Syntax and regular languages

February 15, 2012

Some facts about English

It is widely claimed that English speakers find the following set of sentences acceptable:

1. The man who insisted John ate a sandwich was arriving today
2. The man who insisted the woman who insisted John ate a sandwich was leaving yesterday was arriving today
3. The man who insisted the woman who insisted the person who insisted John ate a sandwich was on trial was leaving yesterday was arriving today
4. ...

In particular, the set of sentences acceptable to English speakers is a superset of $mDep$, the language of the following pseudo-regular expression:

$$(1) \ (\text{the:} (\text{man + woman + person}):\text{who:insisted:})^m \text{John:ate:a:sandwich:(was:(((\text{on:trial}) + (\text{arriving:today}) + (\text{leaving:yesterday}))})^m}$$

On the other hand, it seems that English speakers find the following set of sentences unacceptable:

1. *The men who insisted John ate a sandwich was arriving today
2. *The men who insisted the women who insisted John ate a sandwich is leaving yesterday was arriving today
3. *The men who insisted the women who insisted the people who insisted John ate a sandwich is on trial is leaving yesterday was arriving today
4. ...

The set of sentences unacceptable to English speakers is a superset of $mDep'$, the language of the following pseudo-regular expression:

$$(2) \ (\text{the:} (\text{men + women + people}):\text{who:insisted:})^m \text{John:ate:a:sandwich:(was:((\text{on:trial}) + (\text{arriving:today}) + (\text{leaving:yesterday}))})^m}$$

It also seems that English speakers find the following set of sentences unacceptable:

1. *The man who insisted insisted John ate a sandwich was arriving today
2. *The man who insisted who insisted John ate a sandwich was arriving today
3. *The man who insisted man who insisted John ate a sandwich was arriving today
4. *The man who insisted the man who insisted John ate a sandwich was arriving today

5. …

The set of sentences unacceptable to English speakers is a superset of this language, because it is a superset of the slightly larger language \( mDep_{+1} \), the language of the following pseudo-regular expression:

\[
\begin{align*}
&m^p \\
&\left( \text{the:}(\text{man + woman + person}):\text{who:insisted}) \right)^m \\
&\left( \text{the:}(\text{man + woman + person}):\text{who:insisted} \\
&\text{+ (man + woman + person):who:insisted} \\
&\text{+ who:insisted} \\
&\text{+ :insisted} \\
&\text{John:ate:a:sandwich:(was:((on:trial) + (arriving:today)) + (leaving:yesterday))} \\
\end{align*}
\]

And English speakers find the following set of sentences unacceptable:

1. *The man who insisted the woman who insisted insisted the woman who insisted John ate a sandwich was leaving yesterday was arriving today

2. *The man who insisted the woman who insisted who insisted the woman who insisted John ate a sandwich was leaving yesterday was arriving today

3. …

The set of sentences unacceptable to English speakers is a superset of this language, because it is a superset of the slightly larger language \( mDep_{+2} \), the language of the following pseudo-regular expression:

\[
\begin{align*}
&m^p \\
&\left( \text{the:}(\text{man + woman + person}):\text{who:insisted}) \right)^m \\
&\left( \text{the:}(\text{man + woman + person}):\text{who:insisted} \\
&\text{+ (man + woman + person):who:insisted} \\
&\text{+ who:insisted} \\
&\text{+ :insisted} \\
&\text{John:ate:a:sandwich:(was:((on:trial) + (arriving:today)) + (leaving:yesterday))} \\
\end{align*}
\]

And we could go on to construct \( mDep_{+3} \), all the way up to \( mDep_{+m} \).

