Memory Usage as a Measure of Structural Complexity in Minimalist Parsing

Aniello De Santo

aniello.desanto@stonybrook.edu
aniellodesanto.github.io

JHU
Oct 30, 2019

One Big Question

(How much) does grammatical structure matter in sentence processing?

The MG Parsing Project

Syntactic complexity ⇔ *Parser behavior* ⇔ *Processing difficulty*

The Goal

- Can we give a *maximally simple* parsing model that derives off-line processing effects purely from memory usage?

Outline

A Trivial (?) Observation

Not All Sentences Are Processed Equally

- Center embedding VS Right embedding
  - RE The woman saw the boy that heard the man that left.
  - CE The woman the boy (that) the man that left heard saw.
- Subject VS object relative clauses
  - SRC I saw the horse that kicked the wolf
  - ORC I saw the horse that the wolf kicked
- Attachment preferences
  - 1a. I saw [a girl with the telescope]
  - 1b. I [saw a girl] [with the telescope]

Which aspects of grammar influence sentence processing?

A Trivial (?) Observation

Not All Sentences Are Processed Equally

- Center embedding VS Right embedding
  - RE The woman saw the boy that heard the man that left.
  - CE The woman the boy (that) the man that left heard saw.
- Subject VS object relative clauses
  - SRC I saw the horse that kicked the wolf.
  - ORC I saw the horse that the wolf kicked.
- Attachment preferences
  - 1a. I saw [a girl with the telescope]
  - 1b. I [saw a girl] [with the telescope]

Which aspects of grammar influence sentence processing?

A Trivial (?) Observation

Not All Sentences Are Processed Equally

- Center embedding VS Right embedding
  - RE **The woman saw the boy that heard the man that left**.
  - CE **The woman the boy** (that) **the man that left heard saw**.
- Subject VS object relative clauses
  - SRC I saw **the horse** $[_{RC}$ that **t** kicked the wolf].
  - ORC I saw **the horse** $[_{RC}$ that the wolf kicked **t**].
- Attachment preferences
  - 1a. I saw **[a girl with the telescope]**
  - 1b. I **[saw a girl] [with the telescope]**

Which aspects of grammar influence sentence processing?

A Trivial (?) Observation

Not All Sentences Are Processed Equally

- Center embedding VS Right embedding
  - RE **The woman saw the boy that heard the man that left.**
  - CE **The woman the boy** (that) **the man that left heard saw.**
- Subject VS object relative clauses
  - SRC I saw **the horse** [$_{RC}$ that **t** kicked the wolf].
  - ORC I saw **the horse** [$_{RC}$ that the wolf kicked **t**].
- Attachment preferences
  - 1a. I saw **[a girl with the telescope]**
  - 1b. I **[saw a girl] [with the telescope]**

Which aspects of grammar influence sentence processing?

Sounds familiar?

Which aspects of grammar influence sentence processing?

- What is the relation between grammatical operations and cognitive processes?

Derivational Theory of Complexity (Miller and Chomsky, 1963)

- **One-to-one mapping** between processing complexity and length of a derivation (Fodor & Garrett 1967; Berwick & Weinberg 1983)
- Essentially: there is a **cost** to mental computations.

- What is the right notion of syntactic derivation?
- What is costly? And why?

Sounds familiar?

Which aspects of grammar influence sentence processing?

- What is the relation between grammatical operations and cognitive processes?

Derivational Theory of Complexity (Miller and Chomsky, 1963)

- **One-to-one mapping** between processing complexity and length of a derivation
  (Fodor & Garrett 1967; Berwick & Weinberg 1983)
- Essentially: there is a **cost** to mental computations.

- What is the right notion of syntactic derivation?
- What is costly? And why?

A Formal Model of Sentence Processing

The Power of Explicit Grammar Formalisms

We can relate formal models of competence to formal models of performance in transparent, quantifiable ways.

The Model

1. a formalization of syntax → Minimalist grammars
2. a theory of how structures are built → top-down parser
3. a linking theory → complexity metrics for memory usage

Perks

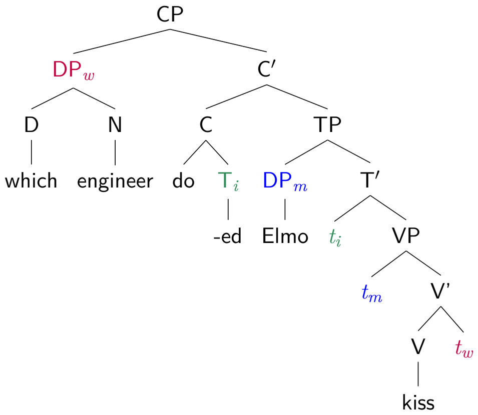
- sensitive to fine-grained structural differences
- bridge between theoretical syntax and processing data

Minimalist Grammars (MGs)

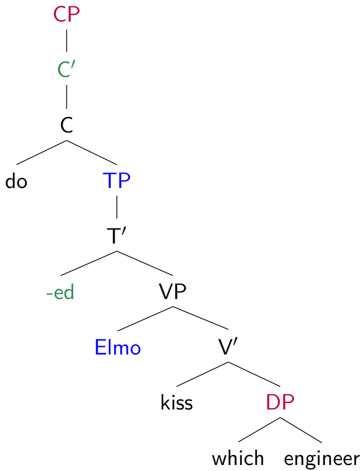
We need a formal model of syntactic structures...

- Minimalist grammars (**MGs**) are a formalization of Chomskyan syntax (Chomsky 1995; Stabler 1997)
- Grammar is just a finite list of feature-annotated lexical items (LIs)
- Operations: **Merge** and **Move**
- Essentially: CFGs with a more complicated mapping from trees to strings

MG Syntax: Derivation Trees



Phrase Structure Tree



Derivation Tree

Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 CP is conjectured
step 2 CP expands to C′
step 3 C′ expands to does and TP
step 4 TP expands to Salem and T′
step 5 T′ expands to T and VP
step 6 VP expands to mock and who
step 7 who is found
step 8 does is found
step 9 Salem is found
step 10 T is found
step 11 mock is found

Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 CP is conjectured
step 2 CP expands to C′
step 3 C′ expands to does and TP
step 4 TP expands to Salem and T′
step 5 T′ expands to T and VP
step 6 VP expands to mock and who
step 7 who is found
step 8 does is found
step 9 Salem is found
step 10 T is found
step 11 mock is found

Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

^{1}CP

- • Who • does • Salem • T • mock

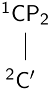
step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found

Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock



step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *does* and *TP*
step 4	*TP* expands to *Salem* and *T'*
step 5	*T'* expands to *T* and *VP*
step 6	*VP* expands to *mock* and *who*
step 7	*who* is found
step 8	*does* is found
step 9	*Salem* is found
step 10	*T* is found
step 11	*mock* is found

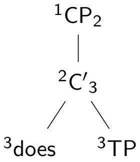
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



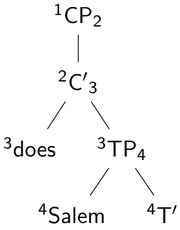
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *does* and *TP*
step 4	*TP* expands to *Salem* and *T'*
step 5	*T'* expands to *T* and *VP*
step 6	*VP* expands to *mock* and *who*
step 7	*who* is found
step 8	*does* is found
step 9	*Salem* is found
step 10	*T* is found
step 11	*mock* is found



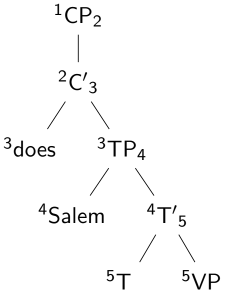
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



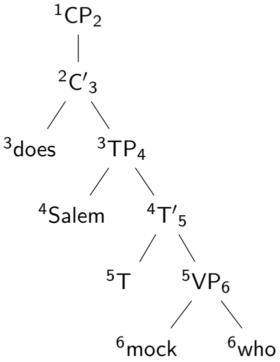
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



