

Examining Mutation Boundaries in Internet Language Through the Lens of Complexity

Minimization

Matt Galindo

Department of Linguistics, Lake Forest College

Dr. Ying Wu

December 7, 2024

Abstract

This paper begins with a literature review examining contemporary theories of grammar. I espouse the dependency model which utilizes certain keywords around which linguistic constructions can be built. These constructions contain both syntactic and semantic properties. I then formalize an equation for mapping instances of these structures and propose the theory of *complexity minimization* for interpreting the purpose behind language mutations. In order to reduce the complexity of my model, I striate the plethora of language mutation types along four dimensions: *creation*, *addition*, *subtraction*, and *conversion*. I then leverage these categories to assess the nature of extant language mutations within the domain of internet language.

My paper concludes with a study proposal which would induct 250 subjects into a testing module. The purpose of this module is to expose the subjects to novel and derivative internet language mutations and test for their ability to interpret the new terms, providing an increasing level of context to aid in their assessment. The result will either support or reject the underlying feature space. If supported, the ML model will be able to accurately predict the degree of interpretability for internet language terminology provided the categorical assessment formulated in this paper.

Chapter 1: Literature Review

Generative & Construction Grammars

Noam Chomsky's 1957 book titled "Syntactic Structures" is one of the cornerstones for the modern interpretations of syntactic structures in language processing. In his book, Chomsky defines an approach, contemporarily called "Generative Grammar" (GG), which has become one of two competing theories for our modern understanding of syntax. His approach argues that grammar is an *innate* structure, meaning that its rules are formulated by human biology, and therefore any inquiry into its structure should isolate syntax from other linguistic properties such as semantics and pragmatics.

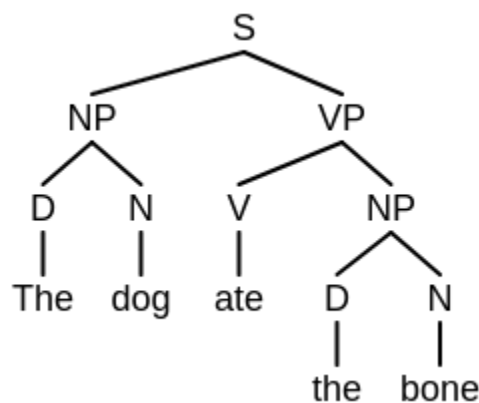
Chomsky specifically focuses on sentence formulation as the root of grammatical considerations, largely leaving morphology (or word-level considerations) out. In "Syntactic Structures" he begins by claiming there is "a set (finite or infinite) of sentences, each finite in length and constructed out of a finite set of elements." From there, he separates all possible sentences "L" into one of two categories: grammatical and ungrammatical. The goal of GG is to discover which set of rules could be utilized to formulate all grammatical sentences and none of the ungrammatical ones.

In addition to the innate faculty (also known as Universal Grammar) of language, GG also differentiates human language from other forms (ie. animal languages) through principles such as recursion, or the ability for human language to generate an infinite number of sentences from finite components. This can be illustrated through the concept of *central embedding*, or embedding a phrase in the middle of another phrase of the same type. For example, "Mary chased the dog" becomes "Mary, who loves apple sauce, chased the dog" becomes "Mary, who loves applesauce that has been refrigerated for at least eight hours, chased the dog", and so on, ad infinitum. This landmark proposal conflicted with the popular behavior theory at the time which argued all linguistic phenomena could be reduced

to operant conditioning (Chomsky, 1980). However, if language was indeed innate and generative, this would mean it was somehow different from mere learned behavior.

Generative Grammar espouses a series of *Phase Structure* rules which define the hierarchical nature of a sentence. The figure below is an example of a syntax tree.

Figure 1



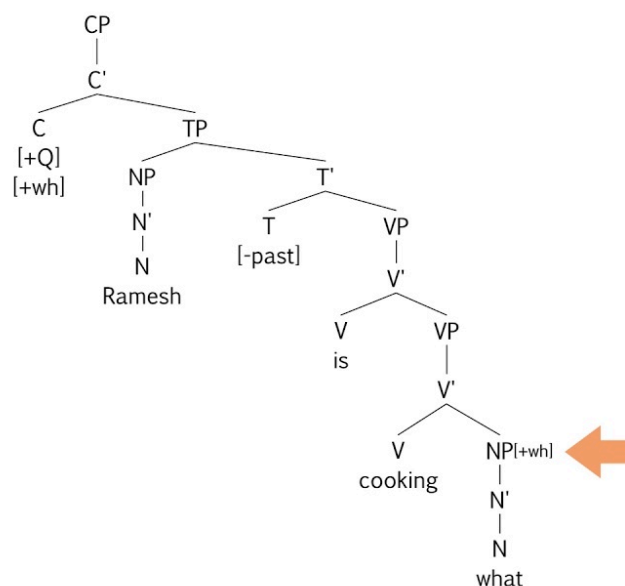
Note how the root of the sentence “S” is broken into a Noun Phrase “NP” and Verb Phrase “VP”, each of which are broken down further according to the phase structure rules. “The” is a delimiter “D” attached to the Noun “N”, “dog”, which again can be combined to form the NP, “the dog”. In GG, all leaf nodes must be generated from their parent nodes, which move up the levels until finally we have the complete sentence “S”.

There are few linguistics today, both within and outside the Generative school of thought, who would argue against a hierarchical representation of language. However, as more research was conducted within the domain of GG, certain *Transformation* rules, or manipulations in a particular sentence structure to generate a different sentence form, began to call into question the efficacy of this approach.

One rule in particular called *movement* became difficult to reconcile with the intuitive approach of phase structure rules.

In Chomsky's 1977 paper "On WH-Movement", he defines movement as the condition where a particular word in a syntax structure is "moved" to a different position in the structure. See the tree below as an example:

Figure 2



Note how in this structure, the initial sentence "Ramesh is cooking *what*" becomes "*What* is Ramesh cooking?". "CP" stands for Complementizer Phrase, which is the "projection" which hosts the wh-word "what". The TP (Tangent Point) also called IP (Inflectional Phrase) represents the rest of the sentence which is separate from the wh-word.

However, several linguists later pointed out two problems: firstly, not all words “move” in the same way. Take the sentence “Levi slept”. If we want to formulate a question sentence, we would have to generate “*Did* Levi sleep?” as opposed to “*Slept* Levi”. Furthermore, it is not obvious how a child could acquire language which incorporates both phase structure rules *and* movement (Fridman & Gibson, 2024).

Ivan Sag, another prominent linguist, later posed an alternative theory in his 1992 paper titled “Lexical Matters”. In his formulation, instead of having a single word “what” which can take different positions within a single syntax tree, we instead encode different *lexical copies* of the word “what” which have different senses. Utilizing Sag’s approach, the word “will” could formulate two different trees. Below are two examples of different use-cases for the word “will”, separated into separate instances.

Figure 3

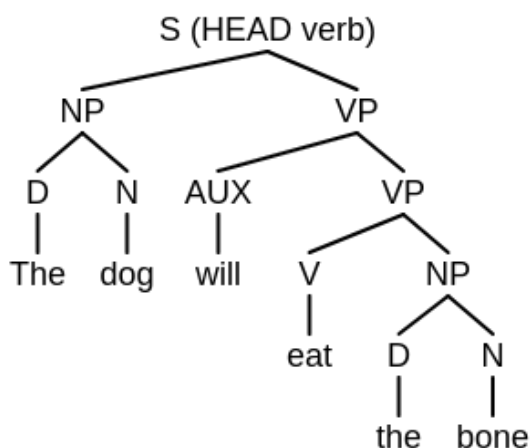
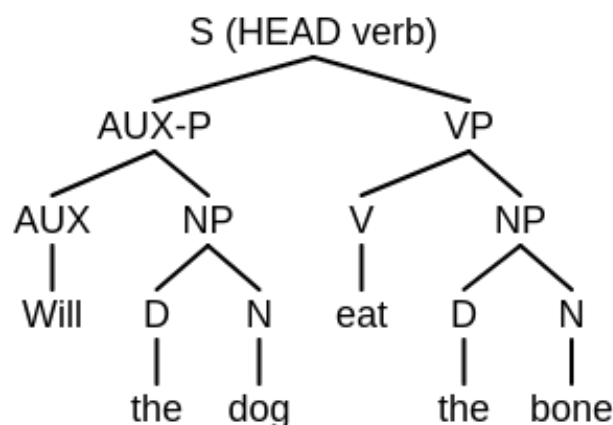


Figure 4



This approach is consistent with a phase structure approach, but restricts movement, instead focusing on morphological (word) units as the basis for interpreting which syntactic structure is utilized.