Some generalizations, in words:

- Sentences can be embedded in the “center” of other sentences

- To be licit, a sentence must have a subject that agrees with the verb in number, and this holds even when there are sentences center-embedded in that sentence, to the effect that each sentence must match a subject with a verb in a particular position

- To be licit, a sentence must match one subject with one verb; the tail end of the subject cannot be repeated freely; again, this holds even when there are sentences center-embedded

This last point might give you an idea: it seems connected to the pumping lemma. The second point is also interesting, but it is not the same as what the pumping lemma says; it represents an idea from Chomsky (1956) which presages a more general condition on regular languages first published in Nerode (1958), now called the Myhill-Nerode theorem.
Using the pumping lemma

Recall the pumping lemma:

**Definition 1.** Given a regular language \( L \), we say string \( w \in L \) contains a substring \( y \) that can be **pumped in** \( L \) if we can decompose \( w \) into \( xyz \) such that:

(i) \( w = xyz \)

(ii) \( y \neq \varepsilon \)

(iii) For all \( i \geq 0 \), \( xy^iz \in L \)

**Theorem 2.** (Pumping lemma for regular languages.) For every regular language \( L \), there is some maximum length \( p \), a pumping-lemma constant, beyond which any string \( w \in L \) with \( |w| > p \) will have a substring \( y \) that can be pumped in \( L \). Furthermore, if \( p \) is a pumping-lemma constant such that \( y \) can be pumped, then \( y \) can always be found within the first \( p \) symbols of \( w \): if \( w = xyz \), then \( |xy| \leq p \).

**Example 3.** Prove that \( a^n b^n \) is not a regular language.

**Proof.** Suppose the language were regular. Then there would be some pumping lemma constant \( p \). Surely \( a^p b^p \in L \). The pumping lemma tells us that there is a prefix of \( a^p b^p \) of length \( \leq p \), part of which can be pumped in \( L \). But since any prefix of \( a^p b^p \) of length \( \leq p \) must consist entirely of \( a \)’s, pumping any substring of length \( k \) would imply that \( a^{p+k} b^p \in L \), \( a^{p+2k} b^p \in L \), and so on. These are not in the language by definition, and so \( a^n b^n \) cannot be a regular language.

**Claim 4.** No language that contains \( mDep \) but excludes the set \( \bigcup_{i=1}^{m} mDep_{+i} \) is regular.

**Proof.** Suppose there exists such a regular language, \( L \). Then there is some pumping lemma constant \( 4p \). Then \((\text{the:man:who:insisted:})^p \text{John:ate:a:sandwich:(was:leaving:yesterday)}^p \) is a string of length \( 4p + 4 + 3p > 4p \) which is in the language, since it is an element of \( mDep \). Let \( xy = (\text{the:man:who:insisted:})^p \).

By the pumping lemma, there is some choice of \( y \) such that, for all \( k \geq 0 \):

\[ xy^k \text{John:ate:a:sandwich:(was:leaving:yesterday)}^p \in L \]

We will show that, for any suffix \( y \neq \varepsilon \) of \((\text{the:man:who:insisted:})^p\), the string

\[(\text{the:man:who:insisted:})^p y \text{John:ate:a:sandwich:(was:leaving:yesterday)}^p \notin L \]

This will falsify the hypothesis for \( k = 2 \).

Let \( n = \left\lceil \frac{|y|}{4} \right\rceil \), the smallest integer such that \( n \geq \frac{|y|}{4} \). Any suffix of \((\text{the:man:who:insisted:})^p\) must be of the form \((\text{the:man:who:insisted:} + \text{man:who:insisted:} + \text{who:insisted:} + \text{insisted:})(\text{the:man:who:insisted:})^{n-1}\). Given that any suffix of \( xy \) with a particular \( n \) will be of this form, \( xy y \text{John:ate:a:sandwich:(was:leaving:yesterday)}^p \) is in \( mDep_{+n} \) (for \( m = p \)), by construction of \( mDep_{+i} \). But by construction no such string is in \( L \), and we have a contradiction. Thus \( L \) is not regular by the pumping lemma.