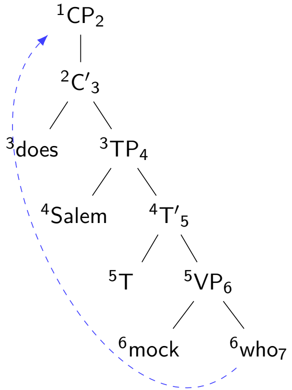
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



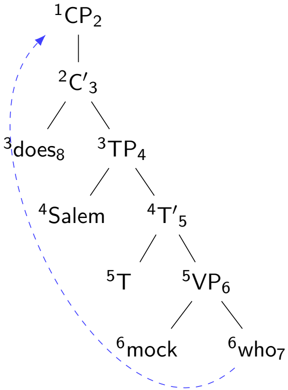
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



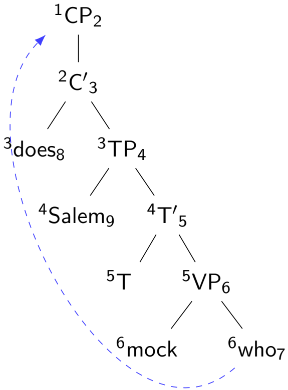
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *does* and *TP*
step 4	*TP* expands to *Salem* and *T'*
step 5	*T'* expands to *T* and *VP*
step 6	*VP* expands to *mock* and *who*
step 7	*who* is found
step 8	*does* is found
step 9	*Salem* is found
step 10	*T* is found
step 11	*mock* is found



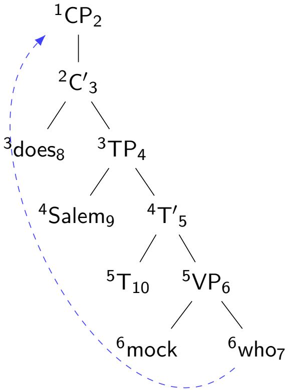
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



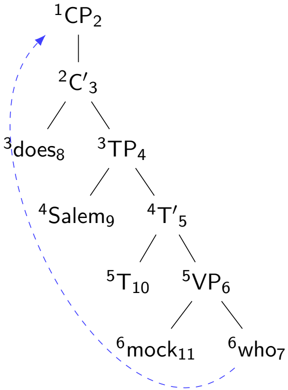
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1 *CP* is conjectured
step 2 *CP* expands to *C'*
step 3 *C'* expands to *does* and *TP*
step 4 *TP* expands to *Salem* and *T'*
step 5 *T'* expands to *T* and *VP*
step 6 *VP* expands to *mock* and *who*
step 7 *who* is found
step 8 *does* is found
step 9 *Salem* is found
step 10 *T* is found
step 11 *mock* is found



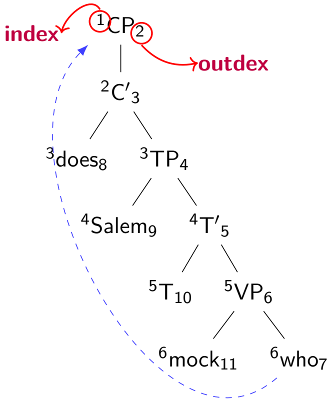
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *does* and *TP*
step 4	*TP* expands to *Salem* and *T'*
step 5	*T'* expands to *T* and *VP*
step 6	*VP* expands to *mock* and *who*
step 7	*who* is found
step 8	*does* is found
step 9	*Salem* is found
step 10	*T* is found
step 11	*mock* is found



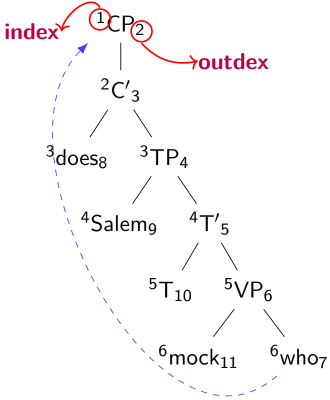
Incremental Top-Down Parsing

How (Modified) recursive descent parser (Stabler 2013)

Strategy Hypothesize structure top-down and verify that words in structure match input string (*string-driven recursive descent*).

- • Who • does • Salem • T • mock

step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *does* and *TP*
step 4	*TP* expands to *Salem* and *T'*
step 5	*T'* expands to *T* and *VP*
step 6	*VP* expands to *mock* and *who*
step 7	*who* is found
step 8	*does* is found
step 9	*Salem* is found
step 10	*T* is found
step 11	*mock* is found



Index and Outdex are our connection to memory!

Memory-Based Complexity Metrics

- **Memory usage**:

Tenure how long a node is kept in memory
Size how much information is stored in a node
$\Rightarrow$ Intuitively, the length of its movement dependency!

- These can be formalized into **complexity metrics** (Kobele et al. 2012)

MaxTenure $max(\{\text{tenure-of}(n) | n \text{ a node of the tree}\})$

SumSize $\sum_{m \in M} size(m)$

Ranked $\langle MaxTenure, SumSize \rangle$

- Currently: **40 base metrics!**

Memory-Based Complexity Metrics

- **Memory usage**:

Tenure how long a node is kept in memory
Size how much information is stored in a node
$\Rightarrow$ Intuitively, the length of its movement dependency!

- These can be formalized into **complexity metrics** (Kobele et al. 2012)

MaxTenure $max(\{\text{tenure-of}(n)|n \text{ a node of the tree}\})$

SumSize $\sum_{m \in M} size(m)$

Ranked $\langle MaxTenure, SumSize \rangle$

- Currently: **40 base metrics!**

Memory-Based Complexity Metrics

- **Memory usage**:

Tenure how long a node is kept in memory
Size how much information is stored in a node
$\Rightarrow$ Intuitively, the length of its movement dependency!

- These can be formalized into **complexity metrics** (Kobele et al. 2012)

MaxTenure $max(\{\text{tenure-of}(n) | n \text{ a node of the tree}\})$

SumSize $\sum_{m \in M} size(m)$

Ranked $\langle MaxTenure, SumSize \rangle$

- Currently: **40 base metrics!**

Memory-Based Complexity Metrics

- **Memory usage**:

Tenure how long a node is kept in memory
Size how much information is stored in a node
$\Rightarrow$ Intuitively, the length of its movement dependency!

- These can be formalized into **complexity metrics** (Kobele et al. 2012)

MaxTenure $max(\{\text{tenure-of}(n) | n \text{ a node of the tree}\})$

SumSize $\sum_{m \in M} size(m)$

Ranked $\langle MaxTenure, SumSize \rangle$

- Currently: **40 base metrics!**

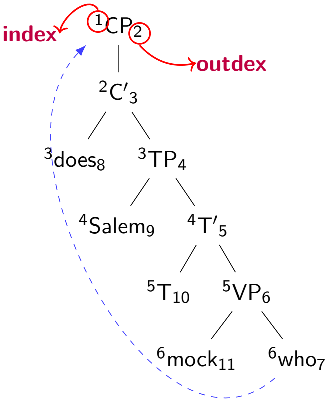
Space of Possible Metrics?

<MaxT,SumS> makes correct predictions cross-linguistically!

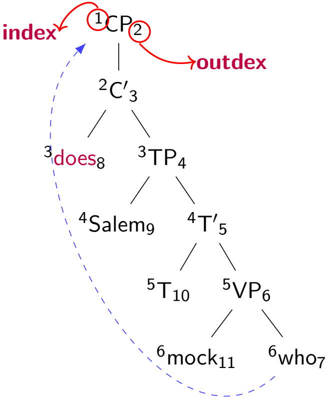
- Right < center embedding (Kobele et al. 2012)
- Crossing < nested dependencies (Kobele et al. 2012)
- SC-RC < RC-SC (Graf & Marcinek 2014)
- SRC < ORC (Graf et al. 2017)
  - English
  - Korean
  - Japanese
- Postverbal subjects in Italian (De Santo 2019)
- ...

Computing Metrics: An Example

1. **Tenure** how long a node is kept in memory
2. **Size** (Intuitively) the length of movement dependencies!



