel nature of neural processing and distribution.

---

<sup>34</sup> web-link no 29.

<sup>35</sup> web-link no. 30. See Marcus for a recent summary.

<sup>36</sup> Web-link no. 31.

### The new PDP promised: <sup>37</sup>

- (i) that the computer architecture would connect many more than the previous G-1 simple 'two-layer unit'. (In fact, PDPs could now use, at least theoretically, unlimited amounts of connective networks (perhaps limited only by the size of its hardware),
- (ii) that rules/symbols would no longer be the engine of computation—rejecting the putative 'human-like' (and more analog) manipulation of rules over the favored (and more digital) connectionist computation (but recall Patricia Churchland's observation above that the human brain/mind is **analog** and **not digital**),
- (iii) that PDPs would allow for real 'deep-learning' to take place via feed-back loops which fed into multiple hidden-units. These new 'post-hidden-structured' unit would in turn loop backward from the incoming data-stream and return it as a new forward signal.

### G-4 Recursion.

*We are about two thousand years from having a serious theory of how the mind works... Jerry Fodor.*

I'd like to address our contemporary Generation-4 (G-4) in specific terms as it currently relates to the micro-world of **autonomous-driving vehicles (ADV)** (currently hot off the press). First of all, most cognitive scientists along with strong AI researchers today recognize the notion (now, a close truism) that the mind is not a digital computer. Sure, at one level, say at the lowest level of processing, specific neuron firing may have certain attributes analogous to binary operations (of off & on). But surely, learning and knowledge is not the mere brute result of any single statistical product: in brief, human learning is not a kind of 'statistical inference' but rather a sort of formation derived by abstract theory. So, at one level, at least at the lower level, a kind of associative/connectionism may in fact be what is going here: it is true, for most micro-word

---

<sup>37</sup>PDPs allowed for: Processing units, represented by a set of integers, to be activated for each unit. Each output function for each unit is represented by a vector of time-dependent functions on the activations. An activation rule for combining inputs to a unit determines its new activation, as represented by a function on the current activation and propagation. A learning rule (algorithm) is implemented for modifying connections based on experience, which are represented by a 'change' in the weights based on any number of variables. An environment that provides the system with experience is represented by sets of activation vectors for some subset of the units.

expert systems (say, analogous to lower-level mind operations), statistics do seem to drive the computational result (and the best result is always on the upside of the ‘statistically averaged’) The bell-shape curve of a competency of a skill follows a statistical curve. But it may be that the skill-competency of activity {x} is not the same as knowledge of {x}. Statistics may grant us an understanding of how a hypothesis of {x} is tested against the body of data {y, z}, but statistics alone cannot grant us an *understanding* of how the hypothesis was generated.

It seems a more abstract, higher-level processing beyond mere statistics is required. But it is true, at lower levels, a kind of Hebbian<sup>38</sup> calculation seems to hold: Hebb’s expression ‘*what fires (statistically) together wires (statistically) together*’ is now a well-accepted biological truism. But it seems, still at G-4 PDP-connectionism, that all we have done is push the associative, lower-level ‘binary statistic’ up towards a pretend ‘higher-level’ processing. But it has not arrived there. Are we just faking learning? We do this faking whenever we claim that self-driving cars can ‘learn’ beyond statistics: but statistics have no means of learning: numbers can’t ‘learn’, they can only ‘average’! Sure, it may turn out that their averages improve and the bell-shape curve shifts upon more and more experience, but there is no learning in the sense of how humans learn. Considering the ADV method of learning, it may be that a hypothesis of the driving act can be measured and tested against a body of data, but even so, we still don’t know how that hypothesis was generated in the first place.

For some AI researchers the question may not even come up: What does it matter to them if they don’t understand how the hypothesis was generated when they have successfully tested their data? But, sooner or later, it will matter—as when ADV hypothesizes that a very large white scene just in the view ahead is statistically analyzed, averaged, and wrongly interpreted as free-open road (and the self-driving car runs into the lateral side of a tractor-trailer killing its unaware passenger), or, as a preliminary report from federal safety regulators have detailed, when an ADV’s sensors had in fact detected a ‘woman pedestrian’ but its decision-making software discounted the sensor data, concluding it was likely a false positive (and the ‘woman’ is struck and killed). Again, **Learning** is never a statistical gathering of atomic parts, but rather true learning happens at a ‘higher-level’ processing (not the sum of its parts), it is abstract, and it seems to tap into fundamentally different but parallel modules of the mind. I’d rather prefer to postulate that for future G-4 AI, the question should be ‘Can we get some success, even minor success, out of a machine that goes beyond statistics?’ At least we would then be telling the truth: with such a question, we realistically tether our promised AI to the very primitive associative methods we have at one’s disposal, and nothing more beyond that. As Fodor says, it may take us two thousand

---

<sup>38</sup> Donald Hebb way back in 1949 showed that neuron clustering could alter synaptic coupling which in turn could change synaptic strength.

years to understand how we get from low-level firing of neurons to higher-level consciousness and thought. When we finally do, only then can we claim we understand how the mind works.

At the moment, our best algorithms for ADV semantic imaging (the ability to recognize the environment) rely totally on trained end-to-end, pixels-to-pixels, semantic segmentation.<sup>39</sup> But, while the best AI-imaging software currently being deployed, as connected to deep learning, seems to be very good at very difficult imaging averages, the software has major catastrophic breakdowns when dealing with the simplest of things. Here's such an example (quoted from Gary Marcus *NY times*, 2017):

*Even the trendy technique of 'deep learning', which uses artificial neural networks to discern complex statistical correlations in huge amounts of data, often comes up short. Some of the best image-recognition systems, for example, can successfully distinguish dog breeds, yet remain capable of major blunders, like mistaking a simple pattern of yellow and black stripes for a school bus. Such systems can neither comprehend what is going on in complex visual scenes ('Who is chasing whom and why?') nor follow simple instructions ('Read this story and summarize what it means').*

Such installment of 'semantic information processing' into the operating system reminds me of initial problems beset early SRNs of the 1980s, when problem-solving programs went in operation hoping to 'solve problems' but without any 'knowledge of relevance' of the problem itself. It was the improbability of such tasks that got AI cognitive scientist to rethink the importance of belief, intuition, and commonsense/background information—what would become known as the 'commonsense problem' (See Dreyfus's book *Mind over Machine*).

What Gary talks about in the review is the need for future AI to have two modes of simultaneous processing: namely, slow **bottom-up** processing which is sensual in nature (environmental) (perhaps equating to so-called 'declarative' knowledge), plus rather faster **top-down** processing which is theory-formation based (equating to so-called 'procedural' knowledge). Such an AI dual processing would mimic what we find at two levels of human processing—with the lower-level and slower **serial processing** having a connectionist-network quality (say, at the neuron level in the brain) and **parallel processing**, which is bootstrapped by lower-level **connectionist** but simultaneously acting in an autonomous manner from the lower level having a (very human) symbolic and rule-based theory formation quality. The former serial processing occurs at a much slower in speed in humans and is conscious (as in 'step-by-

---

<sup>39</sup> web-link no.32. See Marcus for a review.

step'/ingredient building), while the latter parallel processing is much faster (and perhaps subconscious in humans), and is rule-based and abstract. I'd extend the analogy by suggesting that the human **brain** is rather a slow serial processor (with neurons firing to make connections)—that's why computers so greatly outmatch human capacity, computers are extremely fast serial processors at every step of the way at the lower levels. Humans get muddled-up between steps as we attempt to surface up bottom-up results to our top-down understanding.

It is my attempt below to try to simulate a thought experiment of what it would mean for an AI algorithm to have such a dual mode of processing, whereby both (lower-level) recurrent processes overlap with (higher-level) recursive processing.

## Neural Networks

Artificial Neural Networks (NN), as the basis for most SRN operating systems, was the AI response to how brains were thought to compute. The analogy of the brain as a computer, on one hand, seemed natural enough since brains are indeed bundles of neuro-circuits (of a binary nature [0,1] off & on switches). NNs, in this way, were thought as sufficiently capable of handling most brain-related tasks since such modeling were essentially non-linear. NNs are extremely capable of modeling nonlinearity tasks such as classification, associative memory and clustering. Such a diverse means of processing allowed NNs to be modeled to solve many related tasks. The problems early NNs faced were mostly due to limitations of training: large amounts of time had to be reserved in order for the NN OS to learn and cope with the task at hand. Also, as stated above, the appeal to NNs was that it, like the brain, consisted of a collection of neurons (or perceptrons/nodes), and each perceptron had its assigned input to output production. Training of such NNs (multilayer perceptrons) involved error-feedbacks: each time an error value is fed back, its weights of that connection is adjusted so that over time, learning can be achieved. Over time (training) the averaged and weights were adjusted so that better approximation could be produced (hence, learning). The factors behind NN amount to how many hidden units or layers were predetermined in the system for feedback. Also, another factor was how the units were labeled (either via mere associative chunk of item (for example 'cat' [cat]), semantic feature of item ([cat]: {+furry}, {+ four-legged}, {+ Purrs}, {+whiskers}, {+ animal}), or even based on its phonological units ([Cat]: onset /k/ nucleus /æ/ coda /t/). The most common use of NNs are called multi-layer perceptron which have multi inputs along with hidden units, where labeling and identification of input is determined by either label of item (chunk), semantic (features) or phonology. However, what's been overlooked so far is how to employ syntax into such NNs. Recalling that it is precisely syntax that allows to depart from strict one-one associative/recurrent formations and allows for recursive/hierarchical structure to be employed as part of its neuro-

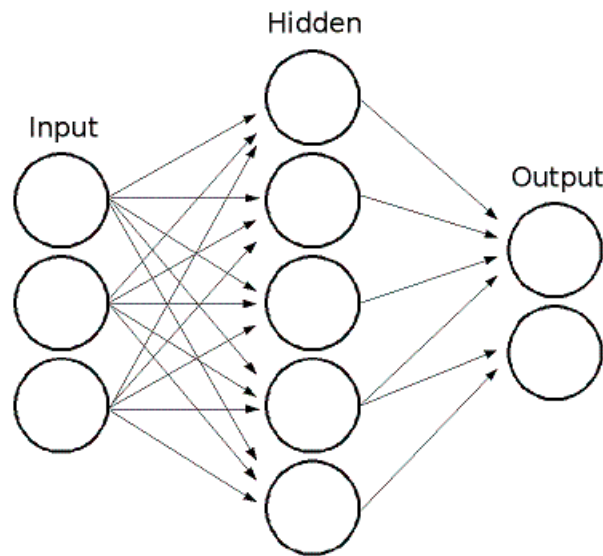


network—recalling that a neuro-node has only one output value. Neurons don't actually calculate. They simply pass on the input value to the next layer. The notion of 'learning' for such Networks is tantamount to associative-memorization systems which operate under 'rate of frequency' (occurrence, distribution, position, etc). As an exercise, try to imagine how such a network would treat {s} in the following two words:

[1] 'Fix' IPA = /flks/ processing structure = [flks]  
 'Speaks' /spiks/ [[spik]s]

(Note: For NNs, there would be no way to tease out the syntactic distinction that both final {s}'s, though both have exact position (final), have the exact distribution (\_ks) (both verbal), and have the same sound /s/, nonetheless would have competently different underlying processing. Only an NN with recursive treatment and comparison of the two {s}'s would show that the {s} in the verb [fix] is part of the stem, while the {s} in the tensed verb [[speak]s] is an inflectional features of the syntactic features {3P, SG, Pres} (third person, singular, present tense)).