We may verify this last claim about the form of a suffix of \( xy \) by explicitly constructing a function \( S_{xy}(i) \) taking indices to elements of the string \( xy \):

\[
S_{xy}(i) \overset{\Delta}{=} T(\text{mod}(i - 1, 4)), 1 \leq i \leq 4p
\]

\[
T(j) \overset{\Delta}{=} \begin{cases} 
\text{the} & j = 0 \\
\text{man} & j = 1 \\
\text{who} & j = 2 \\
\text{insisted} & j = 3
\end{cases}
\]
By induction on \( n \); in the base case, \( n = 1 \), and \( 1 \leq y \leq 4 \). We know that \(|xy|\) is divisible by 4; thus if \(|y| = 1\), then \(S_y(1) = S_{xy}(|xy|) = T(3) = \text{insisted.} \) by the definition of a suffix; if \(|y| = 2\), then \(S_y(2) = \text{insisted,} \) for the same reason, and \(S_y(1) = S_{xy}(|xy| - 1) = T(2) = \text{who,} \) and \( y = \text{who:insisted;} \) similarly, if \(|y| = 3\), then \( y = \text{man:who:insisted,} \) and, if \(|y| = 4\), then \(y = \text{the:man:who:insisted.} \) Thus the claim holds for \( n = 1 \). Suppose the claim holds for \( n = l + 1 \). Let \( y = vz \), where \(|z| = 4l\). Then \( z = (\text{the:man:who:insisted:} + \text{man:who:insisted:} + \text{who:insisted:} + \text{insisted:})(\text{the:man:who:insisted:})^{l-1} \) by the inductive hypothesis, since \( z \) is a suffix of \((\text{the:man:who:insisted:})^l \); furthermore, since \(|z| = 4l\), \( z = (\text{the:man:who:insisted:})^l \). Thus we know the value of \( S_y(i) \) for \(|y| + 1 \leq i \leq |y| \). If \(|y| = 1\), then \(S_y(1) = T(3) = \text{insisted} \) (if \( j \) is the index of \( S_y(1) \) in \( xy \), then we know that \( S_y(j+1) = T(0) \), because \(|z|\) is divisible by 4, and thus \( S_{xy}(j) = T(3) \)). If \(|y| = 2\), then \(S_y(2) = \text{insisted,} \) for the same reason, and \(S_y(1) = \text{who,} \) and so on for \(|y| = 3 \) and \(|y| = 4 \) as above. But then the claim holds for \( n = l + 1 \). Thus any suffix of \( xy \) with a particular \( n \) will be of the form \((\text{the:man:who:insisted:} + \text{man:who:insisted:} + \text{who:insisted:} + \text{insisted:})(\text{the:man:who:insisted:})^{n-1} \). □

Using dependencies

There is another way of using center-embedded sentences to show that the set of acceptable English sentences is not a regular language. From Chomsky 1956:

Suppose that \( A \) is the alphabet of a language \( L \), that \( a_1, \ldots, a_n \) are symbols of this alphabet, and that \( S = x_1a_1x_2a_2\cdots x_m a_m b_1 y_1 b_2 y_2 \cdots b_m y_m \) is a sentence of \( L \). We say that \( S \) has an \( m \)-dependency with respect to \( L \) if there is a unique permutation \( \alpha \) of \((1, \ldots, m)\) meeting this condition: there are \( c_1, \ldots, c_{2m} \in A \) such that for each subsequence \((i_1, \ldots, i_p)\) of \((1, \ldots, m)\), \( S_1 \) is not a sentence of \( L \) and \( S_2 \) is a sentence of \( L \), where \( S_1 \) is formed by substituting \( c_{i_j} \) for \( a_{i_j} \) in \( S \), for each \( j \leq p \); \( S_2 \) is formed by substituting \( c_{m+\alpha(i_j)} \) for \( b\alpha(i_j) \) by \( c_{m+\alpha(i_j)} \) in \( S_1 \), for each \( j \leq p \). Thus replacement of \( a_i \) by \( c_i \) in \( S \) requires, for well-formedness, a corresponding replacement of \( b\alpha(i) \) by \( c_{m+\alpha(i)} \) (this notion can be generalized in obvious ways).

