The nature of linguistic structure building

Terry Langendoen
Department of Linguistics
University of Arizona

Background (part 1)

- Recursion
  - Grammatical structure
  - Patterns of recurrent structural dependencies
  - Human processing of recurrent dependency patterns

Recursive definition

- Base case:
  - \{I stink, you stink\} ⊂ L
- Recursive step:
  - If \( x \in L \), then so are \( I \) think that \( x \) and \( you \) think that \( x \).
- Closure
  - Nothing else is a member of \( L \).

Product of recursive definition

- \( L \) as defined in the previous slide can be characterized as the following regular expression:
  - \((/I/ you) think that\^* (/I/ you) stink

Strings vs. structures

- \( L \) as just defined and described consists of a set of strings of “words”.
- Expressions in natural (human) languages are also structured.
  - Structure is needed to describe structural ambiguity.
  - Let \( s = I \)’m ironing the shirt in the closet
One structure for *s*

I'm ironing the shirt in the closet

I'm ironing the shirt

I'm ironing

I'm ironing the shirt

I'm ironing

Another structure for *s*

I'm ironing the shirt in the closet

I'm ironing

I'm ironing

I'm ironing the shirt

I'm ironing

Background (part 3)

- Recursion
- Grammatical structure
- Patterns of recurrent structural dependencies
- Human processing of recurrent dependency patterns

Grammatical dependencies

- Certain parts of a grammatical structure depend on others.
  - e.g. *m* depends on */in s.*
- Dependencies can be classified into a finite number of types.
  - e.g. subject-verb agreement.

Recurrent dependency types

- Dependency types can recur within the same grammatical structure.
- When they do, they are arranged in one of three patterns.
  - Serial
  - Nesting
  - Crossing

Recurrent dependency patterns

- Serial
  - /--m* principled, you--'re rigid, and he--'s an obstinate fool.
- Nesting
  - Have you met the teacher all the students the woman everyone knows likes distrust?
- Crossing (not illustrated or discussed further)
  - Note: Dependent pairs are highlighted the same and are separated if necessary by “-”.
Degree of serial and nesting dependency

• Let \((c_i, d_i)\) be the \(i\)th dependency type in a structure, and let \(x\) represent the material (possibly null) intervening between dependencies and their parts.
  
  – Degree \(n\) serial dependency
    
    \[ x \ c_1 \ x \ d_1 \ x \ ... \ x \ c_n \ x \ d_n \ x \]
  
  – Degree \(n\) nesting dependency
    
    \[ x \ c_n \ x \ ... \ x \ c_1 \ x \ d_1 \ x \ ... \ x \ d_n \ x \]

Chomsky hierarchy (in part)

• If a language exhibits recurrent serial dependencies of unbounded degree only,
  
  – it is a regular (or finite-state) language.
    
    • The term “regular” is due to Kleene.
  
  • If a language exhibits recurrent serial and nesting dependencies of unbounded degree only,
    
    – it is a context-free language.
      
      • The term “context free” is due to Chomsky.

Background (part 4)

• Recursion
• Grammatical structure
• Patterns of recurrent structural dependencies
  
  ➢ Human processing of recurrent dependency patterns

Comprehending recurrent dependencies

• Recurrent serial dependencies are easy for people to process.
  
  – Structures with very high degrees of serial dependencies and are otherwise well formed are generally acceptable (comprehensible) to fluent speakers of a language.
  
  • Recurrent nesting dependencies are hard for people to process.
    
    – Structures with degree \(>3\) nesting dependencies are generally unacceptable (incomprehensible) in all languages.

Regularity of the class of acceptable expressions

• Consequently, the class of acceptable (comprehensible) expressions in any natural language is a regular language, since there is a finite upper bound on the degrees of nesting dependencies in that class.

Grammaticality and acceptability

• Are linguistic structures with arbitrarily high degree of nesting dependency grammatical?
  
  – If yes, then the natural languages containing them must be at least context free.
  
  – If no, then all natural languages are regular.
Why languages might not be regular

- The classic argument for considering natural languages to be beyond regular is simplicity.
  - Imposing a finite bound on degree of nesting dependencies is arbitrary; “grammars can’t count”.

Why languages might be regular

- Assumption that grammars may be context free (or even more powerful) is a priori.
  - Behavioral evidence (specifically, complete unacceptability of structures with high degrees of nesting) points to conclusion that natural languages are regular.
- Regular languages are more easily computable than context-free ones.

Motivating constraints on nesting (part 1)

- Eliminate nesting dependencies in right- and left-recursive structures by readjustment.
- Constrain nesting of recurring coordinate structures.
- Constrain zigzag (change-of-direction) recursion (≈ center embedding).