Fig. 1 Multilayer feed-forward neural network (Google© 'free-to-use' image).



(For back propagation, neuro-pathways would also go backwards, in so-called error-trained learning).

### Back-propagation models

In back-propagation models (multi-layer perceptrons) an 'external teacher' (or supervisor) is required to steer learning. This supervisor either must be pre-installed in the system (innate architecture) or can come via some pieces of information found in the environment. In any case, learning is not the mere product of input and outputs but rather is steered via some parameterization. For AI enthusiasts who claim that the greatest feature of connectionism is that it assumes no innate structure would be hard pressed to explain-away why such supervision is required in the first place. But recall, that even the most ardent anti-innate people have to at least assume some small amount of innate structure in order to first calibrate an initial (neutral) weight for weighted learning.

But, to a large degree, such an 'external teacher' could be made available in the environment itself—e.g, a small piece of info that could then bootstrap larger chunks of info, thus leading to wholesale learning. Once trained, such models could, for example, predict the next word in a sentence (in a sentence-learning program). For instance: what is the next word in the sentence 'The dog chased the\_\_\_'? For a sentence-learning algorithm, 'the dog chased\_\_\_' would have an weighted averaged for [dog= A] + [chase=B\_\_\_] which would trigger [C=cat] if we were to have weighted nodes which push the probability connection of dog+chase+X as X is a Noun (some other animal, high average is [+N, cat], etc.). This could be learned from the environment if

previous sentences with certain semantic features behaved accordingly (*dogs bark at cats*, *dogs fight cats*, etc). Of course, one problem might be for a sentence-prediction algorithm to ascertain embedded structures such as ‘The dog<sub>i</sub> the girl chased\_\_\_’ (predicate what comes next: what is [\_\_\_])? Such structures which involve object-raising may prove difficult for mere SRNs without necessary recursive networks. (We’ll return to recursion in latter sections).

## Claims ‘For’ and ‘Against’ Multilayer Perceptrons (ML-P).

### ‘For’ ML-P

**Claim 1.** As mentioned above, **multilayer-perceptrons** (ML-P) seemed to have a lot going for it since they did well to mimic what we (we thought) we knew about the brain: nodes are analogous to neurons and connections to synapses. To a large degree it is true: brains are composed of bundles of neuro nodes (synapses), they are binary based (+/-habituated) and are valued (=weighted learning) as determined on the averaged frequency of input. Hence, brains/neuro-nets were understood to be nothing more than little *Skinner schemes* (B. F Skinner) as understood in most behaviorism models of the day (viz., associative-learning schemes). It was taken for granted that connectionism of the day was the best fit for how the brain actually functions. This ‘brain to computer’ analogy gained much favor in the AI world, (at least until Minsky & Papert’s 1969 paper came out to prove severe limitations on such meager associative models). Also, it was becoming well established at the time that the brain didn’t work as a symbol manipulator (based on rules) since rules held no biological foundation (they are only abstract conventions).

These two factors placed together (brain as neuro-net, not as symbol manipulator) paved the way for NNs over the next couple of decades despite Minsky & Papert’s disclaimers<sup>40</sup> and others how question if the brain was really digital after all (Minsky, Fodor and others *pace* Churchland and others) as discussed elsewhere in this text). Others preferred ML-Ps since they required very little in the way of innate architecture (although, that is not quite right since even with advanced ML-Ps, some amount of pre-installed (innate) values had to be assumed in order to get

---

<sup>40</sup> First Minsky & Papert demonstrated the severe limitations of earlier SRNs which lacked hidden units, and even went so far to dispel how systems with abundant hidden units would still remain insufficient in handling human-like thought-processing. (See Hadley 2000 for review).

the ‘weighted’-values started). People such as Churchland were animate that brains could learn directly from the environment and no innate structure had to be presumed. This innate debate that carried over into the AI field was a direct result of debates being had in the field of language and child language acquisition (Noam Chomsky)—arguments such as the ‘poverty of stimulus’ debates, etc.

**Claim 2.** ML-Ps also seemed to gain much favor in the AI community since such networks didn’t assume any innate structure (which was the Holy Grail for AI). But, as hinted above, as it turns out, this is not entirely right: some amount of innate structure is always required—it being either assumed in the architecture, or is assumed in the way a search is conducted of the environment.

**Claim 3.** True: the style of learning achieved by ML-Ps do seem to mimic human learning to the extent that e.g., **over-regularization** can be produced by the learning algorithm when it comes to the learning of language inflection (in child language development, as stage exists which show examples such as *singed*, *breaked*, *taked*, *goed*, and even *wented*—all examples of over-generalization of the regular rule (past {ed}) onto irregulars. Such a thing we would want to see of a real learning algorithm. However, only later with back-propagation do they correct such irregular sound formations (e.g., *\_ing* > *\_ang*, for *sing* to *sang*) and so only when the input has been extremely modified (i.e., abrupt adjustment) to fit such sound pattern sequences. In other words, the whole vocabulary had to be reinstalled in order to cope with irregulars, and an entirely different vocab index was used for regulars. The problem here is that the networks was be sheltered by two different (artificially supervised) inputs since any ML-P can only work via a single mechanism (a single mechanism model). (See Marcus et al. 1992 for review. See Pinker 1999 for review of a dual mechanism model).

Even when trained with irregulars-vocab input, the ML-P was unable to handle so-called *denominals* such as ‘ringed’ (here, where an inherent (irregular) verb takes on regular qualities behind the derivational processes of Noun (*a ring*) to Verb (*to ring*) => [N-V {{ring} ed}]). It seems at the very least a dual mechanism model is needed to handle such diverging data regarding regular vs. irregular processing. (See ‘Dual Mechanism Model’ below).

## **‘Against’ ML-P:**

### ***The Sandwich problem***

**Claim 1.** It remains clear that ML-P can’t deliver the full range of human thought—viz. ML-Ps seem unable to capture a class of functions such as [+/-partitive, +/-individual] (a typical ‘object [+/-specific]’ feature associated with determiners (*each, some*) as compared to pronouns and proper nouns—viz., a comparison of ‘kinds/Noun’ versus ‘individuals/Pronoun’, etc). For instance, in the dual proposition *John kissed Mary<sub>j</sub> & Fred kissed Mary<sub>j</sub> [kiss M [J&F]]* the ML-P can properly treat ‘Mary’ as one in the same persons [+individual] (as an individually-coded item)—viz., as the Pronoun (Mary/she) which properly triggers a [-partitive/+individual] function. But the harder problem rather may be with simple nouns. For instance, in the dual propositions *John ate a sandwich<sub>j</sub> and Mary ate a sandwich<sub>k</sub>* an ML-P may not be able to recognize the referential distinction that the *sandwich* is not one in the same—namely, ML-P would rather treat both *sandwiches* as the same, subscripted with same index (sandwich<sub>j</sub>).

In order to capture the partitive function in coding, the register/node which codes for generic ‘sandwich’ say e.g., [0110110] would also have to be doubly coded for *sandwich-1* [01101101] vs. *sandwich-2* [011011011], etc. Such itemized weighted functions would place a heavy burden on the limit of coding (which may create problems associated with so-called ‘cross-talk’, e.g., when an overlap of coding creates ambiguous functions). (See *Blending* problems below). It rather seems what we want out of a well-functioning ML-P is the ability to represent both relations between **local nodes** (= sandwiches), as well as **distant variables** (= sandwich<sub>j</sub> & sandwich<sub>k</sub>)—and to be able to compute across the two representational systems. (Keep in mind that our upcoming proposal calling for a *dual mechanism mode (DMM)* (see below) will be in a unique position to do both: allowing for (i) local neuro-nets to represent +frequency-based registers (nodes), while (ii) non-local relations between variables of indexes/diacritics are maintained. It is here that we find that non-local representations between nodes and variables are unbound by frequency of the stimulus, but are rather recursive/symbolic in nature. Hence, by definition, a DMM is inherently a *parallel processor*, whereby both [+Freq] sensitivity of nodes and [-Freq] of variables can be simultaneously coded.

Recall, that the only source of information an ML-P has to ‘sandwich’ is a set of input nodes that are activated at the moment the item ‘sandwich’ is retrieved. There seems to be no way a *single mechanism model (and serial processor)*—which is completely reliant of local-node frequency—could go beyond this one-one associative coding. Of course, in human-thought reference (pragmatics), people assume that there must be two sandwiches, whereas the Proper noun Mary codes for an individual.

Another possible problem with ML-P is the matter of frequency (recall, that all neuro-networks rely heavily on one determining learning-factor—that of ‘frequency’). For example, in work carried out by Harald Clahsen, the German plural {s} was found to function as the default setting for plural. So, for example, when a novel word was introduced to a German speaker, the default plural of the Noun [N] was add {s} : [[N]s]=Pl. However, the problem with this is that German plural {s} is not the most frequent in the data. In fact, when compared to alternative plural inflections {en}, {er}, the {s} only occupies less than 10% of productive plurals (Clahsen 1995, Marcus et al. 1995). In this case, if frequency is the generator behind (weighted) learning, then there would be no way to handle such anomalies as default rules untethered to frequency. The only way to guarantee that ML-Ps can generalize from limited data (in the ways that humans do)—given that the German plural data show highest frequency plural for {en}, {er}—is to incorporate a **Dual Mechanism Model** (DMM), a system for frequency (irregulars), and a system for symbols (rules, defaults). The question becomes: How can a non-frequency-based weighted distribution be incorporated in a frequency-based ML-P system?

Perhaps the biggest problem with ML-Ps is that they are simple recurrent networks, that is, they merely can **blend** (combine) input data.

## Blending

The problem with blending is that in the simple combining of [A], [B], both individual values attributed to each item is lost as soon as the two items are calculated and merely averaged together—i.e., *blending* doesn’t preserve the individuated information (information is lost over the spreading/blending of the two). Take for example, the representation of ‘A’ as [1010], ‘B’ as [1100], ‘C’ as [0011], and ‘D’ as [0101]: if we blend, say ‘A’ with ‘D’, we get [1111], but so too would be the result of ‘B’ & ‘C’. Hence, there is no way to distinguish the individuated items. In this case, we have lost the unique code for the specific items once blended. In order to save the information over, say, several computational time-steps, we would need some form of recursive measures, such as [1010[0101]] , set = {‘A’, ‘D’} with order unambiguous set. Here, info is preserved. Conversely, such loss of info as a result of blending matters when ‘words’ are assigned such binary codes (as with standard binary code ASCII)—where e.g., *A dog* could be coded as ‘A’ and *A cat* as ‘D’, hence losing the proposition ‘cats and dogs’.

A second problem with blending (also particulate schemes) is that no word order is gained from out of the mere combination. As is seen below within the treelet structure, the phrase ‘box inside pot’ has to somehow insure that it is the ‘box’ that is inside the ‘pot’ (and not ‘the pot that is inside the box’).