Wh-movement is just one feature which begins to fracture the most austere interpretations of GG, opening the door to more nuanced interpretations.

At this same time, work was being conducted within the field of Analytic Philosophy to address semantic considerations in the role of linguistic structures. One of the foundational publications in this domain was Ludwig Wittgenstein's "Philosophical Investigations" which attempted to reconcile his earlier work "Tractatus Logico-Philosophicus" with a more pragmatic interpretation of language.

The result was his theory on *Language Games* which purports that language is only interpretable within the context of which it is used. He provides an example of how a construction worker may call "Slab!", indicating that he wishes another worker to hand him a slab of stone. He then goes on to describe how "in the practice of the use of language, one party calls out the words, the other acts on them. In instruction in the language the following process will occur: the learner *names* the objects, that is, he utters the word when the teacher points to the stone."

This contextual approach is further exemplified in Tor Nørretranders' book "The User Illusion" where he recounts an event that "took place in 1862. Victor Hugo—famous for writing *The Hunchback of Notre Dame*—had gone on holiday following the publication of his great novel *Les Misérables*. But Hugo could not restrain himself from asking how the book was doing. So he wrote the following letter to his publisher: '?' His publisher was not to be outdone and replied fully in keeping with the truth: '!'" This single symbol conveyed to Hugo that his novel had been a success and was accepted by the public.

Notice how, in both of the above examples, the context within which the words (or symbols) were conveyed play a crucial role in their interpretation. Although these particular examples don't conflict with GG per-se, when combined with Sag's work, they leave the door open for a new perspective which was born in 1959 in Lucien Tesnière's seminal paper titled *Elements of Structural Syntax*.

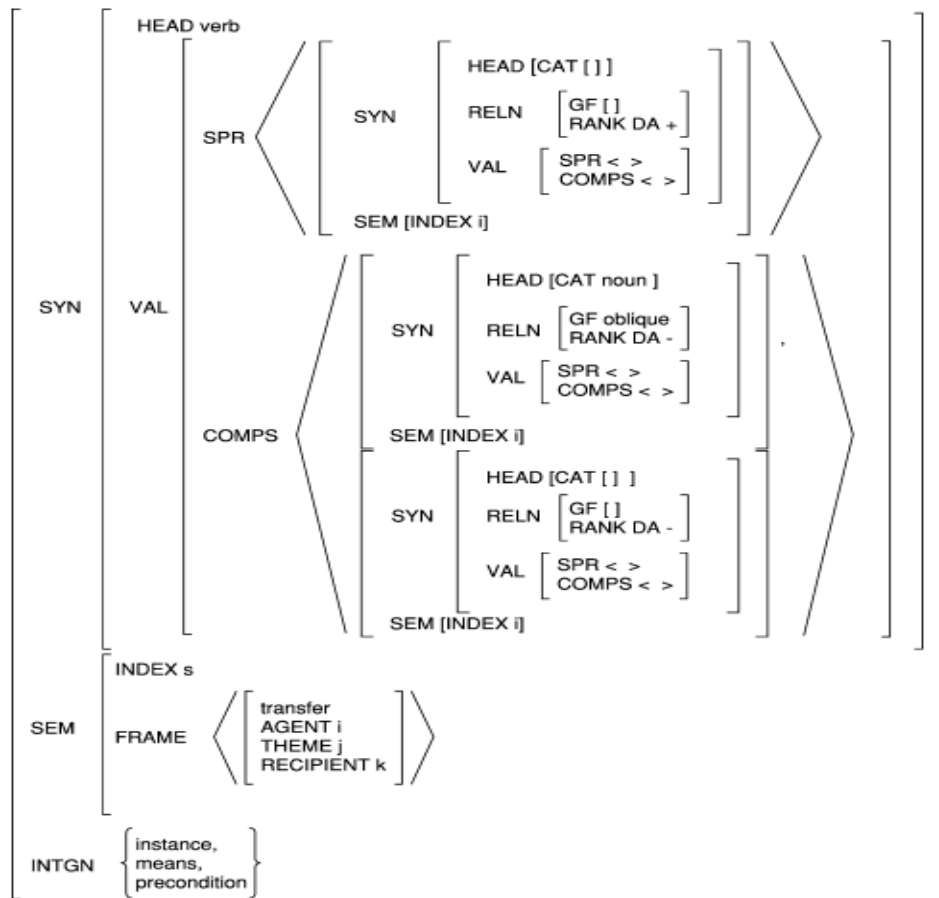
In Tesnière's paper, he describes syntax as a mapping of word-level dependencies, wherein "each connection unites a superior term and an inferior word." In order to decode a complete sentence, "it is

necessary to decide on the central node of the sentence.” From there, the sub-nodes, or dependencies, can be ascertained.

The distinction between GG and dependency grammar lies in how grammaticity is tracked across syntactic structures. While GG posits a universal set of rules which govern how sentences are formed, dependency grammar emphasizes the role of key words around which the rest of the sentence is formed. This bottom-up approach has excited a new approach to grammar, called Construction Grammar (CxG) which emphasizes usage as the key driver of grammar (Newmeyer, 2000). The result is an attempt to build out structures around these dependencies which can be reused and recycled.

Because of CxG’s focus on usage, it becomes necessary to incorporate semantic and pragmatic elements into their structures, since it is these elements which define their use-conditions. The resulting “form:meaning” pairs are constructed through real language data, which are then grounded in cognitive processes (Goldberg & Subtle, 2010). For example, the “ditransitive” construction seeks to understand the use-case of ditransitive verbs like “give” or “send” as in “Mary gave Bob *a haircut*”, and “Denise sent Lucas *an email*.” The following model is taken from Laura Michaelis’s paper “Construction Grammar” which was published in the *Encyclopedia of Language and Linguistics*:

Figure 5



Notice how the model includes VAL (valence) and SEM (semantic) elements in addition to SYN (syntactic) elements. The HEAD indicates the syntactic category of the phrase, which in this case is built around the (ditransitive) verb. SPR indicates the presence of a subject or determiner element, and COMPS describes the complement requirements of the verb. In this example, the ditransitive verb (give) is the central node which defines the dependencies for the rest of the sentence. This allows us to dissect the structure of the sentence without needing to embed it inside a higher structure; we isolate it as a modular construction which is presumed to exist as an independent, psychological structure.

One of the hallmarks of CxG is idiomaticity, which is the creation of phrases which contain meaning outside that of their generative components. Take the phrase “break a leg”. This phrase, when spoken literally could only be viewed as a malicious remark. However, when used colloquially, it is transformed into an encouraging comment. The distance between the “literal” meaning and the “idiomatic” meaning suggests that there are parts of language which must be learned as structures independent from a strictly generative grammar. However, most CxG proponents take idiomaticity a step further, claiming that even common sentences such as “Tim went to school” should be viewed through a similar lens of idiomatic construction. The question then arises, to which degree is grammar “generated” vs. “constructed”? (Ungerer & Hartmann, 2023)

Neither school has been able to, as yet, define a complete model for language according to either GG or CxG; however, on the dependency grammar front, there has been much work done into attempting to mathematically model the dependencies within linguistic forms. In the next part, I will take a more probabilistic approach toward grammar, highlighting recent successes in the field of NLP, with a particular focus on LLMs.