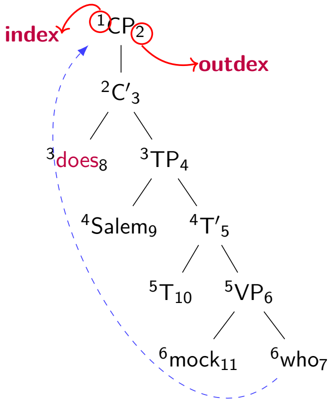
Computing Metrics: An Example



1. **Tenure** how long a node is kept in memory
2. **Size** (Intuitively) the length of movement dependencies!

Tenure($does$**)** $= 8 - 3 = 5$

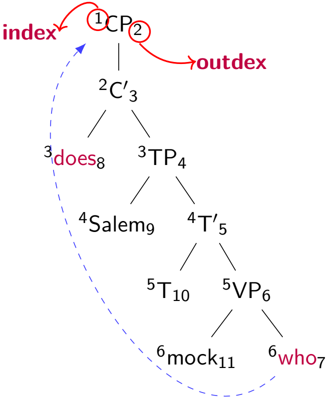
Computing Metrics: An Example



1. **Tenure** how long a node is kept in memory
2. **Size** (Intuitively) the length of movement dependencies!

Tenure($does$**)** $= 8 - 3 = 5$
MaxTenure $= max\{$**Tenure**$(does)$, **Tenure**$(Salem), \ldots\} = 5$

Computing Metrics: An Example

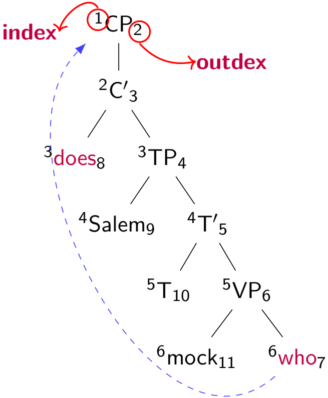


1. **Tenure** how long a node is kept in memory
2. **Size** (Intuitively) the length of movement dependencies!
   index(origin(m)) $-$ *index(landing(m))*

Tenure($does$**)** $= 8 - 3 = 5$
MaxTenure $= max\{$**Tenure**$(does)$, **Tenure**$(Salem), \dots\} = 5$

Size(who**)** $= 6 - 1 = 5$

Computing Metrics: An Example



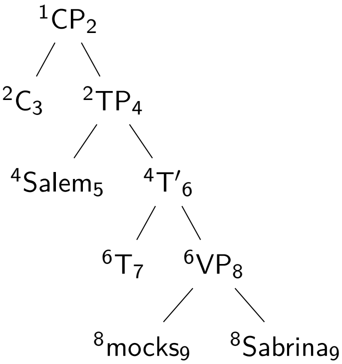
1. **Tenure** how long a node is kept in memory
2. **Size** (Intuitively) the length of movement dependencies!
   index(origin(m)) − index(landing(m))

$$\textbf{Tenure}(does) = 8 - 3 = 5$$
$$\textbf{MaxTenure} = max\{\textbf{Tenure}(does), \textbf{Tenure}(Salem), \ldots\} = 5$$

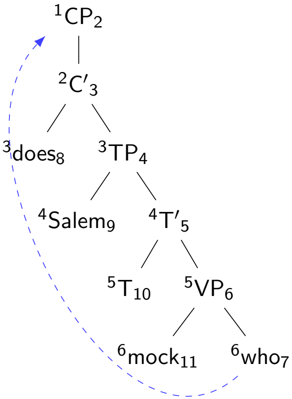
$$\textbf{Size}(who) = 6 - 1 = 5$$
$$\textbf{SumSize} = \sum \textbf{Size}(who) = 5$$

Contrasting Derivations

MaxTenure $= 2$

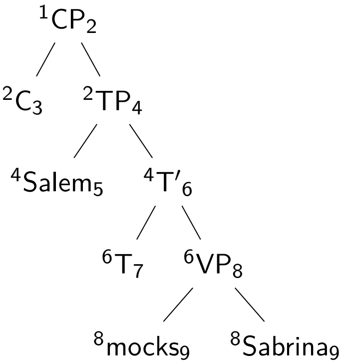


MaxTenure $= 5$

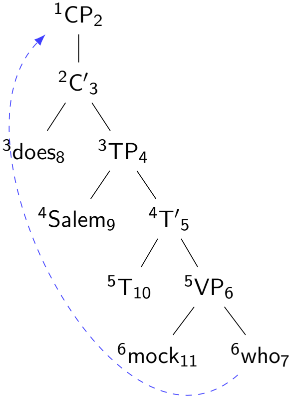


Contrasting Derivations

MaxTenure $= 2$
SumSize $= 0$



MaxTenure $= 5$
SumSize $= 5$



Summary of the Approach

General Idea

(Kobele et al. 2012; Gerth 2015; Graf et al. 2017)

- pick competing derivations
- evaluate metrics over each
- compare parser's prediction to off-line processing data

Simplifying Assumptions

- Parser as an Oracle ⇒ Discard beam search
- factor out cost of finding correct parse

A Case Study: Italian Postverbal Subjects

Asymmetries in Italian Relative Clauses

Italian speakers conform to the general cross-linguistic preference for SRC over ORC (Adani et al. 2010; Arosio et al. 2008)

(1)

Il	cavallo	che	ha	inseguito	i	leoni
The	horse	that	has	chased	the	lions

"The horse that chased the lions" **SRC**

(2)

Il	cavallo	che	i	leoni	hanno	inseguito
The	horse	that	the	lions	have	chased

"The horse that the lions chased" **ORC**

Postverbal Subjects and Ambiguity

Italian allows for postverbal subjects, making some sentences ambiguous (De Vincenzi 1991):

(3) Il cavallo che ha inseguito il leone
The horse that has chased the lion

a. "The horse that chased the lion" **SRC**
b. "The horse that the lion chased" **ORCp**

SRC > ORCp

Agreement can disambiguate:

(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions

"The horse that the lions chased" **ORCp**

Postverbal Subjects and Ambiguity

Italian allows for postverbal subjects, making some sentences ambiguous (De Vincenzi 1991):

(3) Il cavallo che ha inseguito il leone
The horse that has chased the lion

a. "The horse that chased the lion" **SRC**

b. "The horse that the lion chased" **ORCp**

SRC > ORCp

Agreement can disambiguate:

(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions

"The horse that the lions chased" **ORCp**

Asymmetries in Italian Relative Clauses [cont.]

(1) Il cavallo che ha inseguito i leoni
The horse that has chased the lions
"The horse that chased the lions" **SRC**

(2) Il cavallo che i leoni hanno inseguito
The horse that the lions have chased
"The horse that the lions chased" **ORC**

(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions
"The horse that the lions chased" **ORCp**

SRC > ORC > ORCp

Modeling Assumptions

Reminder:

- Parsing strategy
  $\Rightarrow$ Top-down parser
- Complexity Metrics
  $\Rightarrow$ MaxTenure and SumSize

Degrees of freedom: Syntactic analyses

1. RC constructions $\rightarrow$ (Kayne 1994)
2. Postverbal subjects $\rightarrow$ (Belletti & Leonini 2004)

Modeling Assumptions

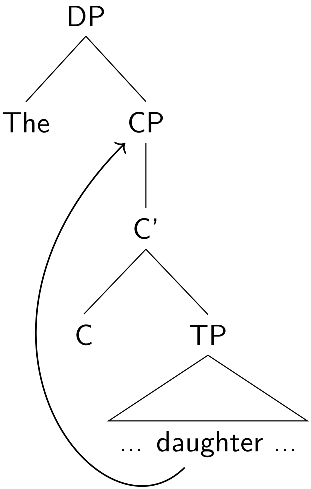
Reminder:

- Parsing strategy
  ⇒ Top-down parser
- Complexity Metrics
  ⇒ MaxTenure and SumSize

Degrees of freedom: Syntactic analyses

1. RC constructions → (Kayne 1994)
2. Postverbal subjects → (Belletti & Leonini 2004)

Kayne's Promotion Analysis (Kayne 1994)