A third problem with ‘blending’ here is known as *superpositional catastrophe*. Recall, that this blending problem is what we find of non-recursive sister relations found in our main discussions of syntax, e.g., of the [house, boat] vs. the [boat, house] variety. Recall, that as long as the two items merely blend as sisters [ ]+[ ], there is no way to capture hierarchical structure—viz., is it a kind of *boat* or a kind of *house*? The same would be lost regarding [dog and cat]: is it a kind of dog or cat? It is only through recursive measures [ [ ] ] that the individuated information of the item is preserved: [house [house, boat]] is a kind of ‘boat’, with unambiguous word order and meaning. Another problem regarding ‘blending’ is if you are dealing with the combining of **semantic features** e.g., {cat: [+furry], [+animal], [+purrs]}. If one feature overlaps, superposition catastrophe of semantics could surface. Such a problem is cited in the literature regarding {penguin} whereby one of its semantic feature [-fly] bleeds into the semantics coding for ‘fish’. (See Marcus 2001, p. 94, where he refers to this as *catastrophic interference*).

#### **Blending > Particulate > Recursive schemes.**

a. **Blending:** [A] + [B] => [C]. Info is lost.

b. **Particulate:** [A] + [B] => [A, B] Info is lost.

#### **(semantic network connectionism)**

c. **Recursive Network:** [A] + [B] + [C] => [A [ \_ B, C]] Info is preserved.

[B [A, \_ C]]

[C [A, B, \_]]

What we want out of a human-like cognitive process is to have a representational system that can operate over variables across two or more operating systems, run in parallel whereby such an overlap of **Dual** systems, along with their representations, creates a representational hierarchy. Only recursive systems can deliver these prerequisites: Only Recursion can allow its operations to spread across variables in this way. In recursive systems, items are not simply **nodes** based on frequency, but rather are such primitive (nodes) become **codes** whose computations themselves act as symbolic manipulations which can change their status as determined by their very arrangements (order). This is what we see below regarding the ‘boat-house’ example. If the scheme was, by innate definition, ‘flat’, then there could be no distinction between the two-word structure [boat-house] since in either ‘blending’ or ‘particulate’ schemes, no necessary word order would arise.

These three types of functions (*blending, particulate, recursive*) can be defined in terms of **Low vs. High-level of processing:**

**Low-level:**

- (i) **Blend** (combine): is seen in SRN, ML-P networks.
- (ii) **Particulate**: is seen in semantic networks.

**High-level:**

- (iii) **Recursive** (Treelet structure)

(Keep in mind that these low-to-high level distinctions on processing map onto neurological processes of the human brain/mind—what I have come to refer to as a **Dual Mechanism Model** of the brain/mind).

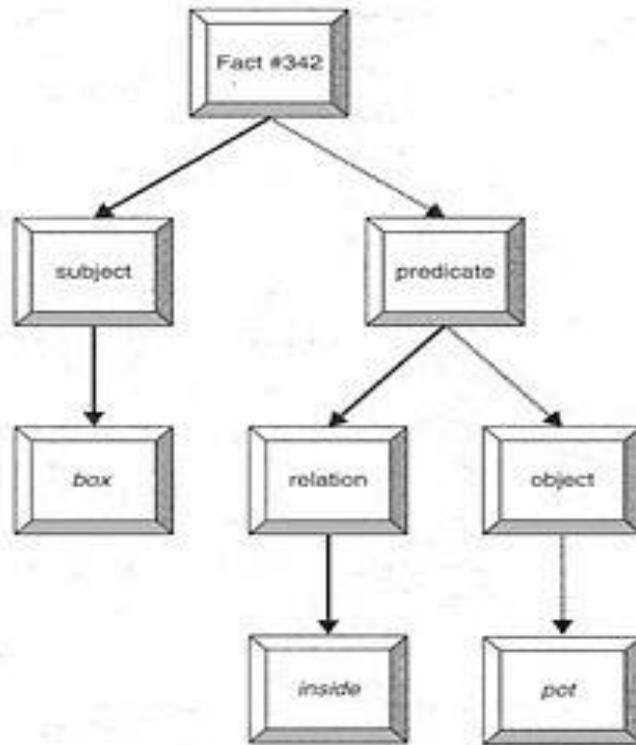
**Low level**

At these lower levels (*Blend, Particulate*), the analogy which works best here is that of ‘Cell transportation’ whereby neuro strength is activated and reinforced by neuro bundling found at local, adjacent levels. The region of the brain best suited for such functions is the **Temporal Lobe** (*insular cortex*) areas where ‘frequency of blend/combine’ works at the highest efficiency. (Note: In terms of language development, this is where we would find *lexical development* (a word-learning processes associated with Wernicke’s area located in the temporal lobe). At such low-level processing, frequency has a very strong role to play—the Hebbian expression holds here: ‘What fires together wires together’.

Low-level, in fact, can be assumed under connectionism/ML-Ps since local-firing of neurons trigger input-outputs as related to established **register-sets**.



Fig.2 Register-sets (Copied and reinserted here as found in Fig. 4 of Appendix 1).



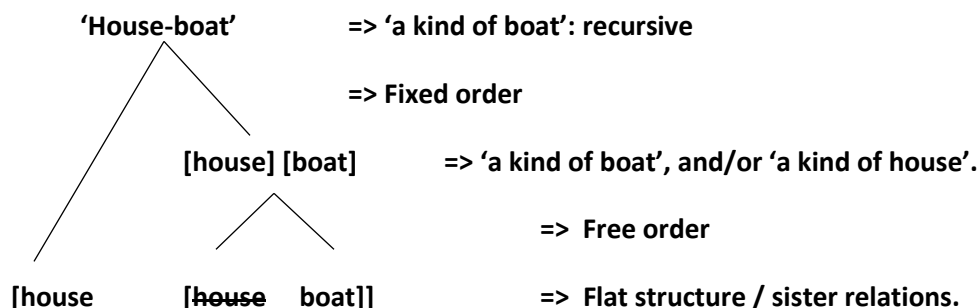
Within a treelet structure (above), consider the ‘register-sets’ [boxes marked *subject*, *predicate*, *relation*, etc) to be analogous to neurons (cells) as found in the brain. The Hebbian express ‘What fires together wires together’ accurately describes this local level of function as ‘frequency-effects’ plays a critical role at this level (just as with word learning)—any associative model that relies on strength of reinforcement must establish such local wiring & firing domains attributed to one-one associations (**behaviorism**). In this sense, keen connectionism is a new form of behaviorism. Notice how treelet structures allow nodes to be coded in ways which establish **categories**, such that (categorical) relations hold between e.g., **subject** and **predicate**. This is the main difference between a serial search which would be conducted exclusively via ‘nodes only’ versus searches which could be done in parallel over nodes and categories. Treelets resolve the problem that other low-level processing had (SRNs, ML-Ps) since treelets, with these new categorical relations between nodes can generalize to new concepts. Hence, **category**, **creativity**, and the ability to function with **variables** inters into the processing domain. Such ingredients would be critical for any true human-like language to be mimicked successfully in any AI machine. So, what AI needs is a program (a domain) that uses (abstract) *codes/symbols* and not just (associative-based) *nodes*. Recall, that in human language, much of our language intuition goes

against the evidence supported by our data—we move beyond data. This fact can only be handled by an Operation System which can process both/dual nodes and symbols simultaneously (leading to hypothesis calling for a Dual Mechanism Model (Marcus 2001, Pinker 1999, Clahsen 1999, Marcus et al. 1995).

Note how *symbolic* codes are reliant on blends, or semantic features (Particulate), but are rather fully **syntactic** in nature—hence, the form of the syntactic tree. For instance, with regards to a simple blend of the lexical items (*box, inside, pot*)—and with there being no syntactic subject-predicate relation— there would be no way to determine the relational predicative arrangements of items: for example, is ‘the box inside the pot’, or is ‘the pot inside the box’? Such <Subject <Predicate>> is crucial in order to determine meaning and simple *blend/merge sequences* do not provide such information. We’ll look at this below.

As another example, consider our ‘house-boat’ example below in terms of **blend** (particulate) vs. **recursive** formation:

## [2] Boat-house example



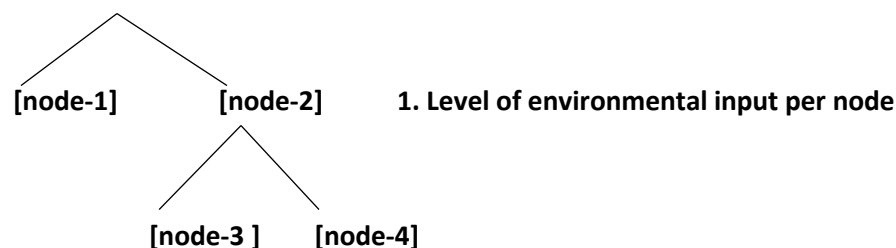
The **particulate** (*semantic*) units would contain the encyclopedic information having to do with the entries of <boat> & <house>. A **Blend** of the two items could yield either <house, boat> or <boat, house> without word order. It is only at a recursive level would we be able to interpret ‘house-boat’ ( a kind of boat) from ‘boat-house’ (a kind of house), etc.

The above ‘Boat house’ example can be projected onto a treelet scaffolding:

[3]

## Treelet structure

## 2. Level of relationship between nodes



[Nodes 1-4 are 'register sets']

Each node plays a neuro analogy to a cell (low-level, frequency driven). This process is analogous to what is referred to as **cell transport** and can easily be replicated by connectionist models such as SRN and ML-Ps. What the high-level processing allows is for relationships to form between such nodes, creating hierarchical arrangements. Let's assume Nodes 1-4 (= **register-sets**) to be analogous to neurons (cells) such that their formation (wiring) is determined by the strength of frequency (firing). The relation between input (environment) and the coding of the register-set has been referred to as a kind of **cell transport**. This cell transport takes place a low-level processing akin to associative mechanisms which drive any kind of behaviorist model. Of course, in order to capture true human thought, other higher-order processes must take place. These higher-order processes inter alia involve relationships between nodes in building a hierarchy (e.g., mother-daughter relations which break otherwise flat sister-sister relations). For example, the relationship between nodes 3, 4 are sister relations such that two words may be constructed to form a phrase {boat-house//house-boat} (showing no order). Once movement takes place, say, between node-3 and node-1 such rising has moved a sister item from out a flat structure into a recursive hierarchical structure (showing necessary syntax) e.g., [boat [~~boat~~-house]].

So with **blending** processing, the specific information of the item [A, B] is lost (as the info is averaged and/or combined): e.g., Blend [A] + [B] = [C]. Again, this is exactly what we find with simple recurrent networks. Regarding **particulate/semantic** networks, what we come up with is [A]+[B] = [A, B]. Recall, what we want out of a human-like operating system is a processing network which breaks symmetry in order to represent unambiguous orders. As presented herein, what we are claiming is that only a true recursive processing procedure can achieve this: e.g., [A] + [B] = [A [B, C]]. Returning to the 'box-pot' example above, what we need is a formation that delivers a recursive [box [inside pot]] and not a serial flat structure of [box inside pot] (the former being recursive, the latter blend/particulate). Marcus has a very clever way to describe the drawbacks found in **particulate/blending processing vs recursion**—he goes on to suggest that, e.g., the two items (colors) [black] & [white] are not somehow halfway between black and white (like some shade of gray) (Marcus 2001, p. 87). Rather, while it may be true that particulate

features must be self-preserving (information must not be lost  $[A] + [B] = [A, B]$ , etc.) there is still no way to represent unambiguous distinct relations between elements or semantic features.

