Granularity in generative syntax and why it matters

Chenchen Julio Song, cs791@cam.ac.uk

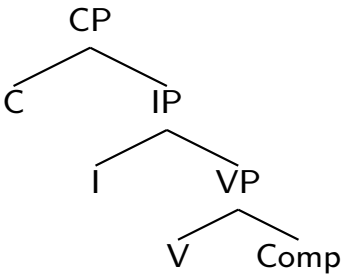
Theoretical and Applied Linguistics
University of Cambridge
Gonville & Caius College

SyntaxLab, 15 October 2019

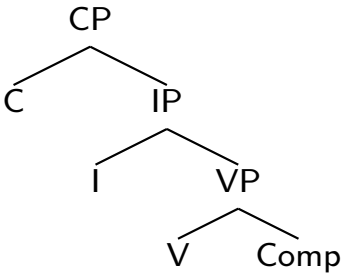
Overview

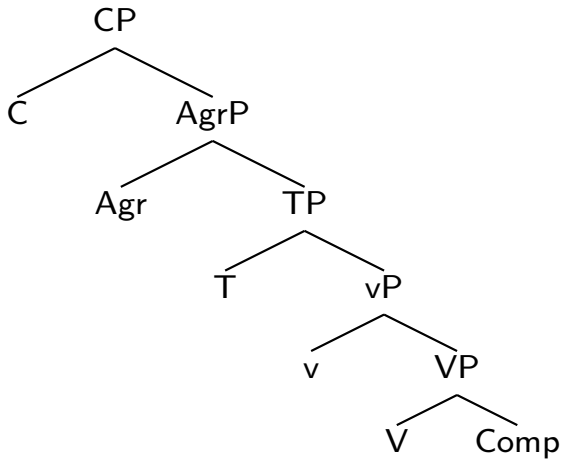
Progress

Trees grow

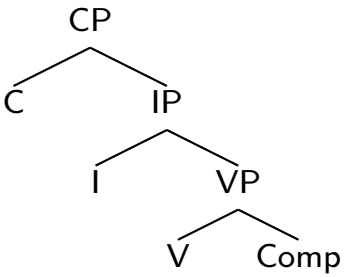


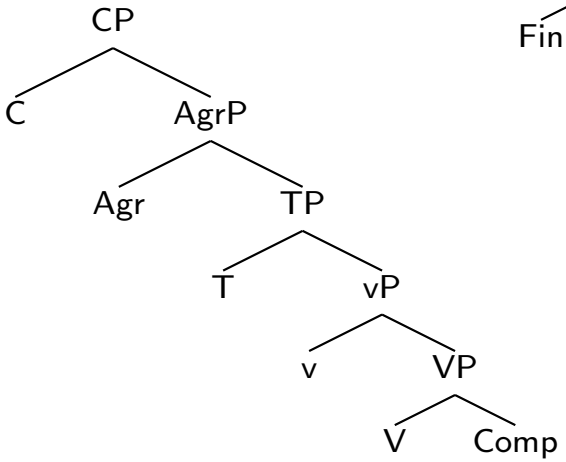
Trees grow





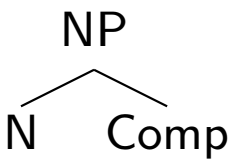
Trees grow


CP
C
IP
I
VP
V
Comp

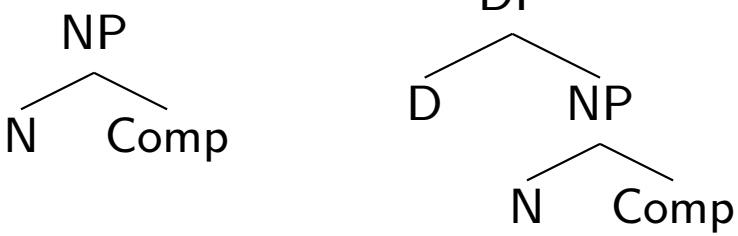

CP
C
AgrP
Agr
TP
T
vP
v
VP
V
Comp

TopP
Top
FocP
Foc
FinP
Fin
MoodP
Mood
TP
T
AspP
Asp
VoiceP
Voice
vP
v
ApplP
Appl
VP
V
Comp