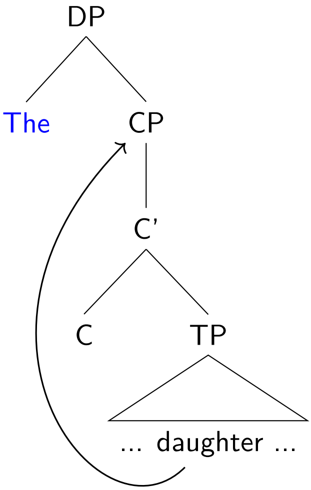


- RC is selected by an external D^0
- the RC head is a nominal constituent
- the RC head raises from its base position to [Spec, CP]

$[_{DP}$ The $[_{CP}$ daughter$_i$ [that t_i was on the balcony]]]

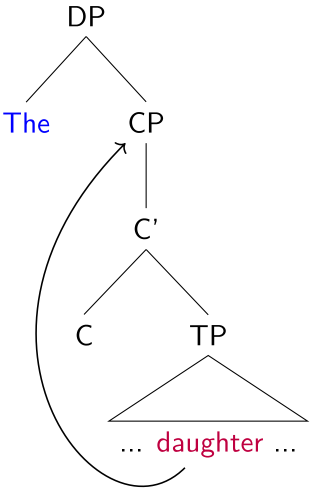
Kayne's Promotion Analysis (Kayne 1994)

- RC is selected by an external D^0
- the RC head is a nominal constituent
- the RC head raises from its base position to [Spec, CP]



$[_{DP}$ The $[_{CP}$ daughter$_i$ [that t_i was on the balcony]]]

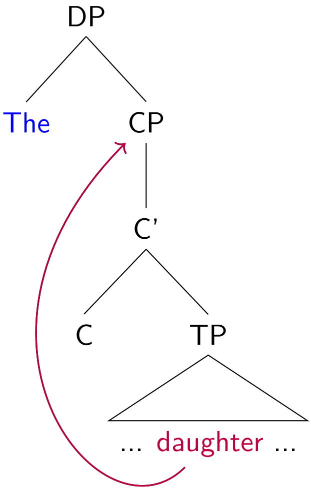
Kayne's Promotion Analysis (Kayne 1994)



- RC is selected by an external D^0
- the RC head is a nominal constituent
- the RC head raises from its base position to [Spec, CP]

$[_{DP}$ The $[_{CP}$ daughter$_i$ [that t_i was on the balcony]]]

Kayne's Promotion Analysis (Kayne 1994)



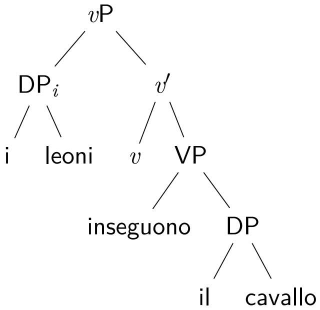
- RC is selected by an external D^0
- the RC head is a nominal constituent
- the RC head raises from its base position to [Spec, CP]

[$_{DP}$ The [$_{CP}$ daughter$_i$ [that t_i was on the balcony]]]

Postverbal Subjects (Belletti & Leonini 2004)

(5) Inseguono il cavallo i leoni
Chase the horse the lions
"The lions chase the horse"

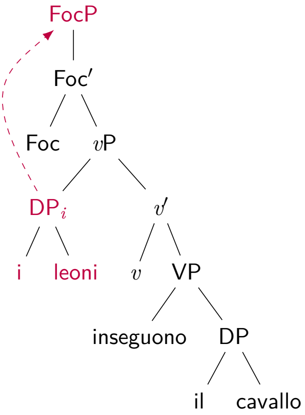
- the subject DP raises to Spec, FocP
- The whole vP raises to Spec,TopP
- an expletive *pro* is base generated in Spec,TP



Postverbal Subjects (Belletti & Leonini 2004)

(6) Inseguono il cavallo i leoni
Chase the horse the lions
"The lions chase the horse"

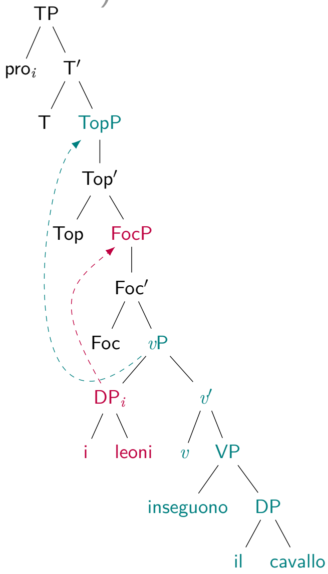
- the subject DP raises to Spec, FocP
- The whole vP raises to Spec,TopP
- an expletive *pro* is base generated in Spec,TP



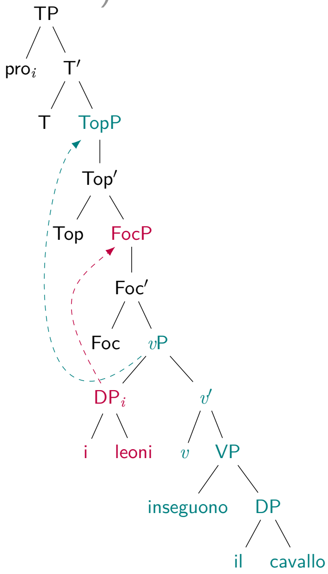
Postverbal Subjects (Belletti & Leonini 2004)

(7) Inseguono il cavallo i leoni
Chase the horse the lions
"The lions chase the horse"

- the subject DP raises to Spec, FocP
- The whole vP raises to Spec,TopP
- an expletive *pro* is base generated in Spec,TP



Postverbal Subjects (Belletti & Leonini 2004)



(7) Inseguono il cavallo i leoni
Chase the horse the lions
"The lions chase the horse"

- the subject DP raises to Spec, FocP
- The whole vP raises to Spec,TopP
- an expletive *pro* is base generated in Spec,TP

Modeling Results

(1) Il cavallo che ha inseguito i leoni
The horse that has chased the lions
"The horse that chased the lions" **SRC**

(2) Il cavallo che i leoni hanno inseguito
The horse that the lions have chased
"The horse that the lions chased" **ORC**

(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions
"The horse that the lions chased" **ORCp**

SRC > ORC > ORCp

Modeling Results

(1) Il cavallo che ha inseguito i leoni
The horse that has chased the lions
"The horse that chased the lions" **SRC**

(2) Il cavallo che i leoni hanno inseguito
The horse that the lions have chased
"The horse that the lions chased" **ORC**

(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions
"The horse that the lions chased" **ORCp**

	SRC	>	ORC	>	ORCp
MaxTenure	8/che		11/ha		16/Foc
SumSize	18		24		31

Modeling Results

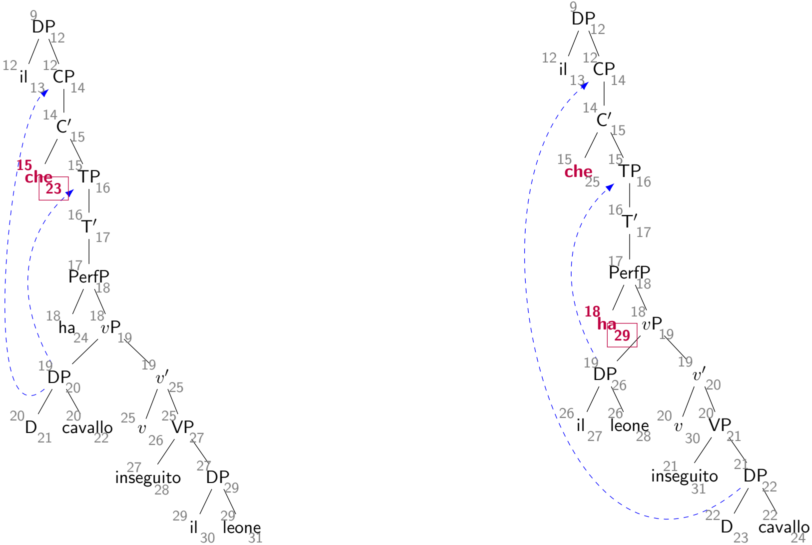
(1) Il cavallo che ha inseguito i leoni
The horse that has chased the lions
"The horse that chased the lions" **SRC**