**Problem:** ‘A box inside a pot’.

It would not be enough to simply activate the individual elements [+box], [+put], [+inside] since the same coding would trigger ‘a pot inside a box’. Hence, semantic features via particulate function cannot generate order. If *box* = A [1010], *put* = B [1100] and *inside* = C [0011], the coding of [A, B, C] even if info is preserved doesn’t render a hierarchical order. Again, what is needed is a recursive structure which yields [A [B, C]] or [box [inside pot]] (as seen in English SVO word order).

### High-level

The relation between register-sets doesn’t seem reliant on local/frequency-based computations, but rather a relation is formed via rule-based/symbolic manipulation. It is in fact the codes with symbol manipulation that leads to recursive treelet-structures—a treelet structure allows *search* to be performed in *parallel* (**parallel search**). The human **brain** (at low level processing) is enriched with the capacity to bootstrap itself up to higher-level processing thus allowing for the creation of the uniquely specific human **mind**. In the same way as the brain bootstraps neurons and creates symbol manipulation, so too do treelet-structures allow for nodes to be abstracted away in order to make room for abstract and symbolic *codes*—codes which can establish **categories**.

Note. Symbolic codes are not reliant on:

- (i) **Blends**  $[A] + [B] = [C]$  or,
- (ii) **Particulates** (semantic features)  $[A] + [B] = [AB]$ ,  
...but rather are syntactic in nature, yielding
- (iii) **Recursive** structures  $[A [B, C]]$ .

This is the main difference between the search of ‘nodes’ (a serial, low-level manipulation) versus the parallel search over codes and categories. Treelet-structures resolve the problem that other low-level processing had regarding SRNs and ML-Ps—treelets can generalize to new concepts. SRNs and ML-Ps have a very difficult time generalizing novel items or extending into realms of

new information. The most celebrating problems cited in the AI literature we find when we attempt to use common set of nodes (in either a semantic capacity or in a combined capacity) is that there is very often a **catastrophic interference** which follows.

**Catastrophic Interferences.** One such example (cf. the ‘Rumelhart-Todd’ network) showed how if a network was taught that <birds have wings and can fly>, and then taught that <a penguin could not fly>, it falsely assumed/learned that <the penguin must be a fish> (and as a learned subsequent of cascading features, assumed it also to have gills, scales, etc.). This is just one example of how semantics and blending may bleed into other unwanted interpretations. Of course, it is always possible to code such additional info into the network, but I think you can see how overwhelmed a model can get by what humans take for granted in terms of irregular exceptions (‘to the rule’), intuition about how the world works, common-sense, etc. The multitude dynamics that enter into any meager attempt to encode all such features into a SRN or ML-P (without recursive parallel search) very quickly outpaces the capacity for such rudimentary models.

### ***The Two-Horse Problem***

Consider the ‘Two-horses’ problem (Norman 1986 cited in Marcus 2001, p. 123), or another example referred to above as the ‘*Sandwich problem*’. The problem is to be able to handle different instances of the same concept, at the same time. For instance, let’s code <sandwich> as [01101], then if Helene eats a *sandwich* [01101] and John eats a *sandwich* [01101], the system has to somehow go beyond the simple one-one coding in order to treat these *sandwiches* as different. Specifically, the problem here is how can we get an operating system (OS) to handle both (i) general properties of *categories* (‘sandwich’) as well as (ii) properties of *individuals* (Helene’s sandwich). If coding denotes only general properties of objects/items, then we need some overlapping code to distinguish between *kind* vs. *token*.

The ‘two-horses’ problem is the same: in order to capture the ASCII code for HORSE as e.g., [100011], how do we distinguish e.g., ‘*John’s horse*’ from ‘*Mary’s horse*’—where the one-one code for HORSE has no way to handle the individual distinction (there are two individual horses involved). Formal semanticists using predicate calculus often denote representations of ‘individuals’ with lower case (<horse> = John’s horse) and representations of KIND as upper case <HORSE> (general property of HORSE). Returning to ‘the box inside the pot’ example above, there is something very similar to how an operating system (OS) might deal with this distinction—namely, we’ll have to devise a single processing domain to handle two fundamentally different procedures at the same time: one processing which handles the items and one which handles

individuals. Embedded structures seem to be the best bet to handle such a duality or processing. The expression ‘The box inside the pot’ has to distinguish somehow that ‘the box is inside the pot’ and that ‘the pot is NOT inside the box’. If simple one-one nodes only codes for items, then there is no way to tease out the distinction. But keep in mind if we can somehow show a coding system which delivers two distinct processes, then we might be in luck. Consider just how a recursive code (using diacritics) might be used:

- [4]      a. [box [inside pot]]                      b. John’s horse [horse [of John]]

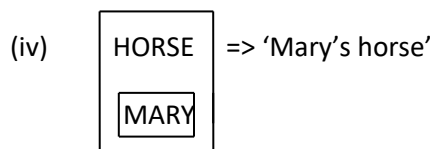
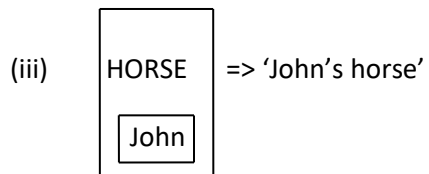


As represented by the ‘two-horse’ problem (as well as the ‘sandwich’ problem above, cited in Marcus 2001), both SRNs as well as ML-Ps—in fact much of what is behind current platforms which support AI today—have a very difficult time in handling both *individual* tracking (over time) as compared to *kind* representations (static). The ability to represent general, associative properties of an object/entity/event is one thing, but to represent individual over kind seems to place an overwhelming burden on the SO. Cognitive scientists and linguists use diacritic indexes (which are by their very nature recursive) in order to handle such dynamic information. Only via an implementation of a DMM where nodes and features (blending & particulate structures) can merge with recursive co-indexing would we have an OS worthy of capturing this full dynamic.

So, what might a DMM look like in capturing this twin processing? Let’s consider again the difference between the ‘kind’ [Horse] vs. the ‘individual’ [John’s horse]. The OS (operating system) of a dynamic DMM would look as follows (using indexes):

- [5]
- |      | <u>Item</u>                          | <u>Kind</u>              |
|------|--------------------------------------|--------------------------|
| (i)  | John’s horse = [horse <sub>j</sub> ] | => [HORSE <sub>j</sub> ] |
| (ii) | Mary’s horse = [horse <sub>m</sub> ] | => [HORSE <sub>m</sub> ] |
- (j = John, m= Mary).

Formal semanticists using predicate calculus often denote representation of *individual* with lower case, and representation of *kind* with upper case. Diacritic usage as a recursive means would look as follows:



Clearly, this is very similar to what we saw above showing how the 'flat recurrent' structures, typical of SRNs and/or ML-PS, have a difficult time handling such embedded dual processing: for instance, just as there is no way to capture 'boat-house' vs. 'house-boat' as discussed above by simply coding for <boat> and <house>, so too would there be no way to handle 'box inside pot' from 'pot inside box', or the kind HORSE vs the individual 'John's horse', etc. A flat structure would ambiguously code both instances as [box/pot inside box/pot] as well as not being able to capture the dynamics that 'John's horse' and 'Mary's horse' (or 'Mary's sandwich' for that matter), is not one in the same item.

The above discussion provides some details into why mere *particulate* argument structure (semantics) and/or simple statistical averages of *blends* cannot suffice to produce human-like learning model (as fully expressed within human language).

### Intermediate Summary on the Dual Mechanism Model (DMM).

The DMM proposal calls for a single domain which encodes for a dual processing whereby a single representational format operates over variables across two or more operating systems run in parallel within a single domain. Such an overlap of operating systems (within one domain) along with their (co-indexed) representations creates a unique ability to recognize and process representational hierarchy.

**The DMM:** Human-brain processing // AI Platform, processing

- (i) Low-level (local): Cell transport<sup>41</sup> Temporal Lobe, Insular Cortex, [+Frequency]  
// SRN, ML-P, 1. Blending:  $[A] + [B] = [C]$ .  
2. Particulate:  $[A], [B] = [A, B]$ .
- (ii) High-level (distant): Broca's region, displacement [-  
Frequency] (symbolic: syntax, music)  
// Rule-based, Hierarchical (diacritic, index)  
3. Recursive  $[A [B, C]]$ , or  $[A [A, B]]$ .

With a DMM, both systems can coexist in a single domain (i.e., the human brain/mind). This duality is what we find as the unique operating system behind human language.

**Intermediate Conclusion to remarks on 'Two-Horses' Problem as it has to do with a Single Mechanism Model (SMM) and/or Multi-layer Perceptions (ML-Ps) versus a Dual Mechanism Mode (DMM).**

The above problems point to processing of representation tasks which have to be performed in a multi-fold manner. If coding (as sequenced in a *single mechanism model* (SMM)) has no way to code for token over kind, or item (individual) over kind, then the operating system essentially becomes very impoverished in the kind of knowledge humans entertain. Of course, there may be ways to override this problem by coding for item and subsequent coding for type (two interdependent codes), but as it turns out, such overlapping coding places huge amounts of burden on processing: a problem that is not easily overcome by ML-Ps. But such processing is easily secured by children, even effortlessly done, by the age of three-years. Young children innately know how to identify individuals over kinds. (See Sorrentino 1998 for experiments such as the so-called 'Zavy' experiment presented in Marcus, 2001, p. 125). Experiments such as Zavy—where stuffed animals are shown to children first as 'kinds' and then as 'individuals', and then where they are tested on tracking one over the other—demonstrate that what poses a significant problem for SMM, ML-Ps, is easily performed by children as young as three years of age. (Such knowledge is often termed **Object Permanence**).

We'll take a look on how the '*kind versus individual*' distinction can easily be coded in recursive structures via diacritics. The claim for recursive/syntax (=DMM) over recurrent/semantic (=SMM) will be extended in the following sections where it will be argued that the distinction (along with its tracking, as demonstrated by the Zavy experiment) requires

---

<sup>41</sup> (Local) 'cells which fire together, wire together' (an adjacency condition) (See Hebb).



**two types of search capacities:** one search for the item itself (associative/recurrent), and another parallel search for how to determine how the items are manipulated as a category (intervention/recursive). So, what we want out of a well-functioning ML-P (if that is what is called for our AI operating system) is the ability to represent relations between local nodes and distant variables, and to be able to compute across the two representations.

### **ML-P intermediate summary**

In sum, ML-Ps have several difficulties, as listed below:

The model has a hard time—

- (i) presenting novel items, since the model can only blend/combine pre-existing representations previously encoded.
- (ii) computing across two different representations, such as nodes plus variables.
- (iii) using semantic features in a syntactic manner.