LLMs and Probabilistic Modeling of Grammatical Dependencies

A single English letter has $1/26$ possible representations. Two letters in combination have 26^2 possibilities. However, not every letter in position[1] can combine with every letter in position[2] to form a real English word. For example, we must exclude “xz”, “yu”, “gp”, and so on. This means that the effective possible combinations of two unknown English letters “XX” are somewhere between $1/27$ and 26^2 possibilities. What if we are given the knowledge that the letter in position[1] is “a”? We now know that we have “aX”. If we try every possible letter for “X”, we come up with the following possibilities: [aa, ab, ac, ad, ae, af, ag, ah, ai, aj, ak, al, am, an, ao, ap, aq, ar, as, at, au, av, aw, ax, ay, az].

In conventional English, we would say that there are only 4 possible words, meaning that knowledge of a single letter ($1/26$ possibilities) can actually constrain the possibility of our second letter ($1/4$) which

is less than that of random chance ($1/26$). However, I use the delineative term “conventional English”, because there are also morpheme considerations, as in abscond, modular, procedural, as well as *unconventional* usages such as *ad* to mean “advertisement”, *ak* as shorthand for ak-47, *ai* which is an acronym for “artificial intelligence”, and finally exclamation markers like “ah” and “aw”. However, even when considering all possible use-cases, we have constrained the total possible combinations to $1/16$.

The purpose of this experiment is to show how the English language can be viewed as a probabilistic structure. In Tor Nørretranders’ book “The User Illusion”, he explains how Claude Shannon came to this same conclusion when attempting to assess how much it would cost for telecom companies to transmit messages from one place to another. In his analysis, he proposed a “surprise value” which could be calculated given a string of English words. He used the fact that there are twenty six letters in the English language, and our “surprise” is precisely the fact that a particular letter was chosen in place of all other possible letters.

Nørretranders then wraps this discovery within his previous formulation on entropy, where he explains how human beings are cognitively limited in our ability to calculate *microstates*, or all possible permutations which make up a *macrostate*. His chief example in this regard is temperature. Temperature is an expression of heat, which is the movement of subatomic particles within a given area. The net sum of these vectors is what we call “temperature”. However, the underlying organization of each individual subatomic vector remains a mystery to us. There could be one of an infinite number of configurations of subatomic particles which combine to create the condition of 20 degrees celsius. Despite this ignorance, it is sufficient for us to understand 20 degrees celsius, or the *macrostate*.

Returning to Shannon’s theory, he proposed that language can be interpreted as a probabilistic structure which has a weighting scheme that corresponds to how “surprising” a particular letter is in

the string of words. In other words, some letters convey more information in a strictly entropic perspective than others.

Let's consider a second two-letter word, except this time we know the first letter is "g", such that "gX". Here is our new array: [ga, gb, *gc*, gd, ge, *gf*, *gg*, gh, *gi*, *gj*, gk, *gl*, *gm*, *gn*, go, gp, gq, gr, *gs*, gt, gu, gv, gw, gx, gy, gz]. In this set there is only one conventional word (1/1 chances) as opposed to four in the previous set (1/4 chances). Even if we extended usage, we would still have a 1/11 probability as opposed to 1/16. This means that knowing the first letter is "g" reduces the number of possible microstates corresponding to the 2-letter-word macrostate. As a result, we would say that the letter "g" provides more information.

As a final exercise, consider the show Wheel of Fortune where participants have to propose letters which will fill out an empty field, then guess the words based on the context of those scattered letters. There is a cost associated with "buying a vowel", since it is certain that every word will contain a vowel, thereby reducing the risk associated with choosing one. On the other hand, choosing a letter like "z" or "j" is perceived as "riskier" since these letters are less commonly used letters. However, a well placed uncommon letter will provide more information than a common letter like "s" or "t", meaning that discerning that particular word will be easier to the extent that it will reduce the number of possible words (macrostates) that it could be (Norretanders, 1999).

We can then build on this formulation. For example, while words contain letter combinations, sentences are word combinations, and paragraphs are constructed with sequences of sentences. In other words, language involves a complex layering of probabilistic frameworks, each of which is hierarchically organized. This provides us with a descriptive representation of generating the form of language which operates as the basis for contemporary NLP (natural language processing) models.

Modern LLM's scan language inputs using transformers, or neural network architecture that parallel processes input sequences by analyzing the relationships of elements within those sequences and attempts to find patterns which would indicate the next most likely output result (Change, et.al, 2024). One of the competing theories of *how* LLMs do this is through the lens of dependency grammars. In other words, LLMs are able to parse text in order to calculate long-range dependencies, which then allows it to contextualize the data and provide an extremely compelling response.

Returning to the CxG perspective, we can assess a sentence like "Mary asked Ben a question" through the lens of dependency grammar. One intuitive way to do this is by identifying *asked* as the central node which defines the construction. In an "asked" construction, we need an "asker" (Mary), an "asked" (Ben), and an "object" (a question). But what happens when we increase the distance between the dependencies? For example, let us consider the following sentence: "Mary *asked* Ben, who earlier threw Gary, who has never written a resume, a ball, *a question*." This sentence, which actually contains three embedded clauses, is extremely hard to track. The reason for this is because the central node is separated from its complementary sub-nodes, leading to the possibility of a limitation in the aforementioned recursion ability of language. In fact, when we propose the same questions to LLMs, they *also* find it difficult to understand! (Fridman & Gibson, 2024)

This phenomenon has led researchers like Dr. Christopher Manning, the current Director of the Stanford Artificial Intelligence Laboratory, to develop models for Universal Dependencies, or certain morphological features and part-of-speech tags which annotate grammatical structures *across* languages. Their current goal is to utilize the computational power of LLMs to build out a dependency model which could generate the rules for building language constructions (Manning, 2015).

However, despite the revolutionary nature of LLMs and their accuracy on mimicking the form of language, researchers are still largely dubious as to how exactly LLMs calculate these dependencies, as they contain billions and trillions of statistical layers all working in concert (not a very "intuitive"

model). Furthermore, there is a vast expanse between the way humans and machines formulate language. That expanse is *semantics*.

In a Lex Fridman podcast, Edward Gibson, the head of the MIT Language Lab, said that “I would argue [LLMs] are doing the form... really well. Are they doing meaning? No. Probably not.” He then provides an example of how one can use the Monty Hall problem to trick an LLM.

“The Monty Hall problem is this silly problem... You have three doors, and there’s a prize behind one, and there’s some junk prizes behind the other two, and you select one. Monty knows where the item is, he knows where everything is back there. [Then] he gives you a choice, to choose one of the three [doors]. And after you choose, he opens one of the doors, and it’s some junk prize, and then the question is: should you trade to [choose] the other [door]? And the answer is, ‘yes’, you should trade, because he knew which ones you could turn around, and so [trading] would give you two-thirds odds. And if you just change that story a little bit... and say ‘there’s three doors, and behind one there’s a good prize, and behind the other two there’s a junk prize, *I happen to know [the prize] is behind door #1—the good prize, the car, is behind door #1*, then Monty Hall show’s me door #3, should I trade for door #2?’. The Large Language Model would say ‘yes, you should trade’, because it just goes through the forms it’s seen before so many times.”

In other words, while LLMs are extremely proficient at interpreting and generating *the form* of human language, it is unlikely they understand *the meaning* of human language. This leaves the question open: just how exactly does form and meaning interact to generate the different words and sentences we encounter and use everyday? In the next chapter, I will define a basis for bridging this gap through the creation of semantic structures. Then I will formalize a system for analyzing the efficiency of these semantic structures which I will call “signal maps”.