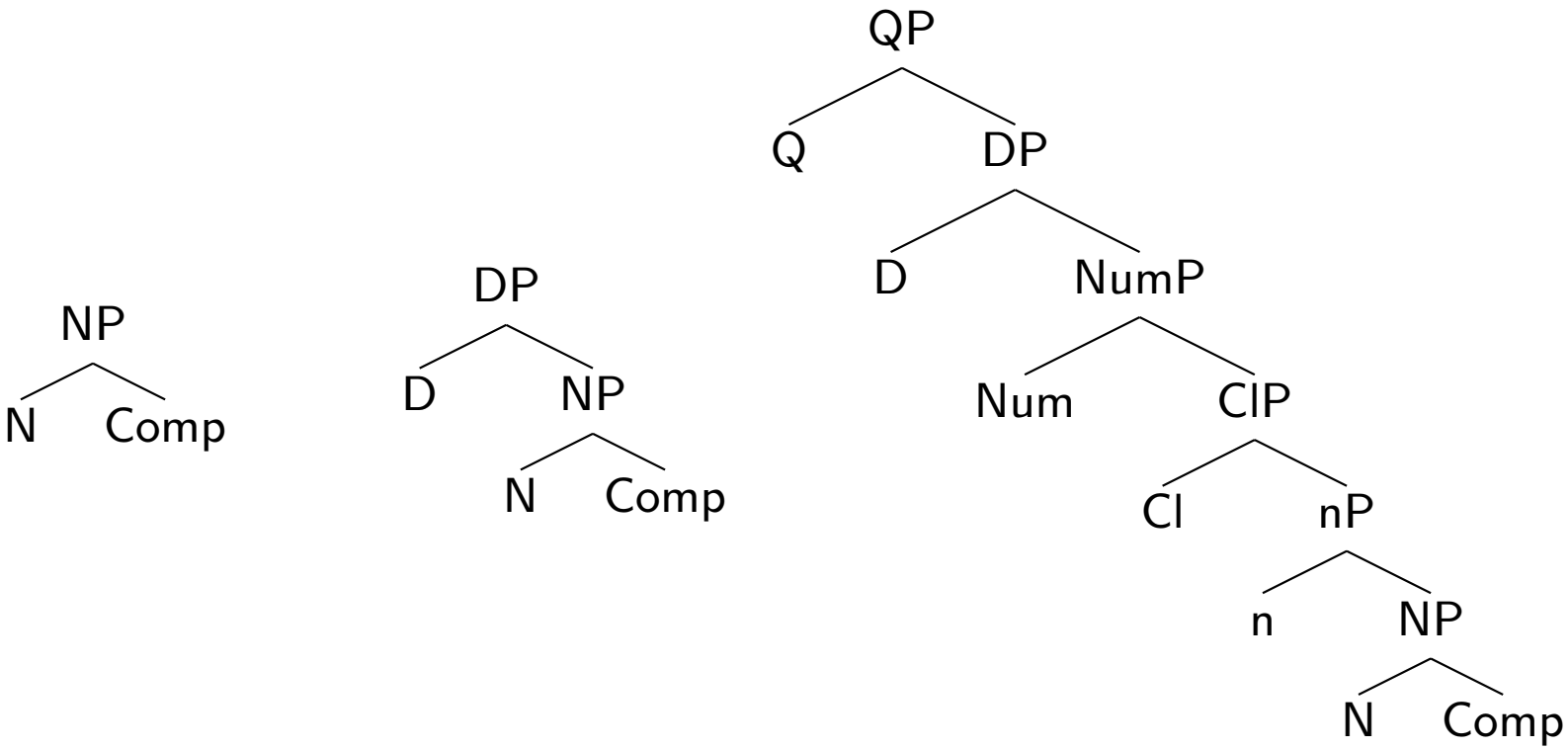
Trees grow



Trees grow



Trees grow



Trees grow

And the same has happened to adjectives and adpositions. . . [1]

- AdjP → [AgrP [DegP [QP [AP]]]]
- PP → [P_{Dir}P [P_{Stat}P [DegP . . . [DeicticP [AxPartP [PP]]]]]]]

Trees grow larger and larger in generative syntax, with more and more functional categories being proposed.