This is complicated, but the idea is simple:

**Definition 5.** Let \( w \in L \). Consider \( m \) symbols \( a_1, \ldots, a_m \) and \( m \) symbols \( b_1, \ldots, b_m \) so that \( w = xy \), \( x = \ldots a_1 \ldots a_2 \ldots \ldots a_m \ldots \), \( y = \ldots b_1 \ldots b_2 \ldots \ldots b_m \ldots \) (that is, all \( a_i \)’s precede all \( b_i \)’s, but \( b_1, \ldots, b_m \) do not necessarily appear in that order in \( w \)). We say that \( w \) has an \( m \)-dependency in \( L \) if there is at least one selection of symbols in \( w \), and exactly one indexation of the \( b_i \)’s, such that, for some other symbols \( a_1', \ldots, a_m' \) and \( b_1', \ldots, b_m' \), replacing any non-empty subset of the \( a_i \)’s by the corresponding symbols \( a_i' \) results in a string which is not in the language, while replacing those same \( a_i \)’s by \( a_i' \) and replacing the corresponding \( b_i \)’s by \( b_i' \) would result in a string which is in the language.

**Example 6.** To see how this works, consider our example above: the sentence *the man who insisted the woman who insisted John ate a sandwich was leaving yesterday was arriving today* has a 2-dependency. Consider the following selection and indexation of symbols:

the man, who insisted the woman, who insisted John ate a sandwich was, leaving yesterday was, arriving today

If we replace *man* with *men* and *woman* with *women* results in a member of the set of unacceptable strings, but if we replace both instances of *was* with *were*, then the string again falls into the set of acceptable strings; if we replace only *man* with *men*, then the string is unacceptable, but if we replace the *was* with *were*, then the string again falls into the set of acceptable strings; similarly for *woman*. Furthermore, this indexation is unique for this selection of underlined symbols. Consider:
It is still true that if we replace both man and woman, then replace both instances of was, the string is again in the acceptable set; however, if we replace only man, then we cannot make the string again grammatical by replacing was with were; in more usual terms, that first was agrees with woman, not man. Thus the original indexation is the only indexation of the underlined symbols for which the symbols in the second part of the sentence have dependencies with the corresponding symbols in the first part of the sentence, and thus the sentence has a 2-dependency.

**Theorem 7.** If \( w \in L \) has an \( m \)-dependency in \( L \), then any automaton \( A \) with \( L(A) = L \) has at least \( 2^m \) states.

**Proof.** Suppose \( w \) has an \( m \)-dependency in \( L \) corresponding to the following decomposition: \( w = xzy, x = x_1 a_1 \cdots x_m a_m, y = y_1 b_1 \cdots y_j b_j. \) There are \( 2^m \) prefixes \( x_i \) that can be formed by replacing some subset \( s \) of the \( a_i \)'s with the corresponding \( a'_i \)'s. We show that each such prefix must lead to a different state, by contradiction. Suppose two prefixes \( x_s \) and \( x_t \) constructed in this way, for two different subsets \( s \) and \( t \), lead to the same state \( q_{st} \). There must be some index \( 1 \leq j \leq |x| \) such that \( S_{x_s}(j) = a_k \) and \( S_{x_t}(j) = a'_k \), (or vice versa), for some \( i \). Now construct \( y_s \) and \( y_t \) by replacing the corresponding subsets of the \( b_i \)'s with \( b'_i \)'s. Then, by construction of \( w \), \( x_t y_s \) must lead to a non-accepting state, since it contains \( b_k \) rather than \( b'_k \), while \( x_t y_t \) must lead to an accepting state. But since \( \delta(q_0, x_s) = \delta(q_0, x_t) = q_{st} \), this means that \( \delta(q_{st}, y_s) \) must lead to both an accepting state and a non-accepting state; this is a contradiction; thus we conclude that each prefix constructed by replacing a subset of the \( a_i \)'s with the corresponding \( a'_i \)'s leads to a distinct state; thus the automaton must have at least \( 2^m \) states.