Chapter 2: Utilizing the Principle of Complexity Minimization for Analyzing Semantic Structures

Theoretical Basis for Semantic Structures

Imagine you enter a completely dark room. The dimensions, size, and contents of the room are unknown to you. Then you hear the utterance of the word “left”. You turn left in expectation of a corresponding stimulus, but there is none. You then hear the word “left” again. Again, you turn left, but still there is nothing. The meaning of “left” in this context is its *pure* meaning. In other words, “left” is *always on the left*.

Now suppose we introduce a single source of light which illuminates nine identical objects placed equidistantly all around you. The word “left” is uttered again, and when you turn, you see four of the objects present in your field of view. The word “forward” prompts you to move forward five steps, and directly at your feet is one of the identical objects. You pick it up. This reveals the contextual meaning of the words “left” and “forward”. They are orienting structures which serve the purpose of isolating dependent objects.

Now suppose you are once again in an illuminated room and there are two identical objects spaced equidistant from you on either side. The word “object” is uttered, and you have to make a choice. The decision to move right or left is arbitrary, but the decision itself is not. There are only two objects in the room, which constricts the dependents to two.

In this thought experiment, there are four conditions necessary to identify an object: the room, the light source, the utterance, and the object. The utterance alone provides an orientation without an object. The object alone is unfindable. The light source alone provides a view of the objects with no way to reach them. The room alone provides a space to move, but nothing to find.

We can liken a lighted room to an identity space. The utterance is a locative. And an object is the meaning. Therefore, the meaning of the word is dependent upon the room we’re standing in and the instructions provided to identify the object.

A *signal* is any linguistic sign which projects at least one identity space which contains at least one object. This could be likened to the denotative phrase “that” as in “that book” which references one book in one space. However, language need not be this direct. Take, for example, the word “ball”. The

word *ball* has multiple senses, which is akin to saying that it projects more than one identity space. Consider the following sentences:

“He kicked the ball.”

“She danced at the ball.”

“I like to ball.”

Without more context, which in this case takes the form of surrounding words, the meaning of “ball” is one of several possibilities. The purpose of a phrase or sentence, then, is both to collapse the identity space of the constituent words and provide coordination for accessing the object or “meaning” of those constituents. This means language requires a parallel processing of both linguistics and semantic dependencies.

For example, the sentence “Jim stick” is incomplete, since it lacks a verb to coordinate the two nouns.

Similarly “Jim went to stick” is incorrect, since it coordinates the words linguistically, but does not have a discernable semantic meaning.

The first sentence is akin to standing in the lighted room with two equidistant, identical objects on either side. There are no clear instructions provided to orient yourself in relation to those objects. The second sentence is akin to being in a lighted room and told to turn “left”, but there are no objects present there. The form of language must conspire to direct the observer toward a single or a set of objects (meanings) with an appropriate relationship.

Even if a sentence contains a complete clause, it may have markers which make it incomplete. Take the following two sentences:

“*Since* Jim likes baseball.”

“*The reason* he went to the bank.”

Both sentences create an identity space and point us toward a meaning, but the *full* meaning of the sentence is incomplete, because there is a promise of more context. In this way, identity spaces can make up higher dimensionalities. This is linguistically known as *embedding*. To complete the sentence, we must add another clause, such as:

“Since [Jim likes baseball], he went to see the game.”

“The reason [he went to the bank] was so he could withdraw money.”

And just like simple sentences, these also require semantic continuity.

“Since [Jim likes baseball], he sailed on a pirate ship.”

“The reason [he went to the bank] was to see a doctor.”

In this way, the corresponding identity spaces must converge such that all objects are present in the meta-space. This “meta-space” can be likened to an instance of an Identity, which can loosely be defined as a framework for interpreting a contextualized space. For instance, we can imagine driving to a gym. When inside, we have a certain expectation of what we’ll find: workout equipment, staff, a locker room, etc. Within the context of this larger structure, we can modularize our understanding and create a domain of expectation for which instances are likely to occur, and which are unlikely or impossible.

Ludwig Wittgenstein’s “Language Games” touch on the contextual spheres within which we build out our understanding of the utility of language. Some of the clearest examples of this are within the contexts of stories or internet memes. In both cases, there are rules, often not explicitly defined, which encapsulate our understanding of the story or meme. We can then interact with either according to these rules. For example, in “Harry Potter”, it would not be strange to read the sentence “she cast a spell”, but the meaning would change if we were speaking about one of our relatives. Similarly, a popular meme which showcases an image of a smirking young girl with a house on fire behind her might pair meaningfully with the caption “There was a spider in the bathroom. It’s gone now.” but not with, “I’m afraid of doughnuts.”

While defining an Identity is difficult, we have an innate sense when we are within the scope of one, as well as when we deviate. The most compelling case for this is when we see a comment thread on Youtube, Reddit, or one of the other social media platforms, where each comment builds off one another. This may be a concatenation of lyrics, a sequence of numbers, or a string of similarly structured jokes. You will probably also see the moment a comment is added which breaks the chain, leading to a string of complaints.

Leaving aside the formulaic aspects of language for a moment, we can focus on the structure of the semantic components of an ever increasing string of words, such that:

$$S_m \rightarrow P = J$$

Where “S” is the *signal* and “m” is the *method* of transmission, which results in a dependent “P” *projection* (a set of identity spaces with possible object references) that leads to a judgment (a collapse of identity spaces into one perceived meaning). Returning to the word “ball”, we read left to right: “Adam fetched the ball.” Our judgment is precisely the perceived meaning of this sentence in relation to the next sentence (or signal), such that: “Adam fetched the ball which Caroline threw. *He* then threw it *back* to *her*.” Notice how if we started with the second sentence, we’d have semantic ambiguity. *Who* threw *what* to *whom*? Therefore, if we process a continuation of meaning, we’d see that:

$$\begin{aligned} [S_m]_0 &\rightarrow P_0 = J_0 \\ J_0 + [S_m]_1 &\rightarrow P_1 = J_1 \\ J_1 + [S_m]_2 &\rightarrow P_2 = J_2 \end{aligned}$$

And when we formalize this into a single equation:

$$\begin{aligned} P_i &= f([S_m]_i, J_{i-1}) \\ &\dots \\ \text{Meaning}_n &= \sum_{i=0}^n P_i + h(J_n) \end{aligned}$$

Where $h(J_n)$ is a function which captures the cumulative influence of the final judgment on overall meaning.

Utilizing this equation, we can establish certain thresholds for discerning discrete semantic units. We would say that the *meaning* of a semantic structure is *complete* when no more signal is necessary to make a *judgment*. We would say that two discrete semantic structures combine to create a *meta-structure* when the second signal is *semantically dependent* on the first structure. The meaning of

a *meta-structure* is the resultant judgment of the sum of the projections constructed from both signals. There is no limit to the number of signals which can be added to this process.

Signal Mapping for Complexity Minimization

Language mapping, formalized by the equation above, can be best understood in reference to a concept which I will call the *Ideal Map*, where the time to formulate a semantic structure is 0 seconds, and the time to interpret that semantic structure is 0 seconds. In other words, the form of a word, sentence, or string of sentences is absolutely minimized such that a judgment could be made instantaneously. In practice, achieving an *Ideal Map* is impossible, since even the smallest thought must be communicated in a time space, and the interpretation of the *intended meaning* of that thought is also calculated in a certain amount of time.

Therefore, we can operate under the assumption that language operates under a principle which I will call *complexity minimization*, which attempts to minimize the time complexity of generating a form in order to achieve its appropriate judgment. If we view language as a probabilistic structure wherein a meaning is equivalent to the scope of possible meanings (objects in an identity space) subtracted from all discarded meanings (objects which exist in this identity space which *aren't* the intended meaning), we can calculate the efficiency of a signal. Take for example the following two remarks:

The young and small cat entered the room.

The kitten entered the room.