(2) Il cavallo che i leoni hanno inseguito
The horse that the lions have chased
"The horse that the lions chased" **ORC**

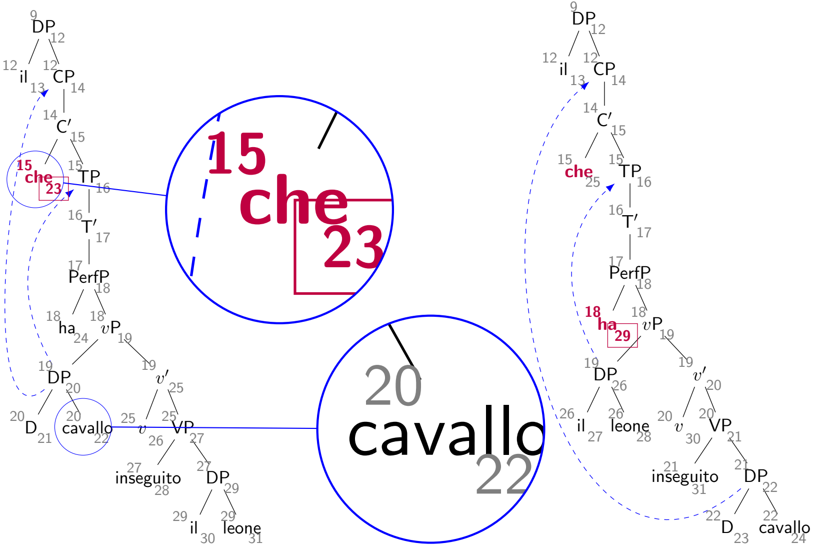
(4) Il cavallo che hanno inseguito i leoni
The horse that have chased the lions
"The horse that the lions chased" **ORCp**

	SRC	>	ORC	>	ORCp	
MaxTenure	8/che		11/ha		16/Foc	✓
SumSize	18		24		31	✓

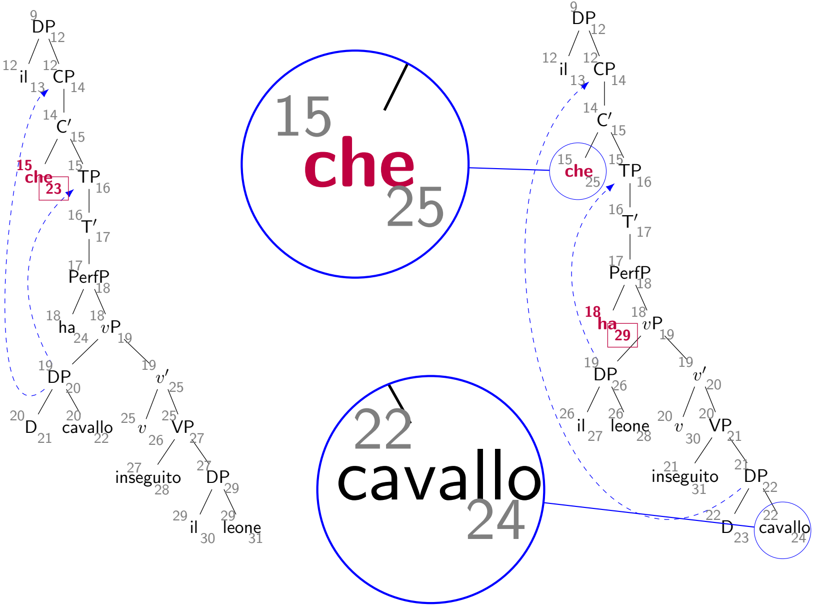
Results: SRC > ORC



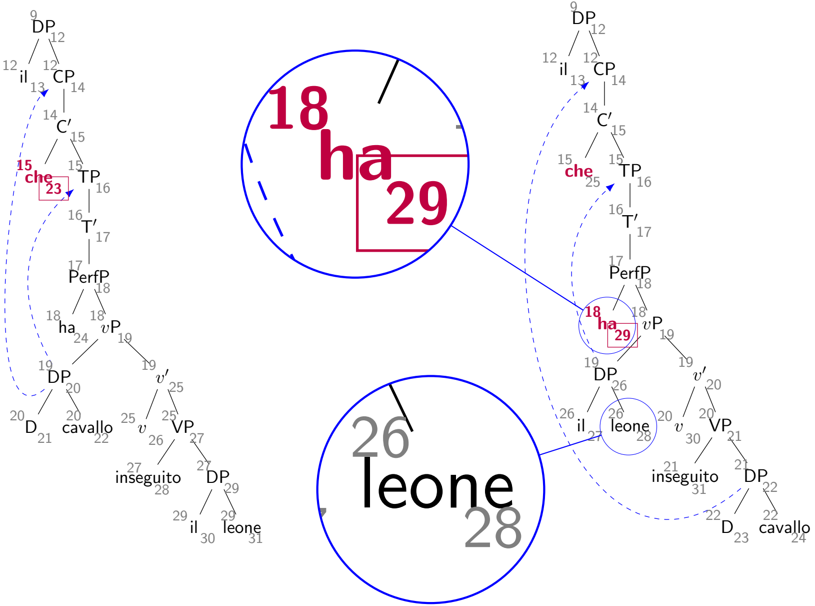
Results: SRC > ORC



Results: SRC > ORC



Results: SRC > ORC



Summary of Results (De Santo 2019)

Clause Type	MaxTenure	SumSize
obj. SRC > ORC	✓	✓
obj. SRC > ORCp	✓	✓
obj. ORC > ORCp	✓	✓
subj. SRC > ORC	tie	✓
subj. SRC > ORCp	✓	✓
subj. ORC > ORCp	✓	✓
matrix SVO > VOS	✓	✓
VS unacc > VS unerg	✓	✓

Table: Predictions of the MG parser by metric and contrast.

Interim Summary

<MaxT,SumS> makes correct predictions cross-linguistically!

- Right < center embedding (Kobele et al. 2012)
- Crossing < nested dependencies (Kobele et al. 2012)
- SC-RC < RC-SC (Graf & Marcinek 2014)
- SRC < ORC (Graf et al. 2017)
- Postverbal subjects in Italian (De Santo 2019)

Interim Summary

<MaxT,SumS> makes correct predictions cross-linguistically!

- Right < center embedding (Kobele et al. 2012)
- Crossing < nested dependencies (Kobele et al. 2012)
- SC-RC < RC-SC (Graf & Marcinek 2014)
- SRC < ORC (Graf et al. 2017)
- Postverbal subjects in Italian (De Santo 2019)

Can we get theoretical insights?

- Modeling Gradient Acceptability
  ⇒ Gradience in Island Effects (De Santo 2020)

Interim Summary

<MaxT,SumS> makes correct predictions cross-linguistically!

- Right < center embedding (Kobele et al. 2012)
- Crossing < nested dependencies (Kobele et al. 2012)
- SC-RC < RC-SC (Graf & Marcinek 2014)
- SRC < ORC (Graf et al. 2017)
- Postverbal subjects in Italian (De Santo 2019)

Can we get theoretical insights?

- Modeling Gradient Acceptability
  $\Rightarrow$ Gradience in Island Effects (De Santo 2020)

Gradient Acceptability and Categorical Grammars

Acceptability judgments are not binary but *gradient*:

> *An adequate linguistic theory will have to recognize degrees of grammaticalness [...] there is little doubt that speakers can fairly consistently order new utterances, never previously heard, with respect to their degree of belongingness to the language.*
>
> *(Chomsky 1975: 131-132)*

But mainstream syntactic theories rely on categorical grammars!