The limits of a SMM/ML-P vastly reduce how representations can be computed over time and across space. ML-Ps are prewired and can only be altered via so-called strength condition (*à la* behaviorism) of frequency. This is not ‘learning’ (learning is not of a sole, brute-force memorization). What we want of a true AI operating system that comes closest in mimicking human learning/thought is a system of input/units that are not prewired as determined by strength alone, but rather that their unit-values are computed freely over competing pathways—pathways, possibly cascading in parallel over a dynamic and hierarchical domain. This uncanny ability to track similar and/or competing inputs across two parallel operating systems is what is called upon for learning. Only a DMM run in parallel could perform this feat. A SMM would not only be too slow, but also would lack the architecture required for such true learning.

### **Poverty of stimulus.**

In addition to arguments that have been laid out in our 4-sentences section, consider that claims made by the linguists Peter Gordon (1985) that young children know not to keep plurals embedded within compounds—what do you call a person who eats rats? Children respond ‘rat-eater’ (they delete the {s}) and they never respond \*rats-eater. Gordon suggests that children innately know that inflectional morphology {s} can’t be kept embedded within a compound, even though they have never been explicitly shown that such data is in violation of some English grammar. The mere fact that they never hear it (because it is, in fact, ungrammatical) doesn’t

explain why children never entertain the prospect: children say loads of erroneous things that they have never heard before. Hence, even though children have no empirical evidence (negative stimulus) that such constructs are wrong, they still shy away from compound-embedded plurals. This is what is referred to as the ‘poverty of stimulus’—namely, when children’s inferences go beyond the data they receive.

Gordon suggest in this sense that there must be some innate built-in machinery constraining child learning of language. So, to put this into our discussion of ML-Ps, if input-to-outputs models are the square product of environmental learning, one question that will come up is: How does such learning deliver a result such as found with the poverty of stimulus case? Perhaps symbol manipulation of rules will be required in some fashion after-all. But, if so, perhaps we need to rethink the brain as a mere neuro/digital net. In fact, the analogy of the brain as a digital computer had been under attack for some time—as Gary Marcus (2001) claims (we still know very little about how the brain works at the higher level. Perhaps at the lower level the brain-to-computer analogy holds (where local firing of neurons takes place, etc.) but with higher functions, when we talk of a ‘mind over brain’ of how a brain bootstraps a mind) there may need to be a fundamentally different processing with an entirely different underwriting.

### **Michael Tomasello vs. Ken Wexler** (See ‘Lenneberg’s Dream’, Wexler 2003)

Perhaps one of the more passionate debates which have arisen regarding how this distinction of ‘recurrent/association’ vs. ‘recursive/rule-based symbolism’ can be played-out in developmental child syntax is that (naïve) notion that the errors found in early child syntax is simply the product of a lack of memory, or a ‘bottle-neck’ of cognitive-processing sorts. Let’s just take a last moment to flesh this hypothesis out.

There is a case to be made that if, for example, a child says ‘He open it (where agreement {s} is missing (e.g., He opens it)), one could claim that such an utterance is in fact available in the input via the utterance ‘[Should [he open it]]’, etc. Likewise, ‘She eat grapes’ could come from available positive evidence of ‘Does she eat grapes? etc. where if the initial finite verb is dropped (forgotten) of the matrix main clause, the remaining structure would be consistent of what young children say. This theory relies on a partial **mimicking theory** of an X=x type, where the initial mimic has been lost. Tomasello argues for such a theory of child language which is completely based on associative mimicking (and/or the lack thereof of certain parts of the base mimic). This is well and good!...

But problems quickly emerge from such a naïve theory. For one thing, if a partial mimicking theory is correct, young children (two years of age) would also say things like ‘Did you want’? as derived from the partial mimic of [what [did you want]]? However, they don’t say such

utterances. Rather, a more typical stage-1 expression (years and younger) might be ‘What \_ you want?’ where the second word (auxiliary verb) is missing from the string. Children at stage-1 don’t produce such Aux-first words with Wh-words missing. They rather deleted Aux and maintain the fronted Wh-word. The fact that positional errors can no longer be part of the theory suggests that the child is operating under a structure-dependent hypothesis of language, and not a positional-dependent (or structure independent) hypothesis.

Other even more interesting examples include why a child might say ‘Him do it’. Tomasello might argue that they hear [I saw [him do it]], and so on. But still, even if children process in this way, we would have to account for why the first part of the sentence is ignored given that in frequency, first-parts are most marked. This would seem to strike at the heart of Tomasello’s theory. Note that such a theory of mimicking is completely reliant on Frequency. (But the highest frequency word in English is the word ‘The’. But the word ‘the’ is notoriously the last word acquired by a young child (e.g, ‘The car is broken’ => ‘Car broken’)). ‘Him do it’ as shown above are so-called **Small Clauses**. While it may be true that such small clauses are abundant in the data, such a theory would also predict the following: [Mary knows [I like candy]], and so the prediction would be that the child might drop the initial ‘Mary knows’-clause and say [I like candy] with nominative case ‘I’ (the adult utterance and the small clause are homogeneous)<sup>42</sup>. However, the child does nothing of the sort. The child strictly says ‘Me like candy’, etc. Such utterances pose a problem for any naïve theory based on mimicking or forgetfulness of mimic. In this case, children could be said to go beyond their data, they go beyond frequency of input. Their stage-1 utterances are rather the result of a lack of structure, and not the lack of positional memory. Stage-1 child grammars are systematic, rule-based (or the lack thereof), are constituency-sensitive and based on syntactic properties (provided by UG).

In sum, returning to our AI-discussion, any ML-P system which hopes to learn human language would have to be sensitive to such a processing of learning. The system must be symbolic and rule-based and must be able to go beyond mere +Frequency/mimicking of structure.

ML-Ps, as prewired, can alter settings via strength—but this is not learning. Again, what we want of a true-learning algorithm is that units/inputs are NOT prewired as determined by strength alone, but that their values can be accessed over competing pathways, cascading in parallel over a dynamic hierarchy. Language learning/acquisition is the ability to track similar and/or competing inputs across two or more operating systems. Such ‘checking-off’ of a retrieval

---

<sup>42</sup> There is a clear relationship (syntactic) between [+Nom] nominative case and [+Fin] finite verbs.

of input between the two or more pathways (operating systems (OS)) must work in parallel since a serial OS between the two would be too slow. Recall, this is the minimum of what we want...to be able to handle human-like processing between \_\_\_ks as found in [1] above.

### Brain-mind bootstrapping

The idea that a *brain can bootstrap a mind* may have its origins in theoretical linguistics, particularly looking at child language development of syntax. One very promising model which has implications to AI and the cognitive sciences is the notion that these low-level brain processes which map local configurations on a frequency-based threshold may be only one part of the brain's processing of language (a more primitive part which is responsible for lexical look-up, retrieval mechanisms dealing with objects, items, etc.) while a second more abstract mode of processing takes such general properties of items and spreads them over categories whereby recursive operations may allow diacritics and indexes to work as variables. (Steven Pinker's 1999 book entitled *Words & Rules* captures this dual mechanism model distinctly).

By extension, much of what I am on about here in this section speaks to the notion that within a single domain of processing, a dual operating system may be in use which allows for this *individual* versus *kind* distinction. Minds can nicely grapple with categories and recursive structures which can handle the tracking of *individuals* (where indexes, diacritic variables are in operation) while the brute-force calculating brain serves one-one properties of *kinds*. (Recall in 'Against ML-P' Claim #1 above the use of diacritics to help resolve the so-called 'two-horses (sandwich) problem').

### The 'Brain-to-Computer' Analogy: 'Low vs High' levels as a linguistic function.

**Low-levels.** At the lowest levels of the hierarchical brain spectrum are found neuro connectionist systems which rely on approximation to 'local-dependency'—these are so-called finite-state grammars of the SRN type discussed below (G-1). (Included in such OS's would be so-called 'multilayer perceptron'). At these low levels, statistical regularities work in local configurations so that they bundle together. In linguistics terms, these low-level grammars create so-called **Merge** properties:  $\{X, Y\} = \{XP \{X, Y\}\}$  (where X is Head and Y is complement) as when two word/items merge in becoming a phrase. Also linguistic compounding can be said to be a result of such merge. For a slightly more sophisticated example, consider **linguistics Root-compounds**: e.g., '*chain-smoker*' where only the two items merge  $\{chain\} \{smoker\}$  and where no movement is required. (Notice a move-based product becomes ungrammatical, one can't derive the compound as *\*A smoker of chains*). It's only at a higher-level of processing where two merged

items become more than the sum total of their parts. Consider what happens to the seemingly similar linguistic compound ‘*cigarette-smoker*’ where one can say ‘*A smoker of cigarettes*’. This is such an example of a higher-level processing (albeit linguistic) which shows how the two merged items retain their specific past memories over two or more computational time-steps (CTS).

Consider the two distinct processes below (starting with low-level/Merge to high-level/Move:

[6] (1) Local:  $[X, Y] \Rightarrow \{XP\}$ :  $[Chain] + [smoker] = [chain-smoker]$

(2) Distant  $[X, [X, Y]] \Rightarrow \{XP\}_t$  (where t is trace of prior memory of structure)

$[Cigarette] + [smoker] = [cigarette-[smoker \text{ of cigarettes}]]$

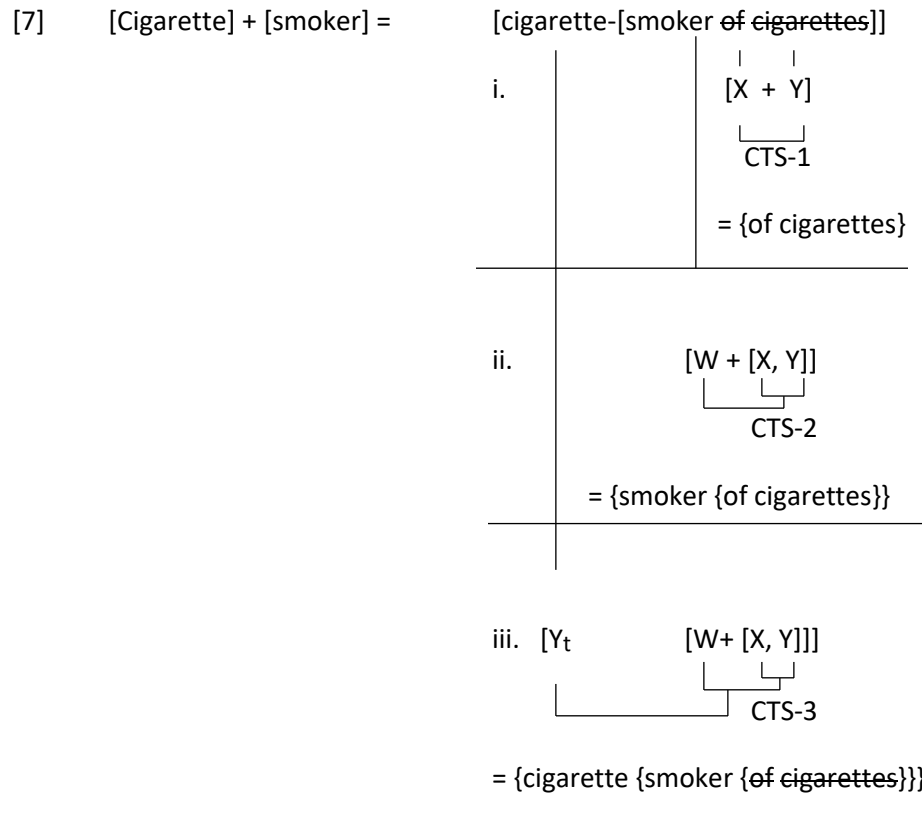
In [6,1], a local neuro function, say within one CTS (CTS-1), shows the memory limit in how two adjacent inputs (and *adjacency* does seem to be a prerequisite, common to how neuro firings work in adjacent bundling) combine to achieve an averaged-weighted product. However, if we were to apply the same ‘local neuro-firing’ to the syntactic compound found in [6,2] above, the interpretive distinctions between root vs synthetic compounds would be lost. Consider (1-2) restated in (3-4) below showing CTS numerations:

### Root Compounds (RC) vs. Syntactic Compounds (SC) as an analogy to computational time-steps (CTS)

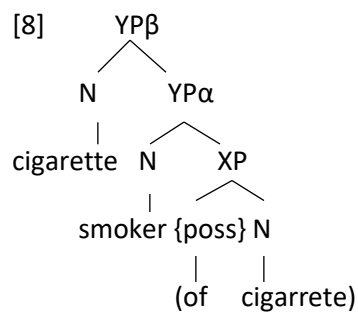
(3) Local:  $[X, Y] \Rightarrow \{XP\}$ :  $[Chain] + [smoker] = [chain-smoker]$

$$\begin{array}{ccc} [X] & + & [Y] \\ \hline & & \end{array} \Rightarrow \text{(memory between one CTS)}$$
  
 CTS-1

CTS-1  $[X= chain], [Y= smoker] \Rightarrow$  local firing of two units as found in a multilayer recurrent system.



Showing a linguistics syntactic tree, the CST-1 found in [7i] would be represented in [8] below



The above notion of ‘locality vs. distance’ as bound by computational time-steps (CTS) has antecedence to levels of computational processing found in the brain—with ‘Low-level’ computations being assigned to exact one-to-one neuro firing (say, having to do with the triggering of a specific node within a connectionists model), while ‘High-Level’ processing establishes distant relationships, say, between nodes. The expression that the brain bootstraps

itself in creating a mind can be played out in such a dualist scenario of local vs. distant neuron triggering (with the brain being pegged to locality conditions (the ‘associative brain’/Temporal lobe region) and the mind to non-local freedom (the symbolic ‘rule-based’ brain/Broca’s region). (See ‘treelet’ structure below for further discussion).

**High-levels.** At the higher levels, statistical regularities seem not to be dependent on local constraints, as shown in [7] above. Hence, the syntactic/semantic interpretational distinctions found between the above ‘*root vs syntactic*’ compounds could be drawn as analogous to ‘local vs distant’ neuro/unit firing (where unit would be labeled here as **word** (*cigarette*) and **grammatical feature** (*possessive*). This same dual distinction is also very nicely seen within linguistic rules (so-called regular-rules/distant versus irregular-rules/local).

For instance, notice how sound patterns of **irregulars** work on a low-level where frequency-effects of ‘bundling of feature’ can impact either neuro or linguistic processing: e.g. [ \_ [ing]] > [ \_ [ang]] > [ \_ [ung]] which generates *sing>sang>sung* may over-generate (over-trigger) based on a sound-frequency effect to *bring>brang>brung*, (but not *\*bling>blang>blung*). Notice how such novel words (so-called made-up ‘nonce’ words used for experiments) generate the distant rule—e.g., *Today I bling it, yesterday I blinged it*.

Such distant true rules never become dependent on local frequency-effects (or local neuro-firings), with true rules projecting over a variable/category such as instruction: <do x to category Noun>, <do y to category Verb>: add {s} to N when plural *one wug > two wugs* (see Berko), or, as just demonstrated above, but with an altered final consonant from /g/ to /t/ e.g., *today I blink*, and *yesterday I blinked*. True rule-formations of [N+s], [V+ed] work independently of frequency-bundling and their productivity allows new and novel items to be freely expressed as categorical variables across data spreads. Such a multilayer-perceptron model would have difficulty showing such a dual-level processing since perceptron models (SRN’s, CRN’s) would only code for a single mechanism model (SMM)—viz., the same mechanisms would have to be involved between RC’s and SC’s, thus losing the distinction.

In other words, regarding CTS’s, ‘cigarette-smoker’ would be forced into a local neuro-firing between two local units as expressed in [9] below:

[9] [cigarette] [smoker]

$$\begin{array}{ccc} [X] & + & [Y] \\ \hline & & \text{CTS-1} \end{array} \Rightarrow \text{(memory only between one CTS)}$$

For example, multilayer-perceptrons can only average (approximate) a broad range of functions, based on local distributions of a single mechanism model (SMM). It has been found (Marcus, Brinkmann, Clahsen, Wiese & Pinker, 1995; (Hadley, 2000) that these SMM's cannot capture such a class of operations which spread of two or more CTS's and/or that follow from a recursive, embedded coding: (noticing how the progressive structure in [7i,ii, iii] require embedded clusters (i.e., **recursive nesting**)). (See Pinker 1984 for initial reports of grammar modeling, and Pinker 1999 for review of a dual mechanism model, Galasso 2016 for 'First Merge, then Move' model in early child syntax).

The conclusions reached here are that multilayer-perceptrons cannot generalize from a limited data the same way as humans do.



## Thought-experiment on recurrent vs. recursive implementation in AI

<Insert>

[0] **Code snippets** (taken from Fitch 2010. p. 77).

(1) define function  $[A^n B^n](n)$  :

if  $n$  is 1, then return “AB”

else return (“A” +  $[A^n B^n](n-1)$  + “B” ; //recursive call

$\Rightarrow$  This grammar generates a recursive  $[A [AB] B]$ , structure-preserving embeddedness.

(2) define function  $[A^n B^n](n)$ :

interger counter  $i$

A\_section = “A”;

B\_section = “B”;

If  $n > 1$  then {

for ( $i=2$ ) to ( $i=n$ )

A\_section = A\_section + “A”;

B-section = B\_section + “B”;

end

}

Return A\_section + B\_section

$\Rightarrow$  This grammar generates a flat-recurrent/non-recursive  $[AB]$ , a simple recurrent network.

### AI-Statements (*code snippet functions*):

[1] True AI-Recursive (AI-R) (strong generating) grammars project at least a dual tracking of constituencies:

a). AI-R codes AB-strings in the middle of other AB-strings (center-embedded): [A [AB]B]. Such a recursive rule has the unique property of self-embedding.

b). AI-R is structure-preserving and can span memory across two or more phrases. (A circuit which remembers past structures only for a single computational time-step is not capable of representing complex structures. Memory must be able to span across multiple time-steps, while, at the same time, being able to look into multiple center-embedded structures).

[2] Center-embedded structure can affect left/right peripheral strings and vice versa.

e.g, [A [AB [AB]]] :

Recursive = structure-preserving loops, and not just 1-way feedback loops.

[3] ‘The ability to retrace a stack of function calls [*from peripheral to embedded strings*] must be specifically designed into the programming language and hardware’ (Fitch 2010, p. 77, *italics* belong to JG). Any recursive grammar which cannot maintain statements 1 & 2 are not full, true AI-R grammars and may be more **recurrent** in nature. This ability to retrace (track) also includes not only strings but variables associated with strings—some additional memory mechanism must be designed into the system to keep track of differences between variables<sup>(nm)</sup>—e.g., [A<sup>n</sup>B<sup>m</sup>]-grammars where the variable n= “repeat number of string” and the variable m= “alternate string within embedded structure”. The fact that a single function must be able to count and compare requires feedback loops which may arise from one structure and fall into another, whereby an embedded unit/circuit-(<sup>1</sup>) may affect a string unit/circuit-(<sup>2</sup>) found in the left/right periphery.

[4] The code itself must contain representation of structure—codes, and tracking of codes must be structure-dependent (like natural language where syntactic tree diagrams contain representation of structure). A nice analogy to the tree would be that words/strings Nouns, Verbs make-up a beads-on-a-string grammar (a *weak-generation* found e.g., in non-recursive flat grammars [AB]), while the phrase NP/(DP), VP/(TP) would make-up representation of structure (a *strong-generation*). The embedded nature between the two (weak vs strong) is both implicit and explicit in the design.

[5] So, while we read through this brief note on recursive AI-grammars, let's keep in mind that what we want from a well-advanced AI-operating system—a system which could come closest in dealing with what we know of human-behavioral processing—is a grammar-system which not only looks into its 'left/right-peripheral' adjacent AB-strings [AB] (so-called *weak* generation), but also has the ability to look into its 'center-embedded' [A [AB] B]-structure (so-called *strong* generation). In the latter case, the grammar calls itself. This is found in the defined function of the snippet code found in [0] above.

## [6] Introduction

Much of current AI, as I understand it, either uses as an operating system (OS) (i) a simple recurrent network (SRN) (as much discussed in Jeff Elman's early work (cf), or (ii) a complex recurrent network (CRN)—the former which is more or less an approximate calculation of the combined net-value reached of two inputs, (with or without architectural hidden levels), and the latter which uses not only feedback loops at the 'on-time' computational time-step (CTS)<sup>43</sup>— (so-called time-1 'hidden structures'), but also 'backward-looking' (time-2) interactions across at least one previous CTS. In either case, both (non-recursive) SRN & CRN AI-platforms are of a 'brute-force' nature which delivers an on-time catastrophic decision based upon probabilistic on-time calculi. Let's consider how the two aforementioned operating systems might differ from a truly recursive operating system in regards to what we might hope to build of an AI machine worthy of simulating true human behaviors (e.g., autonomous vehicles, machine language learning and language translation, decision-making tasks based upon CRN-encyclopedic knowledge, etc.).

Let's first consider an  $[A^n B^n]$ -grammar which derives e.g.,  $A^n B^n$  (4) as AAAA, BBBB (as discussed in Fitch 2010). This OS is what is referred to as a recurrent grammar whereby the *stacking* of two items (which could go on indefinitely) works in a flat non-recursive manner. This is a non-recursive structure and would be similar to any SRN: the crucial note here is that the variables within  $AB \{^n, ^n\}$  would not need to be tracked since they are identical. Now consider a more complex system (CRN) with memory of backward feedback-loop connections (between stem-nodes and their respective variables) which may compare inputs and outputs over the span of maximally one CTS. This could potentially give us an  $[A^n B^m]$ -grammar where variables:

---

<sup>43</sup> The time-step is the incremental change in time for which the governing equations are being solved.

(n= “repeat” form a list of what you know),

(m= “alternate” from a list of what you know)

yields  $A^n B^m(4)$  as AAAA, AB A BA BAB\*.