**Corollary 8.** If \( L \) is a regular language, then there is some \( n \) such that \( L \) has no strings with \( m \)-dependencies in \( L \) for \( m > n \).

**Proof.** Let \( A \) be an automaton for \( L \) with no more than \( 2^k \) states; if \( w \in L \) had an \( m \)-dependency in \( L \) for \( m > k \), then \( A \) would have at least \( 2^m \) states; thus by contradiction \( L \) has no strings with \( m \)-dependencies in \( L \) for \( m > k \).

---

**The Myhill-Nerode theorem**

Recall that last time we introduced the idea of equivalence of states in a DFSA:

**Definition 9.** Two states \( p, q \) of some DFSA \( A \) are **equivalent** if they share the same set of acceptable suffixes.

We now construct a corresponding notion for strings. First we must a concept parallel to acceptable suffixes:

**Definition 10.** Given some string \( x \), the set of **acceptable extensions of \( x \) in \( L \)**, \( E_L(x) \), is the set of all strings \( z \) such that \( xz \in L \).

Now we can proceed to use this concept to give a notion of equivalence with respect to a particular language:

**Definition 11.** Two strings \( x \) and \( y \) are **\( L \)-equivalent** if \( E_L(x) = E_L(y) \).

**Theorem 12.** (Myhill-Nerode theorem). A set of strings \( L \) is a regular language if and only if it partitions \( \Sigma^* \) into finitely many sets of \( L \)-equivalent strings ("\( L \)-equivalence classes").

**Proof.** “Only if” similar to the proof above: go through the sentences of the language—when we add a sentence that needs to be distinguished from all the other sentences we’ve seen up to now, we also need to add a state to handle distinguishing it from the others; idea behind “if”: add one state for each \( L \)-equivalence class. □
Example 13. $a^*b^*$ yields exactly three $L$-equivalence classes: $a^*$ (these can be extended by any string in $a^*b^*$); $a^*bb^*$ (these can be extended by any string in $b^*$); and $\Sigma^* - a^* - a^*bb^*$ (these cannot be extended in the language). Since these obviously form a partition, to show that the language is regular, it is sufficient to show $L$-equivalence within each of these sets (i.e., that there is no distinguishing extension within any set) and lack of $L$-equivalence across these sets (technically this last point is superfluous, since if we show equivalence within sets, then surely there could be no more than three $L$-equivalence classes, thus finitely many).

On the other hand:

Example 14. The set of acceptable English sentences has infinitely many equivalence classes, because there is a different equivalence class for each level of center-embedding, i.e., there is a distinguishing extension for each string the man who insisted the man who insisted . . . John ate a sandwich, namely an agreement error.

Some discussion

The topic of discussion:

1. Every normal human’s mind must, in one way or another, be capable of computing a function partitioning the set of possible human language strings, distinguishing one subset (the “grammatical” strings of a language) from another set of human language strings (the “ungrammatical” strings of that language).

2. Question: do we know anything about what kind of function this could be?

Proposal:

1. The characteristic function implicitly computed by a mental grammar on sentences can always be characterized by a finite-state automaton

Counterargument:

1. There are some humans whose mental grammar characteristic functions compute sets with unbounded nested dependencies. These functions cannot be characterized as finite-state automata. Thus the claim is false.

Today, there is general agreement that this is correct. But, in the sixties, there were debates about this. Here is the main counterargument that was put forth at the time:

1. I call bullshit. Whoever told you English speakers can distinguish matched from mismatched numbers of NP/VP pairs in nested dependencies was lying. Thus you have no evidence that there are non-finite-state mental grammars.

This is an important wrinkle. We will investigate this in just a minute.

Some possible counterarguments:

1. I reject your facts. English speakers find these sentences confusing, but they can distinguish grammatical from ungrammatical, if given enough time.