Gradient Acceptability and Categorical Grammars

Acceptability judgments are not binary but *gradient*:

> *An adequate linguistic theory will have to recognize degrees of grammaticalness [...] there is little doubt that speakers can fairly consistently order new utterances, never previously heard, with respect to their degree of belongingness to the language.*
>
> *(Chomsky 1975: 131-132)*

But mainstream syntactic theories rely on categorical grammars!

Gradient Acceptability and Categorical Grammars

Acceptability judgments are not binary but *gradient*:

> *An adequate linguistic theory will have to recognize degrees of grammaticalness [...] there is little doubt that speakers can fairly consistently order new utterances, never previously heard, with respect to their degree of belongingness to the language.*
>
> (Chomsky 1975: 131-132)

But mainstream syntactic theories rely on categorical grammars!

Models of Gradience

(At least two) theories of gradience:

- gradience incorporated in the grammar
  (Keller 2000; Featherston 2005; Lau et al. 2014)
- gradience due to extra-grammatical factors
  (Chomsky 1975; Schutze 1996)

But: these approaches aim to explain the same data!

The contribution of formal models?

Quantify what each approach needs to account for the data:

- additional syntactic assumptions
- additional complexity in acquisition, processing strategies, etc.

Models of Gradience

(At least two) theories of gradience:

- gradience incorporated in the grammar
  (Keller 2000; Featherston 2005; Lau et al. 2014)
- gradience due to extra-grammatical factors
  (Chomsky 1975; Schutze 1996)

But: these approaches aim to explain the same data!

The contribution of formal models?

Quantify what each approach needs to account for the data:

- additional syntactic assumptions
- additional complexity in acquisition, processing strategies, etc.

(Quantitative) Models of Gradience

Gradient Grammars (Keller 2000; Lau et al. 2014)

- OT-style constraint ranking
- Probabilistic grammars

Extra-grammatical Factors (Chomsky 1975; Schutze 1996)

- processing effects
  - plausibility
  - working memory limitations
  - **But:** few models for quantitative predictions!

Hypothesis

We can use the MG parser to test the relation between categorical grammar, processing difficulty, and gradience!

(Quantitative) Models of Gradience

Gradient Grammars (Keller 2000; Lau et al. 2014)

- OT-style constraint ranking
- Probabilistic grammars

Extra-grammatical Factors (Chomsky 1975; Schutze 1996)

- processing effects
  - plausibility
  - working memory limitations
  - **But:** few models for quantitative predictions!

Hypothesis

We can use the MG parser to test the relation between categorical grammar, processing difficulty, and gradience!

Modeling Gradience with an MG Parser

The model is the same as before

1. a formal model of syntax → Minimalist grammars (MGs)
2. a theory of how structures are built → MG parser
3. a linking theory: higher memory cost ⇒ lower acceptability

- sensitive to fine-grained structural differences!
- minimal, pairwise comparisons are maximally interpretable!

A proof-of-concept:

- variation of Island effects in English (Sprouse et al. 2012)

A Proof of Concept: Island Effects

1. What do you think that John bought *t*?
2. What do you wonder whether John bought *t*?

Gradience in Islands: Sprouse et al. (2012)

- A factorial design for islands effects:
  1. Gap Position: Matrix vs. Embedded
  2. Structure: Island vs. Non-Island (Kluender & Kutas 1993)

Results in pairwise comparisons ideal for the MG parser

A Proof of Concept: Island Effects

1. What do you think that John bought *t*?
2. What do you wonder whether John bought *t*?

Gradience in Islands: Sprouse et al. (2012)

- A factorial design for islands effects:
  1. Gap Position: Matrix vs. Embedded
  2. Structure: Island vs. Non-Island (Kluender & Kutas 1993)

Results in pairwise comparisons ideal for the MG parser

A Proof of Concept: Island Effects

1. What do you think that John bought *t*? — Non-Island | Embedded
2. What do you wonder whether John bought *t*? — Island | Embedded
3. Who *t* thinks that John bought a car? — Non-Island | Matrix
4. Who *t* wonders whether John bought a car? — Island | Matrix

Gradience in Islands: Sprouse et al. (2012)

- A factorial design for islands effects:
  1. Gap Position: Matrix vs. Embedded
  2. Structure: Island vs. Non-Island (Kluender & Kutas 1993)

Results in pairwise comparisons ideal for the MG parser

A Proof of Concept: Island Effects

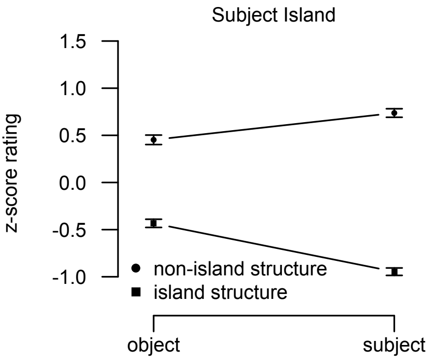
1. What do you think that John bought *t*? — Non-Island | Embedded
2. What do you wonder whether John bought *t*? — Island | Embedded
3. Who *t* thinks that John bought a car? — Non-Island | Matrix
4. Who *t* wonders whether John bought a car? — Island | Matrix

Gradience in Islands: Sprouse et al. (2012)

- A factorial design for islands effects:
  1. Gap Position: Matrix vs. Embedded
  2. Structure: Island vs. Non-Island (Kluender & Kutas 1993)

Results in pairwise comparisons ideal for the MG parser

Deriving Pairwise Comparisons



- Subj | Non Island > Obj | Non Island
- Subj | Non Island > Obj | Island
- Subj | Non Island > Subj | Island
- etc.

Sprouse at al. (2012)

Four island types

Subject islands

- What do you think the speech about *t* interrupted the show about global warming?

Adjunct islands

- What do you laugh if John leaves *t* at the office?

Complex NP islands

- What did you make the claim that John bought *t*?

Whether islands

- What do you wonder whether John bought *t*?

Gap Position $\times$ Structure

1. Matrix vs. Embedded
2. Island vs. Non-Island

Sprouse at al. (2012)

Four island types

Subject islands

- What do you think the speech about *t* interrupted the show about global warming?

Adjunct islands

- What do you laugh if John leaves *t* at the office?

Complex NP islands

- What did you make the claim that John bought *t*?

Whether islands

- What do you wonder whether John bought *t*?

Gap Position × Structure

1. Matrix vs. Embedded
2. Island vs. Non-Island

Modeling Results (De Santo 2020)

Island Type	Sprouse et al. (2012)			MG Parser
Subj. Island 1	Subj. | Non Isl.	>	Obj. | Non Isl.	✓
	Subj. | Non Isl.	>	Obj. | Isl.	✓
	Subj. | Non Isl.	>	Subj. | Isl.	✓
	Obj. | Non Isl.	>	Obj. | Isl.	✓
	Obj. | Non Isl.	>	Subj. | Isl.	✓
	Obj. | Isl.	>	Subj. | Isl.	×
Subj. Island 2	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	>	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓
Adj. Island	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	>	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓
CNP Island	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	=	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓

Modeling Results (De Santo 2020)

Island Type	Sprouse et al. (2012)			MG Parser
Subj. Island 1	Subj. | Non Isl.	>	Obj. | Non Isl.	✓
	Subj. | Non Isl.	>	Obj. | Isl.	✓
	Subj. | Non Isl.	>	Subj. | Isl.	✓
	Obj. | Non Isl.	>	Obj. | Isl.	✓
	Obj. | Non Isl.	>	Subj. | Isl.	✓
	Obj. | Isl.	>	**Subj. | Isl.**	×
Subj. Island 2	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	>	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓
Adj. Island	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	>	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓
CNP Island	Matrix | Non Isl.	>	Emb. | Non Isl.	✓
	Matrix | Non Isl.	=	Matrix | Isl.	✓
	Matrix | Non Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Emb. | Isl.	✓
	Matrix | Isl.	>	Matrix | Isl.	✓
	Emb. | Non Isl.	>	Emb. | Isl.	✓