\*But the problem here is that from this point in the memory, a backwards look-up device would have to store in memory two simultaneous non-congruent inputs (and both inputs would be structure dependent)—namely,

[7] (i) that [AB...] is the product of the sequence, and

(ii) that [BA...] is not the product of the sequence (since ‘BA’ is the result of [B [AB]]).

In other words, there would have to be some additional memory mechanism (of variables) embedded within memory of nodes, viz., [memory<sup>2</sup> [memory<sup>1</sup>]] or, say, a [declarative [procedural]] interchange held across at least two or more phases.

It is my current understanding that what we are usually implementing in all AI/OSs across the board these days is mostly based on either a upgraded SRN or a CRN OS, but that there are few if any currently implemented AI-OS’s that can:

[8] (i) maintain a sufficiently dual-memory embeddedness (DME<sup>2</sup>),

(ii) hold and compare memory inputs from prior CSTs across at least two or more phrases, and perhaps most crucially, and

(iii) whose code itself contains a representation of structure<sup>44</sup>.

Why are these three properties found in [8] so important? Well, it may have something to do with how the unique human brain/minds works, how *intuition* can ultimately be gained, and how humans over our evolutionary-time span have bootstrapped a simple

---

<sup>44</sup> Note that our hypothesized DME<sup>2</sup> may lead to the two types of human knowledge systems, *declarative* and *procedural* knowledge.

neuro-network *brain* into a ‘much-more-than-the-some-of-its-parts’ complexity of theory of *mind*.

Let’s flesh out what (DME<sup>2</sup>) might look like via the following thought-experiment (having to do with, say, an AI-encyclopedic OS): The task is: Who is the first president of the United States?

**A ‘look-inside-two-boxes’ scenario:**

[9] AI-OS: so, here’s the (very tentatively) imagined system:

(i) Let’s say the *left/right periphery* [x, y] of [x [a,b,c] y] codes for overt *declarative processing* of the kind associated with questions and responses of a certain task. Only this overt/declarative operation has an interface with the outside world and is reinforced by the input received (self-learning). So, for example, when the (Q)uestion is asked, it processes the language of the question and maps the Q onto a relevant (R)esponse. The periphery [x, y] however is able to look into the covert center-embedded [a,b,c]-grammar as its *procedural-knowledge* source, with both peripheral and center-embedded structures allowing for ‘feedback loops’ for reinforcement of net-values, adjustments, etc.

(ii) Let’s say the *center-embedded* [a,b,c] codes for covert *procedural processing* of the kind associated with the mapping of an internal look-up (“a list”) which then allows retrieval of the item to surface to the declarative mode of processing (with the interface): [declarative [procedural] declarative]

[10] (Q)uestion put to our AI-OS : Who is the first president of the United States?

So, there are two boxes:

box-1 = three (P)eople (P1,2,3) (box-1 = time-step-1 of input for box-2)<sup>45</sup>

box-2 = AI (with DME<sup>2</sup>-recursion)

---

<sup>45</sup> Recall, that for box-1, this constitutes (G)eneration-1 of input—virgin input which would be used for the first time to deliver an approximate result. Of course, over many thousands of generations, any input/answer may change over time until it reaches stabilization. Usually, such G-1 (as a starting point) is either soft-ware/ programmed into the OS (top-down) as initial default settings, or that some predesigned architectural template is built-in (innately) to capture tailored types of weights and distributions of the specified incoming-data stream.

Box-1: P1 and P2 think they know the answer to Q, but P3 is unsure. As each of the three people in box-1 (outside the computer, say a room) enters into box-2 (the computer) to deliver his input/answer, a scenario unfolds: the third person (P3) is unsure of the answer and so waits until P2 exits box-2 and returns to box-1 (imagine P2 and P3 bump into one another, and P3 whispers to ask what the (R)esponse is to Q: P2 says **incorrectly** (as P2's grammar is erroneously operating under an 'alternating response' procedural grammar) that R=John Adams (erroneously generated from an mistaken  $A^n B^m$  grammar):

= Center-embedded code [A **[AB]** B] (procedural processing, "a list")

[A [(George Washington), John Adams<sup>1</sup> [George Washington<sup>2</sup>, John Adams]]] B]

P3 now enters into Box-2 to deliver his R-input 'John Adams'. At this time-step, the AI (box-2) finds the net-value R to Q (2 over 1 out-weight responses) and moves on. So when P3 returns to ask google (AI) a follow-up Q 'Who the first president of the United States?' the Response from a 2-1 weighted input-to-output system is 'John Adams' and so P3 feels reassured of the 'correct' response and assumed 'knowledge' of Q.

[11] Now the question is precisely: What is wrong with all of this? For the computer, there can be no knowledge outside of box-2, (outside of its own OS) and so the matter is moot. All that an AI-OS can do is approximate the values of multiple inputs ( $n=three$  at this step) to average the answer (and there can be no looking inside another box for any additional reference, e.g., another encyclopedic entry which may list all the presidents in order (say, an [AB]-grammar generating a list which can't be item-based manipulated upon and which is quite distinct from the task at hand). And so the matter comes to a close.

But for P1,2,3, when the three learn what AI 'thinks' is the right answer they become confused: namely, the group's *intuition* begins to determine that *something has gone wrong*. But for the computer, nothing could have gone wrong (outside of the input)—there is nothing that the AI system could learn from such a scenario without the capability to look inside of an embedded structure. (Of course, the more input (correct) the system receives over time (over time-steps), the *noise* of incorrect input will become increasingly less and less significant (as its input-to-output weight (frequency) will gradually dissipate over time). This is *True!* But the question of how the system goes about the *learning, de-*

*learning and adjusting* is of interest to the computational theorist, and this poses an interesting challenge for AI.

What we want from a fully operational (recursive) self-learning AI machine is the **processing-capacity** to “learn” not only from *mistakes* within its own box-2, but also to learn from a *memory* that allows the comparing of different variables across two or more different phrases (phrases = boxes). Sure, no one would expect the AI system to have to deal with the (human) “intuition factor” as presented by the three people as they discern the wrong answer (the group intuition). But surely, an AI system would want to be able to trace, and cross reference competing models and structures (declarative/procedural) as might be associated with the input of a singular question. In other words, we want AI to understand how the slip was made by P2 (but not by P3).

[12.1] Correct: Let’s consider how a “center-embedded” structure would look like for the R given by P1:

[A: who is the first US Pres? [AB list of all US presidents in order] B: George Washington]

[A: Question [AB<sup>n</sup> = George Washington, John Adams, Thomas Jefferson.....etc.] B: Response]

(where variable {n} = repeat list as you know it)

[12.2] Incorrect: Let’s consider how a “center-embedded” structure would look like for the R given by P2:

[A: who is the first US Pres? [AB list of all US presidents in order] B: John Adams]

[A: Question [AB<sup>m</sup> = John Adams, George Washington, \_\_\_Thomas Jefferson.etc.] B: Response]

(where variable {m} = alternate list as you know it)

Where AB<sup>m</sup> = [John Adams, George Washington, ~~John Adams~~, Thomas Jefferson.....etc.]



[13] Recall, what we want from a self-learning AI-OS ability is two-fold:

- (i) To identify that the nature of the error is actually encoded in the center-embedded structure—to realize that such encoding can actually be examined by the periphery [a,b,c] in determining the nature of P2's error (that  $[AB^m]$ -procedural grammar was wrongly put in operation rather than the correct  $[AB^n]$ -grammar).
- (ii) To assume the correct hypothesis despite the counter input given to the peripheral nodes [x, y].

[14] **Note for such embedded recursive structures:**

- Grammar:  $S \Rightarrow A S B$  (recursion) where a phrase structure rule contains S center embedded.
- S has the ability to look inside an AB function  $[A^1 [AB^2] B]$ —both  $AB^{1,2}$  is preserved in memory and the structure can be recalled to be examined.

Let's imagine as the strongest approximate to (DME<sup>2</sup>) a single non-squared (DME) with CST (1-4) (the first 4 time steps in a derivation). So, the combined net value of (1-4) get read at 4 as an approximate of steps 1-3. In this case there is no need for backwards integration, only a look-ahead value is given for the combined approximate net value. But the problem here is that the earlier values of 1-3 have been lost. What (DME<sup>2</sup>) allows us to have is a memory of past input-output products to hold and compare across time. Allowing for such embedded memory takes into consideration not only the net value but also how the value was reached and may have evolved over a span of several CSTs.

For instance, imagine that a combined net value (output) is reached and read as 'John Adams' to the question (input) Who is the first president of the United States? (Reached by our G-1 AI). And of course, as AI gathers more and more input to stock-pile its database, (i.e., more CSTs have been accumulated over time) the best approximate answer becomes George Washington (self-correcting learning, so to speak). At this point the old/wrong answer of 'John Adams' is lost. But why should we care, we have reached the 'correct answer' (and John Adams was a fluke, certainly not of a significant ratio).

But this hypothetical question strikes at the heart of theory: In theory, if input and outputs are, and can only be catastrophic with no ability for *graceful degradation* (not in terms of hardware but in terms of software), then the OS is not able to preserve its past



structure over time (and memory is lost). In fact, the most oft-remark assumption made about both SRNs and CRNs is their inability for self-preserving of previous structure...

[15] So, the upshot of this exercise is that we want to be sure—in whatever AI-OS we are dealing with which attempts to mock true human learning/behavior (and not just to gleam encyclopedic knowledge, since all that that entails is a feed-back looping CRN)—is that the statements found above are considered when coding for true recursive tasks.

We want an AI-OS to always get the answer right to the question Who is the first president of the united states? irrespective of the two diverging input-responses from P1 and P2: (P3's "intuition" is outside of the AI-OS, (perhaps for now—something the cognitive sciences will be grappling with for some time to come)). In other words, what we expect a fully recursive (DME<sup>2</sup>) operating system to do is not simply *count* frequency and distributional net-values within its matrix processing layer, say the peripheral [x, y] connectionist layer, but also to be able to dip into non-matrix connection-layers in reaching a (corrected) net-value result. Hence, oft-spoken hidden units can't just be a matrix orientated, for its own nodes, but must be able to cross-reference connectionist pathways and routes from other structure dependent layers, making AI truly recursive and structure dependent.

This last point is exactly what we find for human/nature language. Consider the embedded sequence below:

[16] [John knows that [Mary doesn't know that [Bill knows the truth about what happened]]]

For such positive to negative embedded strings/clauses, we need to keep in memory 'who' is in the 'know' and 'who' is 'not'...and this memory spans over the length of the sentence. Sure, very long embedded strings may place heavy burdens on memory, but it's only via recursive structures that allow such processing to take place—certainly, a processing of mere adjacency-based input factors could not perform the task. Consider one last example (\*marks for ungrammatical processing):

[17] \*[The boy Bill asked to meet Mary thinks he is clever] (Bickerton 2010: p. 202).

If the reading of mere flat/adjacent nodes [a, b, c,] was all that was needed, then 'Mary' would most certainly be primed as the subject of 'thinks' (based on the fact the 'Mary' sits leftward via a positional node of right-positioned finite verb thinks'). Of course this is

wrong. Not unless the parsing mechanism is allowed to dip into non-matrix embedding clauses would the processing be validated:

(1) [The boy<sub>i</sub> [Bill asked to meet Mary] thinks<sub>j</sub> he is clever]

(2) [The boy<sub>i</sub> [Bill asked ~~the boy~~<sub>i</sub> to meet Mary] thinks<sub>j</sub> he is clever]

⇒ The boy thinks he is clever.