Problem with this argument: if you are right about this, doesn’t this pose serious problems? Aren’t there a lot of weird sentences that we could get people to accept or reject if we gave them enough time?

Or (a more nuanced version):
1. I accept your facts: in particular, I accept that, beyond two or three dependencies, English speakers cannot reliably distinguish grammatical from ungrammatical sentences. However, this may not give us evidence about the mental grammars of English speakers, because there may be independent reasons that English speakers fail to distinguish these, such as memory limitations.

Problem with this argument: if you are right about this, doesn’t this pose even more serious problems? How will we ever distinguish memory limitations from grammatical limitations? Is there even a distinction?

Or:

1. I accept your facts: in particular, I accept that, beyond two or three dependencies, English speakers cannot reliably distinguish grammatical from ungrammatical sentences. However, this point is moot, because there exist other languages with nested dependencies where speakers can reliably distinguish grammatical from ungrammatical nested dependency sentences, with no limit on the number of dependencies.

This is also important. We will investigate this in just a minute.

Or:

1. I accept your facts: in particular, I accept that, beyond two or three dependencies, English speakers cannot reliably distinguish grammatical from ungrammatical sentences. However, this point is moot, because there exist other languages with cross-serial dependencies where speakers can reliably distinguish grammatical from ungrammatical nested dependency sentences, with no limit on the number of dependencies.

This is also important. We will investigate this in just a minute.

**The missing VP effect**

The missing VP effect is an effect that undermines the claim that requirement that $mDep_{+n}$ is ungrammatical for all $n$. Gibson and Thomas (1999) ran an experiment in which they presented participants with sentences like the following, and asked them to read them only once:

*(all three VPs)* The ancient manuscript that the graduate student who the new card catalog had confused a great deal was studying in the library was missing a page.

*(missing VP)$_3$* The ancient manuscript that the graduate student who the new card catalog had confused a great deal was studying in the library.

The idea was to confirm the intuition that you lose track of the dependencies to the point of finding “pumped” sentences acceptable.

[[They also included sentences like this, which you can think about in your free time:

*(missing VP)$_1$* The ancient manuscript that the graduate student who the new card catalog was studying in the library was missing a page.

*(missing VP)$_2$* The ancient manuscript that the graduate student who the new card catalog had confused a great deal was missing a page.]]

They then asked subjects to rate the sentences on a scale of 1 to 5 on how easy they were to understand (1: easy; 5: hard). Results:
The important thing here is that, at least when given the task of answering which sentences sound good and bad, speakers seem to be insensitive to dependencies above a certain number. This confirms the original counterargument.

Now recall one of the counter-counterarguments was to claim that this was an idiosyncrasy of English. Well, Standard German also has nested dependencies:

1. Der Anwalt, den der Zeuge, den der Spion betrachtete, schnitt, überzeugte den Richter.  
The lawyer that the witness that the spy looked at cut convinced the judge

*The lawyer that the witness that the spy looked at convinced the judge

It is claimed, however, that German speakers do not have the missing VP effect. Vasishth et al. (2010) did several experiments testing this. They had English speakers and German speakers read English and German sentences corresponding to 1 and 2. They used the fact that people tend to slow down when reading something weird to examine whether speakers found (1) or (2) weirder. In English, the result was consistent with the missing VP effect.

The dotted line is the (2) sentence, i.e. the sentence we are saying is ungrammatical. It seems that speakers slow down more when they read the grammatical sentence! Given the results of many other studies, this strongly suggests an illusion of grammaticality in the ungrammatical sentences. On the other hand, here is the corresponding plot for German readers:
The fact seems to be that German speakers do not have the missing VP effect. The evidence suggests that, at a minimum, German speakers are better at noticing the difference between matched and mismatched nested dependency sentences than English speakers (if not perfect without bound).

The authors’ explanation for this was as follows: in embedded clauses, the German default word order is SOV, not SVO. This means that German speakers are used to having to wait around to hear/read the verb. So they are just better at this than English speakers. We might infer that, for English speakers, the problem is a memory problem, not a grammatical limitation. Of course, we might also simply conclude that, while there could be a grammatical limitation in English, the point is moot for the universal claim, because there are languages (like Standard German) which do not suffer from the “missing VP” problem.