TL;DR: **Success in all cases but one!**

Subject Island: Case 1

(5) a. What do you think the speech interrupted *t*? Obj | Non Island

b. What do you think *t* interrupted the show? Subj | Non Island

c. What do you think the speech about global warming interrupted the show about *t*? Obj | Island

d. What do you think the speech about *t* interrupted the show about global warming? Subj | Island

Sprouse et al. (2012)			MG Parser
Subj. | Non Isl.	>	Obj. | Non Isl.	✓
Subj. | Non Isl.	>	Obj. | Isl.	✓
Subj. | Non Isl.	>	Subj. | Isl.	✓
Obj. | Non Isl.	>	Obj. | Isl.	✓
Obj. | Non Isl.	>	Subj. | Isl.	✓
Obj. | Isl.	>	Subj. | Isl.	×

Clause Type	MaxT	SumS
Obj./Non Island	14/*do*	19
Subj./Non Island	11/*do*	14
Obj./Island	23/*T2*	22
Subj./Island	15/*do*	20

Subject Island: Case 1

(5)
a. What do you think the speech interrupted *t*? — Obj | Non Island

b. What do you think *t* interrupted the show? — Subj | Non Island

c. What do you think the speech about global warming interrupted the show about *t*? — Obj | Island

d. * What do you think the speech about *t* interrupted the show about global warming? — Subj | Island

Sprouse et al. (2012)			MG Parser
Subj. | Non Isl.	>	Obj. | Non Isl.	✓
Subj. | Non Isl.	>	Obj. | Isl.	✓
Subj. | Non Isl.	>	Subj. | Isl.	✓
Obj. | Non Isl.	>	Obj. | Isl.	✓
Obj. | Non Isl.	>	Subj. | Isl.	✓
Obj. | Isl.	>	Subj. | Isl.	×

Clause Type	MaxT	SumS
Obj./Non Island	14/*do*	19
Subj./Non Island	11/*do*	14
Obj./Island	23/*T2*	22
Subj./Island	15/*do*	20

Subject Island: Case 2

(6) a. Who *t* thinks the speech interrupted the primetime TV show? Matrix | Non Island

b. What do you think *t* interrupted the primetime TV show? Emb. | Non Island

c. Who *t* thinks the speech about global warming interrupted the primetime TV show? Matrix | Island

d. What do you think the speech about *t* interrupted the primetime TV show? Emb. | Island

Sprouse et al. (2012)			MG Parser
Matrix | Non Isl.	>	Emb. | Non Isl.	✓
Matrix | Non Isl.	>	Matrix | Isl.	✓
Matrix | Non Isl.	>	Emb. | Isl.	✓
Matrix | Isl.	>	Emb. | Isl.	✓
Matrix | Isl.	>	Matrix | Isl.	✓
Emb. | Non Isl.	>	Emb. | Isl.	✓

Clause Type	MaxT	SumS
Matrix | Non Isl.	$5/C$	9
Emb. | Non Isl.	$11/do$	14
Matrix | Isl.	$11/T_{RC}$	9
Emb. | Isl.	$17/T_{RC}$	20

Summary

Gradience from a categorical MG grammars?

Modeling gradience in island effects:

- Overall, a success!
- Outlier is expected assuming grammaticalized constraints.

Preliminary results!

- Modulate range of dependencies.
- Other examples of gradience
- Cognitive vs. grammatical constraints? (Ferrara-Boston 2012; Wilcox et al. 2018)

Summary

Gradience from a categorical MG grammars?

Modeling gradience in island effects:

- Overall, a success!
- Outlier is expected assuming grammaticalized constraints.

Preliminary results!

- Modulate range of dependencies.
- Other examples of gradience
- Cognitive vs. grammatical constraints? (Ferrara-Boston 2012; Wilcox et al. 2018)

Summary

Gradience from a categorical MG grammars?

Modeling gradience in island effects:

- Overall, a success!
- Outlier is expected assuming grammaticalized constraints.

Preliminary results!

- Modulate range of dependencies.
- Other examples of gradience
- Cognitive vs. grammatical constraints?
  (Ferrara-Boston 2012; Wilcox et al. 2018)

Summing Up

Minimalist Parsing

A *maximally simple* parsing model that derives processing effects purely from memory usage.

- fully specified parsing model allows for precise predictions
- tight connection with current generative syntax
- successful on a variety of cross-linguistic constructions
- also derives theoretical insights (Kobele et al. 2012)
  - gradience
  - comparative analyses (De Santo & Shafiei 2019)

From the Trees to the Forest

Cognitive Plausibility

- Tenure & Size compatible with a variety of theories
  ⇒ storage, decay, ...

Extending the Model

- What about features?
  - intervention effects
  - structural recall
  - and more!
- Bringing back beam search
  (Torr 2018; Torr et al. 2019; Hunter et al. 2019)

From the Trees to the Forest

Cognitive Plausibility

- Tenure & Size compatible with a variety of theories
  $\Rightarrow$ storage, decay, ...

Extending the Model

- What about features?
  - intervention effects
  - structural recall
  - and more!
- Bringing back beam search
  (Torr 2018; Torr et al. 2019; Hunter et al. 2019)

<Thank you!>

Selected References I

1 **Chomsky, N.** (1995). The minimalist program. Cambridge, Mass.: MIT Press.

2 **De Santo, A.** (2019). Testing a Minimalist gram- mar parser on Italian relative clause asymmetries. In *Proceedings of CMCL 2019*, June 6 2019, Minneapolis, Minnesota.

3 **De Santo, A.** (2020). MG Parsing as a Model of Gradient Acceptability in Syntactic Islands. (To appear) In *Proceedings of SCiL 2020*, Jan 2-5, New Orleans.

4 **De Santo, A.** and Shafiei, N. (2019). On the structure of relative clauses in Persian: Evidence from computational modeling and processing effects. *Talk at the NACIL2*, April 19-21 2019, University of Arizona.

5 **Graf, T.** and Monette, J. and Zhang, C. (2017). Relative Clauses as a Benchmark for Minimalist Parsing. *Journal of Language Modelling*.

6 **Kobele, G.M.**, Gerth S., and Hale. J. (2012). Memory resource allocation in top-down minimalist parsing. In *Formal Grammar*, pages 32–51. Springer.

7 **Sprouse, J.**, Wagers, M. and Phillips, C. (2012). A test of the relation between working-memory capacity and syntactic island effects. *Language*.

8 **Stabler, E.P.** (2013). Bayesian, minimalist, incremental syntactic analysis. *Topics in Cognitive Science* 5:611–633.

9 **Stabler, E.P.** (1997). Derivational minimalism. In Logical aspects of computational linguistics, ed.

Christian Retore, volume 1328 of Lecture Notes in Computer Science, 68–95. Berlin: Springer.

Appendix

Why MGs?

1. Vast analytical coverage
   - MGs handle virtually all analyses in the generative literature
2. Centrality of derivation trees
   - MGs can be viewed as CFGs with a more complicated mapping from trees to strings
3. Simple parsing algorithms
   - Variant of a recursive descent parser for CFGs
     ⇒ cf. TAG (Rambow & Joshi, 1995; Demberg, 2008)

Why These Metrics?

- These complexity metrics are all related to **storage cost** (cf. Gibson, 1998)
- We could implement alternative ones (cf. Ferrara-Boston, 2012)
  - number of bounding nodes / phases
  - surprisal
  - feature intervention
  - status of discourse referents
  - integration, retrieval, ...
- We want to keep the model **simple** (but not **trivial**):
  - Tenure and Size only refer to the geometry of the derivation
  - they are sensitive the specifics of tree-traversal (cf. node-count; Hale, 2001)

Why These Metrics?