[1] See Corver (1997) and Cinque & Rizzi (2010).

Features expand

For example, the conceptions of NUMBER and PERSON have changed:

- [NUM:SG] → [NUM:[+ATOMIC, −AUGMENTED]] (Adger 2010)
- [PER:1ST] → [PER:[+AUTHOR, +PARTICIPANT]] (Harbour 2016)

Features expand

For example, the conceptions of NUMBER and PERSON have changed:

- [NUM:SG] → [NUM:[+ATOMIC, −AUGMENTED]] (Adger 2010)
- [PER:1ST] → [PER:[+AUTHOR, +PARTICIPANT]] (Harbour 2016)

So have the conceptions of lexical categories:

- Erstwhile: [CATEGORY:N/V]
- GB: [CATEGORY: ±N, ±V]
- Now: What do the symbols N and V really mean?
  - Panagiotidis (2015): two values (SORTAL and TEMPORAL) of a feature PERSPECTIVE
  - Biberauer & Roberts (2015): no universally fixed definitions

Observations 1 and 2

1. Trees grow—but they still describe the same kind of syntactic objects (sentences, noun phrases, etc.)
   e.g., *I saw a dog* can be analyzed either as a simple CP-IP-VP structure or as a cartographic structure.
2. Features expand—but they still describe the same kind of lexical items
   e.g., whether we describe *dog* by [NUM:SG] or [NUM:+ATOMIC,−AUGMENTED], we're still describing *dog*.

What have changed are not the objects of study but the analyses.
More specifically, what has changed is the *granularity* of analysis.

Observations 1 and 2

An overall trend in generative syntax in the past half century is *the increasing fine-grainedness of analyses*. This is witnessed by some popular research paradigms such as cartography and distributed morphology.[2]

- Cartography (Rizzi 1997, Cinque 1999, et seq.): accurately map the syntacticosemantic subtleties of natural language utterances
- Distributed morphology (Halle & Marantz 1993, 1994, et seq.): painstakingly decompose lexical items to their derivational atoms

[2] A question is: Are there limits to such granularity extension? (see Song 2019)

Progress

Granularity

Granularity is the **level of detailedness** in observation/description or analysis.
It is usually but not necessarily theoretically driven.

Observational/descriptive granularity

We can see more details with a microscope without considering any theoretical analysis.