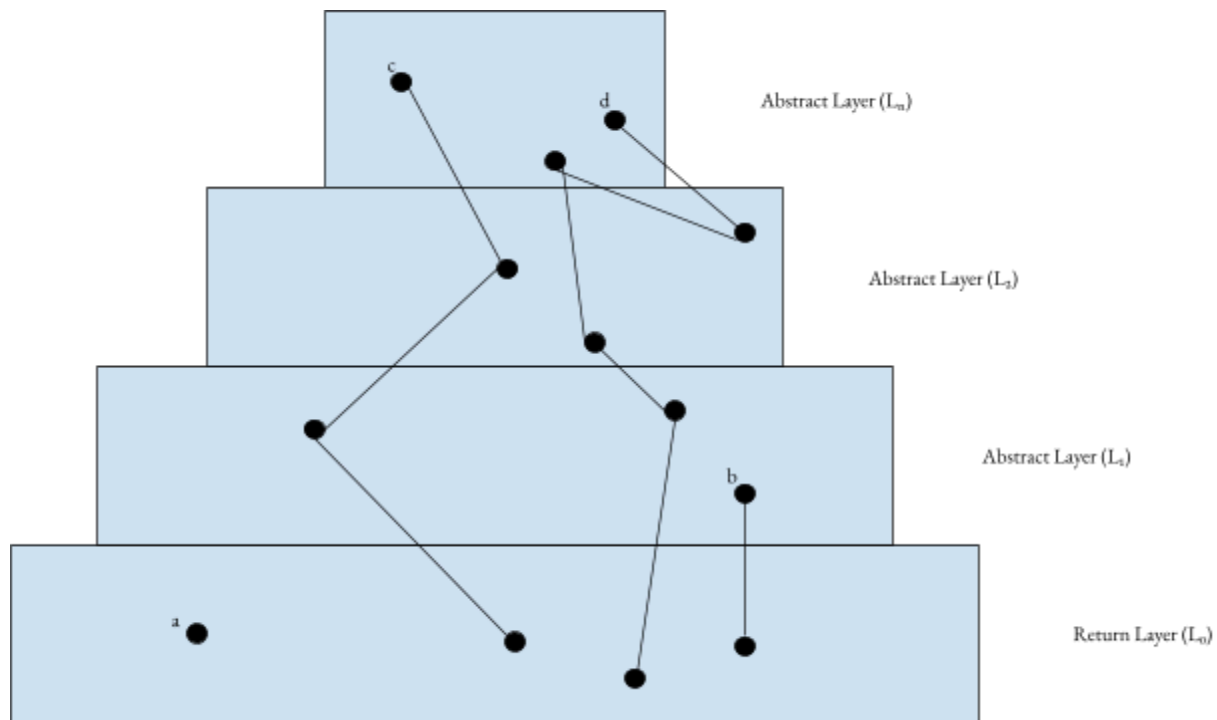
We would say that the second remark is more optimized if the same judgment is achieved. The term *intended meaning* is purposefully vague since it is contingent on the subjective intention of the issuer of the statement; however, we can see a more concrete example if we consider the following dialogue:

“Jim went to see Sarah on Friday. Actually, it was Thursday.”

This sentence is inefficient exactly to the extent that the second sentence requires a semantic modification of the first. In other words, complexity can be expanded if the words used to achieve an equivalent judgment are more verbose, or if a semantically dependent sentence modifies a previous judgment.

In order to more effectively visualize this phenomenon, consider the following diagram:

Figure 6



In Figure 1, the return layer L_0 represents the dimension of judgment as previously specified, and L_n represents the highest layer of abstraction which we must traverse down to achieve the judgment. Subsequently, each variable represents a mapping pattern. In the case of “a”, an *immediate response* is

generated, corresponding to a single iteration ($[S_m]_0 \rightarrow P_0 = J_0$). This is also the level of *reflex*, or immediate signal:response. The pattern “b” represents a single traversal, where two judgments are created to formulate a minimal meta-judgment (summation of projections into a single space). The remaining mapping patterns can fit within two categories. In the case of *efficient mapping*, “c” entails an entrypoint $> L_1$ whereby the path to judgement dynamically maps one point to the next down each layer without returning to a higher layer. *Inefficient mapping*, exemplified by “d”, entails an entrypoint $> L_1$ whereby one or more instances of returning to a higher layer is required.

Before I link signal mapping to language mutation, it is important to first distinguish the kind of inefficiency I describe above, which I will call *external inefficiency*, with the implicit or *internal inefficiency* always extant within any language.

Beginning with our concept of an *Ideal Map*, we can build out the form of language one piece at a time. At the most basal level, there are constraints on phonetic usage. For example, English contains a limited alphabet wherein certain letters cannot be combined with others. For example, we do not see the following combinations in English: “gd”, “zp”, “fb”, etc. These constraints have been naturally defined by speakers of English, perhaps because of conflicts in the place of articulation, or because certain sounds are difficult to generate and interpret (Fromkin et al., 2021).

Secondly, there are word order constraints. We might wonder why not every *possible* combination of letters is utilized to create new words. For example, why is there no meaning associated with “av” or “az”, even though these are concise formulations? For one, “az” has phonetic similarity to “as”, which could cause conflicts in interpretation. On the other hand “av” was simply never lexicalized. The reasons for these missing word associations are important to linguistics research; however, the specifics are not necessary to define in this study. Instead, I will simply note that they exist and are peripheral to *external inefficiency* such that *external inefficiency* can only be calculated *after* all *internal inefficiency* is already accounted for.

Chapter 3: Examining the Logical Link Between Language Mutation and Signal Mapping

On Identification and Calculation

In order to establish a baseline for interpreting language mutation, we must first establish a baseline for interpreting the discrete elements of a sentence in relation to their semantic meaning. My proposal on this front involves two categories: *identification* and *calculation*.

Returning to our thought experiment, *identification* is the process of using word(s) or icons (linguistically equivalent symbols) in order to reference a room (which contains a set of objects) or an object (which may be located in many rooms). These categories are not mutually exclusive and are dependent on the context with which the words (which I will call keys) are used.

Let's once again examine the word "ball". We would say that this *key* can reference one of a set of objects (different types of balls) or a specific ball (as in a baseball). Since these keys are lexicalized, they are dependent on the specific knowledge of the user, and therefore exist as fluid concepts (which is a necessary condition for language mutation). However, we can treat these signifier-signified relationships as objective units when used in the context of a particular instance (Chandler, 2022).

Therefore, once we have establish the signified object, we can then apply transformations to update our interpretation of this particular object according to the following logic table:

Figure 7

Category	Signal	Judgment	Type
<i>Identification</i>	cat	cat(animal)	local
	red	color	global

<i>Calculation</i>	beautiful red flower	object +	amalgamation
	sleeveless shirt	object –	reduction
	he kissed her	$A * B$	relationship
	he is Paul Reed	$A = B$	equivalence

Within the domain of “Identification”, we have a *local* judgment, which singles out an object, and a *global* judgment, which references a category. In our “Calculation” schema, we have amalgamation and reduction features which are exemplified by adjectives. Since we are focusing on semantic distinctions, we are not concerned about the part of speech, only *how* that part of speech affects our interpretation of the identified local or global element. These transformations commonly occur in sentences. Their link to signal mapping and efficiency only becomes apparent when weighing a particular key against other use-cases. Take for example the next three sentences:

1. “I saw an *origami cat* on the table.”
2. “I saw a *piece of origami shaped like a cat* on the table.”
3. “I saw a *piece of paper folded to resemble a cat* on the table.”

Assuming each of these sentences produce the same judgment in relation to the object in question, we would say that sentence #1 has an efficiency advantage along the dimension of form, supposing that the interpreter has lexicalized the meaning of “origami” and “cat” and is able to amalgamate the meaning as effectively as the other sentence examples.

In our reduction example, I use the phrase “sleeveless shirt” which is only a reduction insofar as the interpreter’s schema of a shirt includes sleeves. Therefore, the additional word is necessary to subtract a property presumed in the initial identification of the object “shirt”.