- These complexity metrics are all related to **storage cost** (cf. Gibson, 1998)
- We could implement alternative ones (cf. Ferrara-Boston, 2012)
  - number of bounding nodes / phases
  - surprisal
  - feature intervention
  - status of discourse referents
  - integration, retrieval, ...
- We want to keep the model **simple** (but not **trivial**):
  - Tenure and Size only refer to the geometry of the derivation
  - they are sensitive the specifics of tree-traversal (cf. node-count; Hale, 2001)

Italian Subjects: Probing the Results

Clause Type	MaxT	SumS
obj. SRC	8/*che*	18
obj. ORC	11/*ha*	24
obj. ORCp	16/*Foc*	31
subj. SRC	21/v'	37
subj. ORC	21/v'	44
subj. ORCp	28/v'	56
matrix SVO	3/*ha*/v'	7
matrix VOS	7/*Top*/*Foc*	11
VS unacc	2/vP	3
VS unerg	7/*Top*/*Foc*	11

Table: Summary of MaxT (*value*/*node*) and SumS by construction. Obj. and subj. indicate the landing site of the RC head in the matrix clause.

Postverbal Asymmetries: Possible Accounts?

SRC > ORC

- DLT, active-filler strategy, Competition model, ...

ORC > ORCp

- more problematic (e.g., for DLT)
- can be explained by
  1. economy of gap prediction + structural re-analysis;
  2. intervention effects + featural Relativized Minimality

Can we give a purely structural account?

Postverbal Asymmetries: Possible Accounts?

SRC > ORC

- DLT, active-filler strategy, Competition model, ...

ORC > ORCp

- more problematic (e.g., for DLT)
- can be explained by
  1. economy of gap prediction + structural re-analysis;
  2. intervention effects + featural Relativized Minimality

Can we give a purely structural account?

Postverbal Asymmetries: Possible Accounts?

SRC > ORC

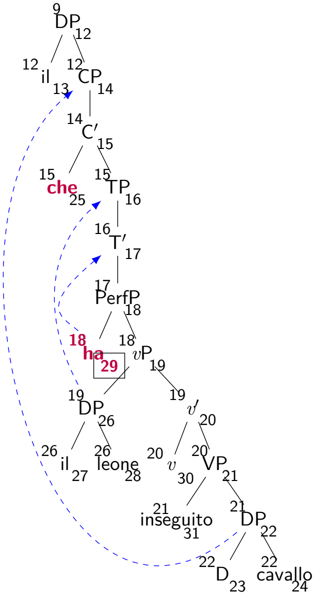
- DLT, active-filler strategy, Competition model, ...

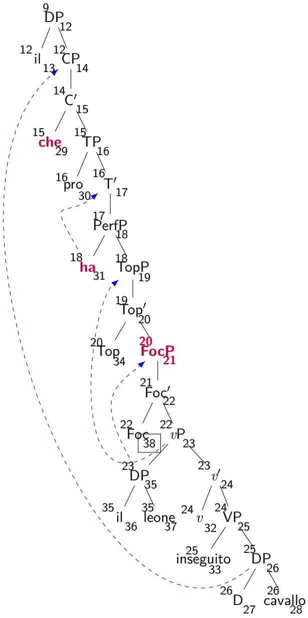
ORC > ORCp

- more problematic (e.g., for DLT)
- can be explained by
  1. economy of gap prediction + structural re-analysis;
  2. intervention effects + featural Relativized Minimality

Can we give a purely structural account?

Results: ORC > ORCp





Additional Constructions

- Ambiguity in Matrix Clauses

(7) Ha chiamato Gio
Has called Giovanni
a. "He/she/it called Gio" **SVO**
b. "Gio called" **VS**

- Unaccusatives vs. Unergatives

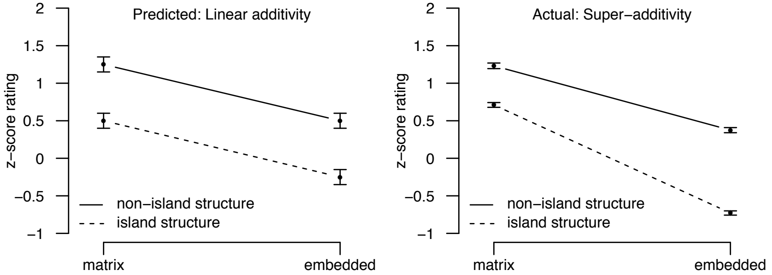
(8) È arrivato Gio
Is arrived Gio
"Gio arrived" **Unaccusative**

(9) Ha corso Gio
Has ran Gio
"Gio ran" **Unergative**

Gradience in Islands

A factorial design for islands effect:

- Gap Position × Structure



A Caveat on Island Effects

The Goal

Can gradience in acceptability judgments arise from a categorical grammar due to processing factors?

- Sprouse et al.'s (2012) design is ideal for the MG model.

But I am not interested in island effects *per se*:

- Islands: grammatical or processing effects? (Hofmeister et al., 2012a; Sprouse et al., 2012a,b)
  - hence, not modeling super-additivity
  - spoilers: maybe we get some insights?
- Islands: syntax or semantics? (Truswell, 2011; Kush et al., 2018; Matchin et al., 2018)

A Caveat on Island Effects

The Goal

Can gradience in acceptability judgments arise from a categorical grammar due to processing factors?

- Sprouse et al.'s (2012) design is ideal for the MG model.

But I am not interested in island effects *per se*:

- Islands: grammatical or processing effects? (Hofmeister et al., 2012a; Sprouse et al., 2012a,b)
  - hence, not modeling super-additivity
  - spoilers: maybe we get some insights?
- Islands: syntax or semantics? (Truswell, 2011; Kush et al., 2018; Matchin et al., 2018)

A Caveat on Island Effects

The Goal

Can gradience in acceptability judgments arise from a categorical grammar due to processing factors?

- Sprouse et al.'s (2012) design is ideal for the MG model.

But I am not interested in island effects *per se*:

- Islands: grammatical or processing effects?
  (Hofmeister et al., 2012a; Sprouse et al., 2012a,b)
  - hence, not modeling super-additivity
  - spoilers: maybe we get some insights?
- Islands: syntax or semantics?
  (Truswell, 2011; Kush et al., 2018; Matchin et al., 2018)

A Caveat on Island Effects

The Goal

Can gradience in acceptability judgments arise from a categorical grammar due to processing factors?

- Sprouse et al.'s (2012) design is ideal for the MG model.

But I am not interested in island effects *per se*:

- Islands: grammatical or processing effects? (Hofmeister et al., 2012a; Sprouse et al., 2012a,b)
  - hence, not modeling super-additivity
  - spoilers: maybe we get some insights?
- Islands: syntax or semantics? (Truswell, 2011; Kush et al., 2018; Matchin et al., 2018)

Technical Fertility of Derivation Trees

Derivation trees made it easy for MGs to accommodate the full syntactic toolbox:

- sidewards movement (Stabler, 2006; Graf 2013)
- affix hopping (Graf 2012; Graf2013)
- clustering movement (Gartner & Michaelis 2010)
- tucking in (Graf 2013)
- ATB movement (Kobele 2008)
- copy movement (Kobele 2006)
- extraposition (Hunter &Frank 2014)
- Late Merge (Kobele 2010; Graf 2014)
- Agree (Kobele 2011; Graf 2011)
- adjunction (Fowlie 2013; Hunter 2015)
- TAG-style adjunction (Graf 2012)

Implementation

Current Implementation available on **Github**. Salem

Allows to

- automatically compare MG derivation trees over a set of complexity metrics
- easily extendable with new metrics
- integrated with LaTeX

Main issues:

1. memory usage grows very fast with the number of metrics
2. tied to a specific parsing algorithm

Subject Islands

Case 1:

(10)
a. What do you think the speech interrupted *t*? — Obj | Non Island
b. What do you think *t* interrupted the show? — Subj | Non Island
c. What do you think the speech about global warming interrupted the show about *t*? — Obj | Island
d. What do you think the speech about *t* interrupted the show about global warming? — Subj | Island

Case 2:

(11)
a. Who *t* thinks the speech interrupted the primetime TV show? — Matrix | Non Island
b. What do you think *t* interrupted the primetime TV show? — Emb. | Non Island
c. Who *t* thinks the speech about global warming interrupted the primetime TV show? — Matrix | Island
d. What do you think the speech about *t* interrupted the primetime TV show? — Emb. | Island