⇒ Bill asked the boy to meet Mary.

⇒ (\*Mary thinks he is clever).

## Conclusion

Coupling these observations made above with what we know of the ‘recurrent vs. recursive’ distinction, it becomes clear that what we want out of a fully human-like operating system (OS) is a recursive structure which can allow its operations to spread across variables. Items of the  $\{X\}=\{X\}$  sort, reduced to simple nodes and based on frequency do not suffice. Rather, what we want out of an OS is an asymmetrical language  $\{x\{y, z\}\}$  whose very computations themselves act as symbolic manipulators, which can change their status as determined by their arrangements (what was seen by our example of [house boat]//[boat house]). If the OS scheme was by innate design ‘flat’, then there could be no distinction between the two structures, since in either a ‘blending’, or ‘particulate’ algorithm, no necessary word order would arise. It was at this point that Marcus’ treelet-structure was advanced in order to create a hierarchical/representational structure.

It was shown, as presented by the ‘two-horse’ problem, and ‘sandwich’ problem (cited in Marcus 2001), that both SRNs as well as ML-Ps—in fact much of what is currently behind platforms which support today’s state-of-the-art AI—have a very difficult time in handling both individual tracking (over time) as compared to ‘Kind’ representations (static). The ability to represent general, associative properties of an object/entity/event/ is one thing, but to represent ‘individuals over kinds’ seems to place a too overwhelming burden on the OS. It seems very likely that only a Dual System (DMM), one which can represent ‘general look-up’ properties with brute-force statistics as well as a second system embedded within the same domain, which can track individual properties over kinds, can fully capture human-like processing. Thus, the human OS is necessarily dual-like in structure, recursively so—where, at times, frequency of data is overridden by sheer categorization.

Finally, we reach the conclusion that only a Dual Mechanism Model which encodes both for *recurrent* as well as *recursive* means could possibly serve as an approximate to AI. The DMM

proposal calls for a single domain which codes for a dual-capacity processing, whereby a single representational format operates over variables across two or more operating systems run in parallel within the same domain. Such an overlap of systems, along with their representations, creates the unique ability to recognize and processes representational hierarchy, the one essential ingredient necessary for human thought.

## Note 5

---

### A Note on the Brain: Broca, FOXP2, and the role of MOVE.

#### A neuroimaging review

Classic neurological accounts of the ‘brain-to-language’ corollary often assume two discrete and lateral regions of the cortex: **Broca’s** area (inferior frontal gyrus/left frontal lobe) and **Wernicke’s** area (superior temporal gyrus/left temporal lobe). Much of this popular demarcation has its origins in 19<sup>th</sup> century studies which dealt in observed brain lesions in post-mortem cases—where identification of bundles of lesions found in specific cortical regions of the brain corresponded to the observed language difficulty suffered by the subject when alive (via., Broca’s aphasia, Wernicke’s aphasia). However, at that time, very little was understood about the normal functioning brain and its processing of language in real time.

In the 1990s, what has been called the ‘decade of the brain’, neurologists teamed up with linguists (neurolinguistics) and began using a variety of brain imaging devices whose usage were becoming increasingly available—imaging scans such as PET, ERPs and fMRIs were newly adapted tools which neurolinguists could use in not only mapping the brain-to-language corollary, but to map it in action, (when subjects were engaged in specific language tasks). Based on these new studies, as it turns out, the classic Broca/Wernicke split is not as fixed as once assumed, with much more fluidity of movement (*spreading*) taking place between the two cortical pathways. I’ll come to call this spreading *Geschwind spreading* (see Fig. 2 below).

The fluidity certainly challenged naïve assumptions that the brain was to be thought of as a mere warehouse for ‘stored & retrieval’ of language components (a so-called ‘language-compartmentalized’ model). New problems seem to appear which suggested an anomaly of sorts: (i) that lesions at the Broca cortex didn’t necessarily trigger the onset of Broca’s aphasia, or that (ii) Broca’s region specifically subserved language grammar such as the decomposition of inflectional morphology. As it turned out, much more was being acted upon in either of the two domains.

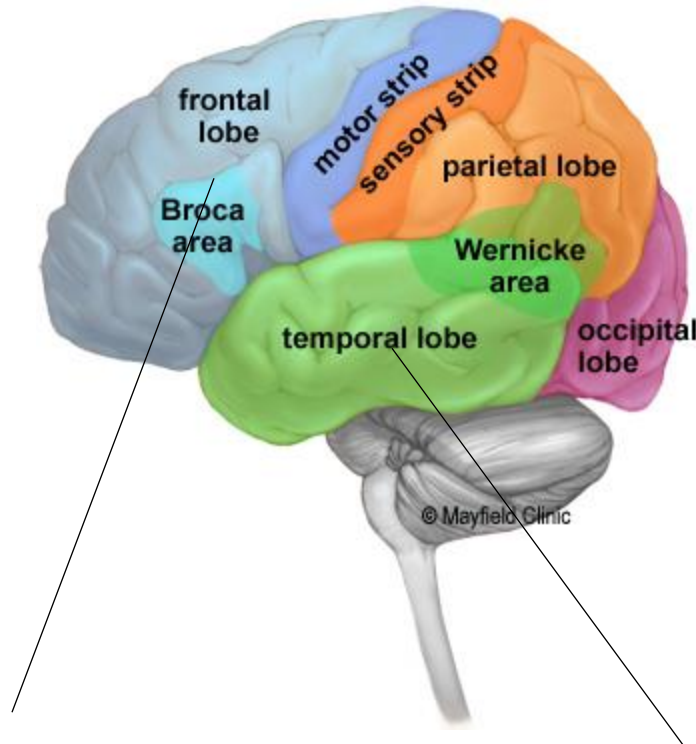
This is what I suggest is behind the term *spreading*. Other examples which show-up in neuroimaging studies demonstrate how FOXP2 (foxhead box: an 80-100 amino-acid motif) disruption seems to lead not only to language-task difficulties related to pervasive movement operations found in syntax (i.e., INFlection), but also with the planning of mouth movements associated with speech articulation. Such pervasive ‘MOVE-based’ difficulties—both in syntax as well as in the motor-control planning of mouth movements—suggest sub-cortical triggers

associated with FOXP2, Brodmann's area 44, 45, specific to Broca's area. Studies have shown that in SLI (specific language impairment), where FOXP2 is not affected, movement doesn't appear to be a problem.

However, when studies looked at DVD (developmental verbal dyspraxia) where the FOXP2 expression was disturbed, movement was impacted, both for syntax as well as for speech articulation. Though the traditional Broca/Wernicke split may be well-accepted as a 'general layout' of the brain-to-language corollary, we now know that language is subserved by quite diverse regions of the cortex—some involving a 'lateral-spreading' (horizontal) while others involve subcortical pathways (vertical).

(Figures 1-3 below are Google© 'free-to-use' images).

**Fig. 1 (below) Brain-imaging studies related to a 'Dual Mechanism Model' of the brain.**



The<sup>46</sup> (The book)

book (Noun)

Do, Be, Have<sup>47</sup> (do cook!, is cook-ing, has visit-ed),

cook, visit (Verb)

Inflectional morphology

Derivational morphology<sup>48</sup>

S (book-s)

S' (Tom's)

V + ing = N (gerund)

ed, en (visit-ed, spok-en)

N + ing = Adj

ing (is cook-ing)

[[stalk]ed]

[walked]<sup>49</sup>

<sup>46</sup> Determiners (*a, the, these, those, this*, etc.) are abstract, categorical based as they carry no semantic weight and merely introduce a Noun [D>N]. (Determiners project abstract-functional features such as Case, Definiteness, Agreement, number, Gender).

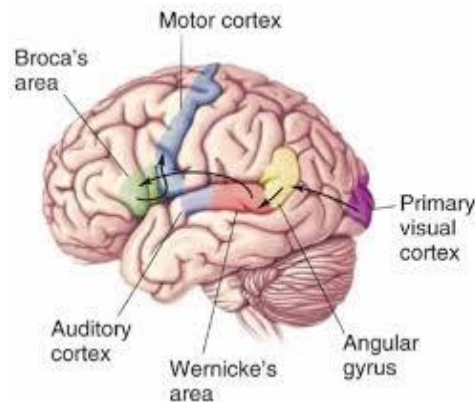
<sup>47</sup> Auxiliary verbs (*Do, Be, Have*) along with modals (*Can, Could, Shall Should, Will, Would*, etc.) serve in a similar counterpart manner to determiners in that they introduce verbs [Aux>V]. (Aux-verbs project functional features such as Tense, Agree on to the main verb).

<sup>48</sup> Note how the INFL Aux-ing' is different from the Derivational 'ing' (as found in gerunds, or adjectival derivational morphologies (e.g., 'The shopping was fun' (shopping = Noun), 'The shopping cart is broken' (shopping = adjective), versus inflectional Aux-grammar of 'Mary is shop-ing' at the mall (V+ing = progressive)

<sup>49</sup> See Clahsen & Rothweiler, 1992 showing how high-frequency regular verbs such as 'walked' could be stored as undecomposed chunks [walked] as compared to low-frequency

**Fig. 2. The Lateral spreading of grammar based on frequency-effects: from Broca to Wernicke (*walked* vs. *stalked*).**

### **Geschwind Model of processing**



### **Cortical-lateral 'Geschwind Spreading'**

<b>Broca/parallel</b>	<b>Wernicke/serial</b>
<b>Top-down</b>	<b>Bottom-up</b>
·Category of rule	·Frequency of item
·Attention	·Practiced
·Understanding/holistic	·Linear/reflexive

What we find regarding such cortical-lateral spreading is, as mentioned above, that the mere location of 'lesion-bundles' in Broca's area doesn't necessarily trigger Broca's aphasia. What this may suggest is that other subcortical pathways are involved in the disruption of processing—other lower sub-cortical neuro-bundles which subserve the surface cortex layer. Similar discrepancies have shown up revealing that once-assumed 'Broca-specific' inflectional morphology (involving movement) may in fact be processed in Wernicke's area (as supported by

---

regulars where the default rule [V+ [ed]] gets triggered, cf. [walked] v. [[stalk]ed]. Distinct Brain storage between two also appear with walked stored as an undecomposed lexical item (Wernicke's area) and where the decomposed [[stalk]ed] shows the inflection {ed} being processed in Broca's region. Spreading involving the insular-cortex suggests similar processes where over-practice of a novel item (Broca) spreads to processing in the insular cortex.

the insular cortex (see fn 68)). This spreading conflicts with earlier assumptions that these brain-regions were rather fixed, and supports a much more flexible model of the brain.

**Fig.3**

Michael Posner and Marcus Rachle, for instance, have studies which indicate that through practice, what starts out in the FLH-Broca's area can shift to the TL-insular cortex, and that what seems to motivate this shift is nothing more than repetitive motor-control of the specific task.