Our last two rows move from operating within the space of a single object into examining inter-object relationships. “He *kissed* her” is a simple SVO word order sentence in English, but it shows how relationships (and verbs in particular) are necessary to chain objects together in order to create dynamic meanings. On the other hand, *equivalence* simply identifies one object as equal to another.

When the length of the sentence grows to map an increasing array of dependencies and object-relationships, an *Identity* or aggregate meaning will form according to the mathematical formula previously supplied.

This table will allow us to identify particular semantic features of a sentence, which when weighed against derivative forms will allow us to assess the efficacy of a particular mutation—which may be as simple as modifying adjectives in a sentence or as complex as examining the trade-offs of iconization of internet language.

What is Language Mutation?

Language mutation occurs naturally in every language. The form and meaning of words change, sometimes even without our noticing. For example, the word “sick” means to be ill; however, when used in the sentence “that’s so sick!”, the valence changes from negative to positive. This is called *semantic inversion*. Another example is the word “acc”, which is a texting short-hand for the word “account”. This is called *clipping*. Within the context of the logic table, we could say the first example is an extension in meaning which is effectuated by an expansion in the *identification* of a particular object, whereas the second example is an equivalence (acc = account) which results in a net subtraction in form. While these examples fit neatly into a particular category, this is not always the case. For example, the word “dead” as in “I’m dead” has become a popular colloquial internet term which means “to be metaphorically dead from laughter”. This particular term is also often iconized and referenced in the form of the skull emoji (pictogram) 🦴 — which provides both a change in form *and* meaning to the signal.

Since hundreds of categorical candidates for language mutation exist in the linguistics literature, all of which pick out a particular type of shift in the form or meaning of a discrete unit of language, it becomes necessary to simplify the feature space while still reserving sufficient complexity to define the boundaries for a particular mutation. My candidates for these features include the following: Creation, Addition, Subtraction, and Conversion. Below, I will illustrate each of these features in detail and describe their specific link to elements in the logic table.

Creation

Creation is the spontaneous production of a form:meaning pair *without any derived meaning*. The word “Google” is a stand in for the company which manages a search engine. We can talk about and reference the company and their dealings by using their name. This type neologic formulation is distinct from other words like *streetlight*, because there is no etymological tracking prior to the creation of the term.

However, Creation is not limited to neologisms. Idioms such as “break a leg” or “two birds, one stone” are lexicalized phrases which also involve a form:meaning pair wherein the association between the form and meaning is 1:1. Furthermore, if we were to apply the Creation function along a phonological dimension, we could add onomatopoeia usage to this category, as a word such as *buzz* corresponds to a particular sound (like that which a bee makes).

Addition

Addition can be further stratified along three dimensions. The first two are *solely form based*. These are *Intra-word* Addition, which involves adding letters to a word, and *Inter-word* Addition, which involves adding words to make a phrase. The third category is *Semantic* Addition, which as we have already described means to add properties to the instance of an identified object.

Intra-word Addition includes *affixation*, or the addition of an affix to a word. Take for example “happiness” is the concatenation of “happy” with the suffix “ness” to describe a state of being happy.

Inter-word Addition can be represented by *reduplication*, wherein the first word is repeated for emphasis, as in “like like” to describe that you like someone romantically. Semantic Addition involves an extension in usage wherein the former definition remains. This is true for a word like “sick”, where it takes on an additional valence.

Subtraction

Like Addition, Subtraction can be broken down into similar categories. Intra-word Subtraction assimilates mutations like *backformation* where an affix is removed. Consider the word *edit* which was derived from the word *editor*. This also includes *clipping*, as in shortening a word for ease of formulation: “def” in place of “definitely”. On the other hand, Inter-word Subtraction includes *blending* (brunch → breakfast + lunch), *contraction* (can’t → can not), and *compounding* (break down → breakdown). Semantic Subtraction is the attenuation of a word’s meaning, often called *Semantic Narrowing*. Take the word “meat” which used to mean “any kind of food” and now specifically means “the flesh of an animal used as food”.

There are also Subtraction quantities which contain a phonological dependency. This is apparent in *Initialism* and *Acronymism*. “N.A.S.A”, pronounced as a single word, is a shorthand for the government agency “National Aeronautics and Space Administration”. For words which can’t be spoken as a single word (due to formation complexity) are spoken and written as hyphenated terms. “F.B.I.”, another government agency, is read as individual letters and stands for “Federal Bureau of Investigation”.

Conversion

Returning to our discussion of “Google”, the verb “google” as in “if you don’t know, *google* it” is a derived word which is distinct from its parent term. Conversion can involve the creation of new term(s) within a language, or a reordering of those terms (as in word order shift or repurposement).

This is distinct from Addition or Subtraction (semantically) since the word itself is new (has its own form:meaning pair), but there is an etymological tracking associated with it.

Conversion can also be inter-language. Loanwords such as “Schadenfreude” (from German, meaning to feel pleasure when witnessing the misfortune of others) or cliché (from French, meaning an overused phrase or idea lacking originality) involve converting terms across languages.

Finally, Conversion includes equivalence, or equating two forms through orthographic change or rebracketing. For example, colour → color is a change in form which becomes lexicalized (meaning the former usage is discarded entirely), or while we used to say “a napron”, we now say “an apron”.

Mutation Boundaries and Limiting the Scope of our Study

While these categories cover much of the language mutation, one disadvantage of reducing the number of features we’re considering is that there is bound to be overlap between the groupings. For this reason, we need to further stratify the dimensions of our categories. Let us return to our Signal map:

$$[\text{Signal}_{\text{method}}] \rightarrow \text{Projection} = \text{Judgment}$$

In order to model the effect of a mutation, we should control for as many variables as possible. We can control for *method* by only focusing on a particular domain of language (internet language), and we can control for *projection* by utilizing instances and focusing on derivative meanings. This leaves *signal* and “→” which is a stand-in for a dependency to generate a projection from the signal. We’ll call this dependency a “*transformation*”, which is simply the path required to *access* the projection.

In *Figure 6*, I provided a chart which illustrates the efficiency of a particular signal path (formalized in the equations). From here, we can limit the scope of mutable signals to an addition and subtraction in form which involves minimizing the complexity of language production and interpretation. Then

transformation will be manipulated by semantic creation and conversion (testing the ability to access particular projections from novel or derived signals).

In the next chapter, we will explore internet language along these dimensions and lay the groundwork for a study which will test the efficacy of these variables.

Chapter 4: Using Internet Language to Test Mutation Boundaries

English Internet Language

The term “bc” is a common Intraword Subtraction for the word “because”. When texting, it would be common to see the sentence “because I’m too busy” converted to “bc I’m too busy”. When processing the two sentences, we could argue for two possible paths: (A) the term “bc” is encoded as the word “because” or (B) the term bc has its own encoding with the same meaning as the term “because”.

Regardless of which solution is chosen, we can already discern that the production of the term “bc” is *faster* to generate than the term “because”, and the typing of “bc” is *faster* than typing “because”.

Notice, however, that we rarely go around saying “B.C. I’m busy” or “bic I’m busy”, illustrating that the method of delivery here is important. The clipping nature for this term seems to only exist in written forms, specifically in texting or online communication. Assuming that the projection for these terms is equivalent (there isn’t an alteration in underlying meaning), we can narrow our interpretation in the change of mapping down to the two pathways provided, which are localized in the transformation domain.

In the case of (A), we encode “bc” as a keyword which has a 1:1 access marking for “because”. In this solution, “bc” is not a novel generation, but merely a new form which adds a computational step in generating an extant form:meaning pair. This would generate an increase in the processing load of

accessing the meaning, but only slightly. In the case of (B), there is no increase in cognitive processing because “bc” is accessed directly as bc:meaning. However, we would have to store this new word, meaning there is an upfront cost of learning this new term, and a subsequent increase in the total storage of our lexicon (but again, only slightly).