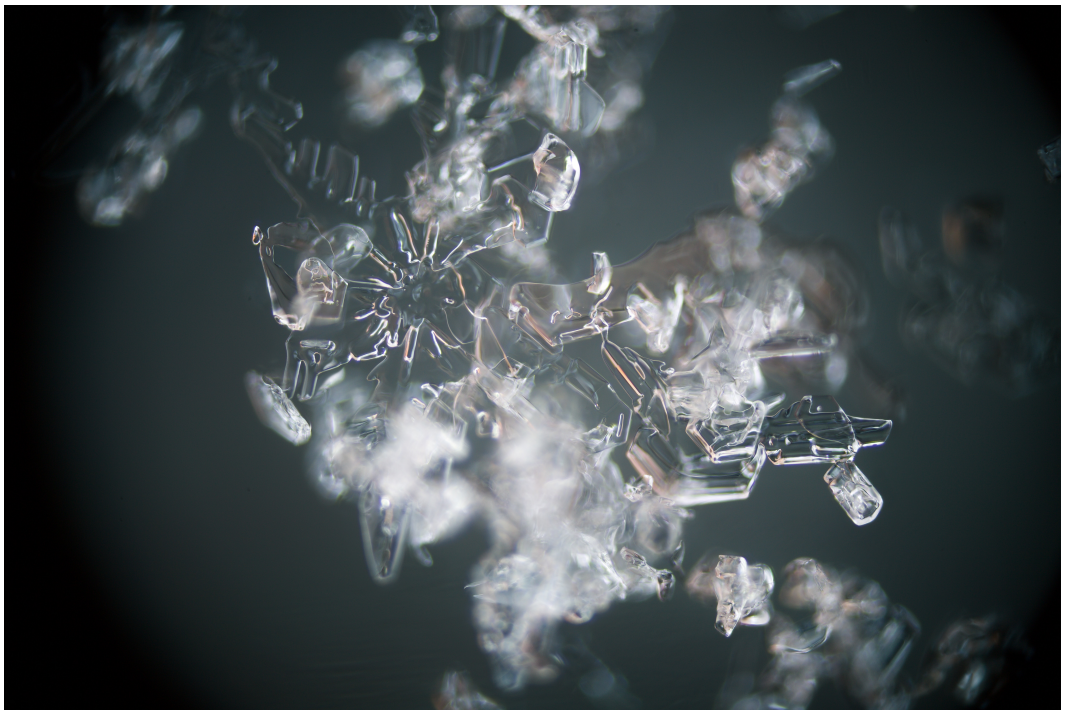


Figure 1: Frosted snow (photo by Andrii Ganzevych on Unsplash)

Explanatory granularity

However, as the descriptive granularity for a phenomenon increases, its explanatory granularity must also increase, because the newly revealed details become new explicanda.

Explanatory granularity

However, as the descriptive granularity for a phenomenon increases, its explanatory granularity must also increase, because the newly revealed details become new explicanda.

In other words, **descriptive granularity and explanatory granularity must match.**

Explanatory granularity

However, as the descriptive granularity for a phenomenon increases, its explanatory granularity must also increase, because the newly revealed details become new explicanda.

In other words, **descriptive granularity and explanatory granularity must match.**

In fact not only descriptions and their explanations but also different aspects of a single explanation must match in granularity.

Example: syntax and semantics

Montague (1974): There is an algebraic homomorphism from syntax to semantics.

Partee (2004): The meaning of an expression is a function of the meanings of its parts and of the way they are syntactically combined. (Frege's principle)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] =?, [\![\text{Voice}]\!] =?, [\![\text{Fin}]\!] =?$
  (an easier task; cf. Ramchand & Svenonius 2014)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] = ?, [\![\text{Voice}]\!] = ?, [\![\text{Fin}]\!] = ?$
  (an easier task; cf. Ramchand & Svenonius 2014)
- we must decide how to compose roots
  $[\![\sqrt{\text{DOG}}]\!] = ?, [\![\sqrt{\text{RUN}}]\!] = ?, [\![\text{n}, \sqrt{\text{DOG}}]\!] = ?$
  (a less easy task; cf. Kelly 2013, Song 2019)

Example: syntax and semantics

How to ensure the Montagovian "homomorphism" when we push syntactic analyses to higher levels of granularity? Among others,

- we must assign denotations to numerous new functional categories
  $[\![\text{Appl}]\!] = ?, [\![\text{Voice}]\!] = ?, [\![\text{Fin}]\!] = ?$
  (an easier task; cf. Ramchand & Svenonius 2014)
- we must decide how to compose roots
  $[\![\sqrt{\text{DOG}}]\!] = ?, [\![\sqrt{\text{RUN}}]\!] = ?, [\![\text{n}, \sqrt{\text{DOG}}]\!] = ?$
  (a less easy task; cf. Kelly 2013, Song 2019)

This is essentially a *model extension*. And if we view syntactic derivations as formal proofs (following Chomsky 1965, 2007), then this extension must not damage the well-formedness of the proof system (with respect to soundness, completeness, etc.).[3]

[3] A syntactic derivation system is *sound* if it can only derive/prove semantically valid sentences and *complete* if all semantically valid sentences can be derived/proved in it.

Example: syntax and semantics

Moreover, the model extension must meet the "interface condition" (Chomsky 2004) on a conceptual level; that is, we can't assign denotations to terms merely based on model-theoretic needs but should also ask "What's the conceptual interpretability?"

Example: syntax and semantics

Moreover, the model extension must meet the "interface condition" (Chomsky 2004) on a conceptual level; that is, we can't assign denotations to terms merely based on model-theoretic needs but should also ask "What's the conceptual interpretability?"

Compare categorial grammar and minimalism:

- Categorial grammar: an intransitive verb is a category that when combined with NP yields S [purely externalist solution]
- Minimalism: a verb can be assigned a denotation $e \rightarrow t$ but its interface interpretation is more than that (e.g., Panagiotidis' "extending-in-time") [half-externalist-half-internalist solution]

Functional categories

Just as purely morphosyntactic considerations may lead to "uninterpretable" categories (e.g., Chomsky's 1995 criticism of Agr), so purely model-theoretic considerations may lead to "proofs" of conceptually vague or void categories.