Cross-serial dependencies

We have seen nested dependencies, and we have seen that languages with nested dependencies are not regular. Now let’s look at another type of dependency: the cross-serial dependency. From Shieber (1985), here are some examples of Swiss German subordinate clauses:

1. Jan säit das mer em Hans es huus halfed aastriiche
   Jan says that we Hans.DAT house.ACC.DEF helped paint
   Jan says that we helped Hans paint the house.

2. Jan säit das mer de Hans es huus lönd aastriiche
   Jan says that we Hans.ACC house.ACC.DEF let paint
   Jan says that we let Hans paint the house.

Swiss German subordinate clauses have a very particular word order. To see this, notice that, in both sentences, the paint clause is embedded in another clause, either helped or let. But help and let have idiosyncratic case-marking properties, such that the person you are helping to do something gets case-marked with dative (em) and the person you are letting do something gets case-marked with accusative (de). Similarly, paint requires
accusative marking on the thing being painted (es) rather than dative (em). It is very simple to see that, if we let these go, we will have unbounded dependencies. But, in this case, they are in the other order: rather than \( x_1a_1 \ldots x_m a_m z_y b_m \ldots y_1 b_1 y_0 \), we have \( x_1a_1 \ldots x_m a_m z_y b_1 \ldots y_m b_m y_0 \). Here are some ungrammatical sentences:

1. *Jan säit das mer em Hans es huus lönd aastriiche
   Jan says that we Hans.DAT house.ACC.DEF let paint
2. *Jan säit das mer de Hans em huus lönd aastriiche
   Jan says that we Hans.ACC house.DAT.DEF let paint

Shieber claims that Swiss German speakers are fine with unboundedly many such dependencies:

1. Jan säit das mer d’chind em Hans es huus lönd hälfed aastriiche
   Jan says that we children.ACC.DEF Hans.DAT house.ACC.DEF let help paint
2. *Jan säit das mer d’chind em Hans es huus lönd hälfed aastriiche
   Jan says that we children.ACC.DEF Hans.DAT house.ACC.DEF let help paint

However, Bach et al. (1986) investigated this, and found that speakers of Dutch (a language that also has cross-serial dependencies in the same construction) rated these sentences as very difficult to understand beyond one embedding. Thus this data is not really pertinent in the way that is sometimes suggested.

**Some other data**

This data from English compound formation is sometimes put forth as relevant. You can decide for yourself!

* anti-missile missile
  anti-anti-missile missile missile
  anti-anti-anti-missile missile missile missile
  . . .

**Current consensus**

Syntactic patterns in natural language (that is, facts about which arrangements of words are acceptable, and which others are not) are not generally functions that can be computed by finite-state automata. However, there are limitations on what speakers can process that suggest that the cognitive mechanisms we use to understand sentences may limit our ability to get full use out of our grammars. This must be true anyway: you could never fully understand a sentence with six embeddings, nested/cross-serial or not. There is clearly more to know here, but probably the most important fact influencing this conclusion has been the (apparent) fact that it would be awfully tricky to expect kids learning English to come to the right generalizations about how English works based only on what they hear in the first few years of life. It would mean that they would learn that they cannot say:

1. The man who insisted that the woman who insisted that the boy who insisted that John ate a sandwich was on time was running late was trying to catch up with both of them
But they can say:

1. This is the man who loves the woman who hates the boy who insisted that John ate a sandwich

2. …

It seems children would need to come to the complex generalization that relative clauses can be arbitrarily nested in object position, but not in subject position. But presumably relativization is a fact about NPs, not a fact about the environments in which those NPs appear. Thus such a theory would require forbiddingly complex grammars. Thus, although there are some caveats that we need to consider when looking at the data, it is generally understood that syntactic patterns are not regular.

References