From this single example, we can create a cascade of derivative contexts to test for the interpretability of the word “b” as a stand-in for “bc”. Consider the following progression:

“because I’m too busy”

“bc I’m too busy”

“b I’m too busy”

“b too busy”

“b busy”

“b”

When we see each of the transformations, it becomes easy to substitute the derived term for the lexicalized mutation, but what if no prior context was given? Determining how much context is necessary to assimilate the meaning between mutations can provide a threshold which can be used to track the extendability of novel language mutations along the dimension of subtraction.





This same principle can be applied to initialized or acronymized terms. “htk” is an acronym commonly used to mean “have to know”. We could then supply the term “wtk” to mean “want to know” and assess the interpretability of this phrase along varying contextual dimensions. In this case, there is a single letter substitution with a limited extension in meaning. Or we could subtract “htk” further into “ht” to generate “ht go to school”.

Instead of simply manipulating form, we can also focus on transformational changes in iconization and lexicalization. An important distinction between strictly mutable forms and the creation of a symbol, word, or phrase in Internet Language is in the level of analysis within which these mutations are processed. In the case of form subtraction, we reference a particular object (holding meaning constant), but many icons reference a set (or a room) wherein we use the term to *identify* the previous signal.

Take for example the emoji 🗨️ which stands in for the term “cap”, a word with etymological origins in African American English Vernacular, meaning “a lie” or untruthfulness. Ignoring for a moment the change in form, let’s focus on how a simple use of 🗨️ can change the meaning of a previously interpreted signal. Person A types “I’m six feet tall and made a million dollars last year.” Person B responds with “🗨️”. The functional use of the emoticon is to shift the way the meaning of the previous sentence is identified, specifically marking it as not truthful.

Returning to a previous example, 💀 a stand in for “dead”, can mean to find something very humorous. While one interpretation of the use of this term is to express one’s feelings, another way of viewing it is through the lens of identifying a different set of signals. For example, Person A types “What’s a string bean?” and Person B responds with “💀”. Person B’s response tells us how to properly identify or interpret Person A’s sentence.

In testing the boundaries for these terms, we can employ one of the following techniques: (1) produce a novel term (Creation) and increment the context until the interpretation is achieved, (2) create a derived term (Conversion) with the same meaning (Projection) and provide increasing context until the connection is made, (3) use a known term but change the meaning (Conversion), then supply increasing context until the new meaning is interpreted.

For example, the term “tea” is a colloquialism which refers to “gossip”. We often see the emoji  used as a standin for this term, which is already a hyphenation. A sentence such as “spill the ” means to “tell me the gossip”. When a response is given, the classification of that response will be within the category of “gossip” which is an identity marker for the signal. However, suppose we wanted to test the boundaries of this term. If Person A were to write “tell me the ”, would this response be understood? In order to make the connection, Person B would have to already have “tea” as part of their lexicon, then make the phonetic connection of the encoded “t-shirt” as “tee” which is a stand in for “tea”. We could perform similar tests with  as in tee-off, while reducing the form once again to “tee” which is converted to “tea”.

Chinese Internet Language

For the purposes of our study, it is important to expand the scope of the languages we assess in order to ensure that this is not a mono-language phenomenon. In the case of Chinese Internet Language, there are also a plethora of mutation examples, although they vary slightly from English due to the tonal nature of Chinese which allows for more phonetic word-play. For example, the *pinyin* (phonetic lettering less the tones) are the same for the words 妈 (ma, mother), 马 (ma, horse), 麻 (ma, numb), 骂 (ma, scold), and 吗 (ma, question marker). Therefore, we see many examples of one word having an association with another at the level of pinyin, irrespective of its tonal differences.

The clearest cultural example of this is presented by the number 4, written as “四” (si). In Chinese, 四 is pronounced with a falling tone, while 死 (si), the word for death, is pronounced with a falling-rising tone. Since both share a common syllable, the number 4 has become an unlucky number in China, and even more broadly (Japan also shares this cultural sense), to the extent that the number 4 is avoided in everyday life. For example, it is not uncommon for hotel buildings to not have a fourth floor, and for individuals to specifically exclude the number four from their license plates and phone numbers.

Other numerical Conversions include “520” which is a common stand in for the characters “我爱你” (wo-ai-ni), meaning “I love you”, and “666” or “六六六” (liu liu liu), which contains a syllabic connection between 六 and 流 (liu) which means smooth, a term which has been colloquialized to identify a “smooth” action or someone who is “smooth” (similar to the English usage). Therefore, 六六六 is interpreted to mean someone who is skilled or can handle something effortlessly. However, there is also a phonetic similarity between 六 and 牛 (niu) which literally means “ox” but has a colloquial meaning of “awesome” or “powerful”. This chaining of meaning further reinforces the metaphor of 六, which is then reduplicated for an intensity effect → 六六六.

Since there isn’t a lettering system in Chinese (*pinyin* is only used to track phonetics for learning the language), tracking subtraction is different and requires us to include elements like numbers, English letter additions, and phone-character deletions. For example, 哥哥 (gege) in Chinese means older brother, but colloquially, a single 哥 (ge) is used to refer to a “bro’ or a male friend. This term can then be added to the phrase “我们都哥们” (women dou ge men) which means “we are all brothers” and is used to extend the familial classification to the others in a party who may be having a heated exchange, in order to pacify the situation.

While 哥 has been a word in the Chinese lexicon for millenia, terms like 比萨 (bi sa) to mean “pizza” and 咖啡 (ka fei) to mean “coffee” are clear examples of English loanwords which have come about as our cultures and languages have become more intertwined. Sometimes, an English word will directly mutate into a Chinese phrase, like the combination 笑cry (xiao cry), which uses the term 笑 “to laugh” in combination with the English word “cry” to mean laugh-crying, which has become so popular it has an emoji association: 😂.

Another unique feature of Chinese is its pictographic form. Many of the original Chinese characters (especially the radicals) were meant to be pictorial representations of the meanings they meant to convey. For example “人” (ren) means “person”, and 火 (huo) means “fire”. One of these historical

scripts is 囧 (jiong) which meant “bright” or “light shining through a window”. It’s use was out of vogue for the longest time until the early 2000’s when it was revived to mean “embarrassed”, due to its resemblance to an embarrassed face. The use of 囧 now used to convey a range of awkward or negative emotions, further extended to 囧事 (jiong shi, shi meaning thing, situation, or circumstance) to mean an awkward situation. In this way, Chinese leverages its pictographic form in many of the same ways as emojis to display certain expressions and emotions, making creation and conversion utilization more versatile.

Chapter 5: Study Proposal

The Testing Model

In Chapter 3, I described the importance of limiting the scope of our study. In order to do this, we will focus on a limited array of variables which are sufficient to assess the type of mutation taking place. Then by incorporating examples of these different types into a study, we can gather data on the interpretability of novel internet language terms along these dimensions. The variables will be defined as follows:

- **Novelty (X_1):** A One Hot Encoded binary value indicating whether the mutation is a completely synthesized form:meaning pair [0, 1]. A value of “1” indicates the mutation is novel.
- **Familiarity (X_2):** A One Hot Encoded binary value indicating whether the tester is familiar with the parent term which the mutation is based off of [0, 1]. If the tester is not familiar with the parent term (“0”), Novelty changes to “1”.
- **Is_Letters (X_3):** A One Hot Encoded binary value indicating whether the mutation is comprised of letters [0 or 1]. A value of “0” indicates that the mutation is an icon.
- **Subtraction (X_4):** A continuous value ranging from (0, 1) where the value represents the percentage of letters subtracted from the parent term as a decimal.