A right-recursive structure

```
we recalled that somebody said that everybody knew that we were engaged
we recalled that somebody said that everybody knew that we were engaged
we recalled that everybody knew that we were engaged
```

A left-recursive structure

```
my younger brother’s closest friend’s college algebra teacher’s hidden agenda
my younger brother’s closest friend’s college algebra teacher’s hidden agenda
my younger brother’s closest friend’s college algebra teacher’s hidden agenda
```
Equivalent readjusted pseudo-coordinate structure

N1 = my younger brother’s
closest friend’s college algebra teacher’s hidden agenda

N2 = tN1 closest friend’s
college algebra teacher’s hidden agenda

N3 = tN2 college algebra teacher’s hidden agenda

Motivating constraints on nesting (part 2)

- Eliminate nesting dependencies in right- and left-recursive structures by readjustment.
- Constrain nesting of recurrent coordinate structures.
- Constrain zigzag (change-of-direction) recursion (= center embedding).

Ordinary, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Ordinarily, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Ordinarily, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Ordinarily, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Ordinarily, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Ordinarily, coordinate structure is non-nesting

I’m principled, you’re rigid, and he’s an obstinate fool

But sometimes it shows limited degree of nesting

Peel the plums and halve them or slice them

Peel the plums and halve them or slice them

The limit is very small

Frightened or defiant, cursing and fighting or sullen and withdrawn

Frightened or defiant, cursing and fighting or sullen and withdrawn

Higher degree coordinate subordination is not attested

Frightened or defiant, cursing and fighting or sullen and withdrawn
Motivating constraints on nesting (part 3)

- Eliminate nesting dependencies in right- and left-recursive structures by readjustment.
- Constrain nesting of recurrent coordinate structures.
- Constrain zigzag (change-of-direction) nesting (≈ center embedding).

Zigzag recursion induced by adverbial modification

- The sentence *Somebody said that everybody knew that we got engaged last December* is three-ways ambiguous, depending on which clause (top, bottom or middle) the adverbial *last December* modifies.
- Each structure exhibits degree 1 center embedding
  - indicated by highlighted arcs; note that middle modification involves two changes of direction (zigzags), whereas the others involve only one.

Top adverbial modification

```
somebody said that everybody knew that we got engaged last December
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```

Bottom adverbial modification

```
somebody said that everybody knew that we got engaged last December
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```

Middle adverbial modification

```
somebody said that everybody knew that we got engaged last December
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```
```
somebody said that everybody knew that we got engaged
```

Preference for top and bottom modification

- Top and bottom modifications are preferred to middle modification.
  - Readjustment of right recursion accounts for this preference.
  - Preference for bottom modification over top modification or vice versa is not accounted for.
  - Such preference may be language dependent.
How regular natural languages might grow

1. Start with nonrecursive structures and non-nested coordinate structures.
2. Add readjusted coordinate and right- and left-recursive structures.
3. Add all structures with degree 1 zigzag recursion.
4. Continue with degree 2, etc. up to any desired finite degree of zigzag recursion.

Start with nonrecursive and non-nested coordinate expressions

- Let $F_0$ be a finite-state transducer for parsing for the (finite) class of nonrecursive and non-nested coordinate expressions $L_0$ in a language $L$. 

Significance of degree of zigzag recursion

- Degree of zigzag recursion (number of changes of direction of a recursive arc)
  - is a measure of processing complexity of recursive structure;
  - is a more sensitive measure of complexity than degree of center embedding;
  - is preserved under readjustment.

Readjusted structure for top modification

Readjusted structure for bottom modification

Readjusted structure for middle modification
Second, add readjusted coordinate and right- and left-recursive structures

- Construct $F_1$ from $F_0$ by adding the necessary states and transitions to parse all readjusted coordinate and right- and left-recursive expressions $L_1$ of $L$.

Third, add degree 1 zigzag recursion

- Construct $F_2$ from $F_1$ by adding the necessary states and transitions to parse all nonrecursive, readjusted right- and left-recursive, and readjusted degree 1 zigzag recursive expressions $L_2$ of $L$.

Finally, add degree $n$ zigzag recursion

- Construct $F_n$ from $F_{n-1}$ by adding the necessary states and transitions to parse all nonrecursive, readjusted right- and left-recursive, and readjusted degree $n$ zigzag recursive expressions $L_n$ of $L$.

Conclusion (part 1)

- The procedure is a model for acquisition of more and more complex linguistic structures, which however remains in the domain of regular languages.

Conclusion (part 2)

- The upper bound on degree of nesting (more precisely degree of zigzag recursion) arises organically from the construction. It is not an arbitrarily imposed restriction.

Conclusion (part 3)

- The context-free language $L$ that the construction approximates but never reaches can be considered an abstract idealization of the psychologically real languages $L_n$ (for various $n$) acquired by human beings.