- **Addition (X_5):** A continuous value ranging from (0, 1) where the value represents the percentage of letters added from the parent term as a decimal.
- **Word Count (X_6):** A discrete value from [1, n] where “n” represents the number of words that are represented by the parent term (in the case of acronyms). A value of “1” represents a single word (not an acronym).
- **Is_Parent (X_7):** A One Hot Encoded binary value indicating whether the mutated term has the same form as the parent term [0, 1]. A value of “1” indicates they have the same form.
- **Conversion (X_8):** A One Hot Encoded binary value indicating whether the mutated term has a different meaning from the parent term [0, 1]. A value of “1” indicates they have a different meaning.
- **Interpretability (Y):** A continuous value ranging from (0, 1) where the value represents the ability of the tester to apprehend the meaning of the mutated term (1) multiplied by a decimal value based on the amount of context needed for the tester to interpret the meaning. A value of “0” means the tester was unable to apprehend the meaning of the mutated term.

The Study

I will enlist 250 college students into the study, separated into five groups. The study will be conducted online, on a website created specifically for the purpose of administering the test. Each test will contain 24 questions, the first three of which will be control questions which familiarize the subject with the testing process. Prior to administering the control questions, I will provide a familiarity test in the following format:

Have you seen “gtg” before?

I will only administer control questions with which the tester is familiar, ensuring data regularity and consistency. Here are two control examples:

Question 1:

1. What is the meaning of “gtg”?
 - a. The user will be provided a text-box to type a response
2. What is the meaning of “gtg” in the following context? “I gtg”
 - a. User-response
3. What is the meaning of “gtg” in the following context? “Sorry, but I gtg”
 - a. User-response
4. What is the meaning of “gtg” in the following exchange:
 - a. Can you stay on?
 - b. “Sorry, but I gtg”
 - i. User-response

Question 2:

1. What is the meaning of 🗨️ ?
 - a. User-response
2. What is the meaning of 🗨️ is the following context? That’s 🗨️
 - a. User-response
3. What is the meaning of 🗨️ is the following context? That’s 🗨️ bro
 - a. User-response
4. What is the meaning of 🗨️ is the following exchange:
 - a. I’m six feet tall
 - b. That’s 🗨️ bro
 - i. User-response

Notice that I begin with only the signal, and I progressively add context to steer the tester toward the intended meaning. This progression will begin with the (Signal) alone, then the (Signal + word), then (Signal + sentence), then (Signal + sentence) as a response to a sentence. By scaffolding the questions this way, I can use the amount of context needed to interpret the term's meaning as a weight to more effectively calculate the interpretability of a particular mutation.

After the control test, I will provide another familiarity test for parent terms of the mutations which the user will see in the study. They will provide the stimulus and a text box for a response. Their responses will toggle the "Familiarity" feature, and a result of "0" will toggle the "Novelty" feature to "1". No feedback will be provided to the tester about the accuracy of their answers, as this could skew testing results.

When scoring the test results, letter mutations will have an exact answer. In the case of "gtg", the answer will be "got to go" (without case sensitivity, meaning "Got To Go" will also be accepted as correct). In calculating the Interpretability, the following weights will be applied:

Level 1 (Signal only): $(1 * (\% \text{ of letters correct in sequential order}))$

Level 2 (Signal + word): $(0.7 * (\% \text{ of letters correct in sequential order}))$

Level 3 (Signal + sentence): $(0.4 * (\% \text{ of letters correct in sequential order}))$

Level 4 (Signal + sentence as response to a sentence): $(0.1 (\% \text{ of letters correct in sequential order}))$

When the tester gets 100% of the letters in the correct order (a perfect response), the following levels will be excluded (this will prevent issues where more context might reduce the user's ability to interpret the mutation's meaning, reducing error around malformed contextualization). The results will then average the score across levels and compute a value between 0 and 1.

In the case of icon mutations, each answer will have an array of keywords which model the sentiment of the interpretation. A threshold of the keywords (or synonyms) will be required for a 100% correct response. The scoring will operate under the same calculation method as the letter mutations.

The purpose of organizing the testers into five groups is to allow for variability within the testing parameters (the stimulus supplied across the four levels). This means that, given the same term, each group will be provided slightly different contexts. This will reduce some of the noise around malformed contexts and normalize the scoring results. For example, if we take the emoji 🧮 to mean “calculated”, a positive affirmation used to say that a successful action was purposeful, we could scaffold the questions with the following progressions:

🧮 → 🧮ed → it was 🧮ed → 🧮 → S: “How did you know that would happen?”, A: it was 🧮ed
 🧮 → 🧮ed → I 🧮ed → S: “How did you do that?”, A: I 🧮ed

At the end of each test, the results will be stored in a database alongside the calculated score for each question. I will then run the results through several supervised Machine Learning models to assess the efficacy of the underlying feature space to predict the target.

Conclusion

The evolution of language, as examined through the frameworks of generative grammar, construction grammar, and probabilistic models, reveals its intricate interplay between form, meaning, and context. Language mutation is not merely a reflection of linguistic creativity but a functional response to the ever-changing demands of communication, particularly in the digital age. By systematically analyzing changes such as semantic inversion, clipping, iconization, and borrowing, this research underscores the adaptive capacity of language to minimize complexity while maximizing efficiency.

The study of internet language highlights the dynamic processes of creation, addition, subtraction, and conversion, illustrating how new linguistic forms emerge and gain acceptance within specific identity spaces. This adaptability, though most evident in online communication, is a universal feature of language evolution across cultures and mediums, as evidenced by comparisons between English and Chinese internet slang.

As we advance into an era dominated by artificial intelligence and natural language processing, understanding these mutations is critical for bridging the gap between the form-centric capabilities of machines and the meaning-driven nature of human communication. The proposed study model, with its focus on testing interpretability and efficiency across mutation boundaries, provides a foundation for exploring how language innovations are assimilated and propagated.

Ultimately, this research not only contributes to the linguistic and computational understanding of language change but also opens avenues for designing systems that better accommodate the nuances of human interaction. By leveraging the principle of complexity minimization, we can further our grasp of how language evolves to serve its fundamental purpose—effective and meaningful communication in an ever-complex world.

Sources

- Chomsky, N. (2002). *Syntactic structures*. Mouton de Gruyter.
- Chomsky, N. (1980). A review of BF Skinner's Verbal Behavior. *The Language and Thought Series*, 48-64.
- Chomsky, N. (1977). On wh-movement.
- Fridman, L., & Gibson, E. (2024, April 17). *Edward Gibson: Human Language, Psycholinguistics, Syntax, Grammar & LLMs | Lex Fridman Podcast #426*. YouTube.
- Szabolcsi, A., & Sag, I. (Eds.). (1992). *Lexical matters* (No. 24). Center for the Study of Language (CSLI).
- Wittgenstein, Ludwig, and Hacker P M.S. *Philosophische Untersuchungen = philosophical investigations*. Oxford u.a.: Wiley-Blackwell, 2010.
- Norretranders, T. (1999). *The user illusion: Cutting consciousness down to size*. Penguin.
- Tesnière, L. (2015). *Elements of structural syntax*. John Benjamins Publishing Company.
- Newmeyer, F. J. (2000). *Language form and language function*. MIT press.
- Goldberg, A., & Suttle, L. (2010). Construction grammar. *Wiley Interdisciplinary Reviews: Cognitive Science*, 1(4), 468-477.
- Michaelis, L. A. (2006). Construction grammar. *The encyclopedia of language and linguistics*, 3, 73-84.
- Ungerer, T., & Hartmann, S. (2023). *Constructionist approaches: Past, present, future*. Cambridge University Press.
- Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., ... & Xie, X. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3), 1-45.
- Manning, C. D. (2015). The case for universal dependencies. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)* (Vol. 1).
- Fromkin, V., Rodman, R., & Hyams, N. M. (2021). *An Introduction to Language*. Cengage.
- Chandler, D. (2022). *Semiotics: the basics*. Routledge.
- Lin, G., & An, F. (2014). The Diachronic Development and Adaptable Interpretation of the Internet.