Functional categories

Just as purely morphosyntactic considerations may lead to "uninterpretable" categories (e.g., Chomsky's 1995 criticism of Agr), so purely model-theoretic considerations may lead to "proofs" of conceptually vague or void categories.

e.g., Rubin's (2003) definition of Mod (an "adjunct shell")

- Syntax: $[_{\text{XP}}\ [_{\text{ModP}}\ \text{Mod YP}]\ \text{XP}]$
- Semantics: $[\![\text{XP}]\!] = e \rightarrow \text{t}$ (a predicate) $\Rightarrow [\![\text{ModP}]\!] = (e \rightarrow \text{t}) \rightarrow (e \rightarrow \text{t})$ (a predicate modifier) $\Rightarrow$ if $[\![\text{YP}]\!] = e \rightarrow \text{t}$ (another predicate), then $[\![\text{Mod}]\!] = (e \rightarrow \text{t}) \rightarrow ((e \rightarrow \text{t}) \rightarrow (e \rightarrow \text{t}))$ (a category that converts a predicate into a predicate modifier)

Functional categories

Just as purely morphosyntactic considerations may lead to "uninterpretable" categories (e.g., Chomsky's 1995 criticism of Agr), so purely model-theoretic considerations may lead to "proofs" of conceptually vague or void categories.

e.g., Rubin's (2003) definition of Mod (an "adjunct shell")

- Syntax: $[_{\text{XP}}\ [_{\text{ModP}}\ \text{Mod YP}]\ \text{XP}]$
- Semantics: $[\![\text{XP}]\!] = e \to \text{t}$ (a predicate) $\Rightarrow$ $[\![\text{ModP}]\!] = (e \to \text{t}) \to (e \to \text{t})$ (a predicate modifier) $\Rightarrow$ if $[\![\text{YP}]\!] = e \to \text{t}$ (another predicate), then $[\![\text{Mod}]\!] = (e \to \text{t}) \to ((e \to \text{t}) \to (e \to \text{t}))$ (a category that converts a predicate into a predicate modifier)

How different is this definition from that of intransitive verbs as "a category that when combined with NP yields S"? (Not really.)

This style of definition defines what a category **does** but not what it **is**.

Roots

Now let's turn to roots. Syntacticians (especially DMers) have decomposed content words like *dog* and *run* into roots and categorizers, but this hasn't had much influence on semanticists, who mostly still treat bare nouns and verbs as (typed) predicates.

- $[\![\text{dog}]\!] = \lambda x.\text{DOG}(x)$
- $[\![\text{run}]\!] = \lambda e.\text{RUN}(e)$

Roots

Lexical decomposition in syntax corresponds to predicate decomposition in semantics, but the latter hasn't reached the root level.

- Jones buttered the toast slowly in the bathroom with a knife.
- $\exists e.\text{BUTTER}(e) \wedge \text{AGENT}(e) = \text{John} \wedge \text{THEME}(e) = \text{TOAST} \wedge \text{SLOWLY}(e) \wedge \text{LOCATION}(e) = \text{BATHROOM} \wedge \text{INSTRUMENT}(e) = \text{KNIFE}$ (Landman 2000)

Roots

Lexical decomposition in syntax corresponds to predicate decomposition in semantics, but the latter hasn't reached the root level.

- Jones buttered the toast slowly in the bathroom with a knife.
- $\exists e.\text{BUTTER}(e) \wedge \text{AGENT}(e) = \text{John} \wedge \text{THEME}(e) = \text{TOAST} \wedge \text{SLOWLY}(e) \wedge \text{LOCATION}(e) = \text{BATHROOM} \wedge \text{INSTRUMENT}(e) = \text{KNIFE}$ (Landman 2000)

If DMers want to develop a comprehensive theory of lexical decomposition, they need a semantics with matching granularity for the sake of Frege's principle (see Kelly 2013 for a proposal).

Example: syntax and semantics

In sum, both exemplary paradigms of high-granularity syntax (cartography and distributed morphology) require some effort in achieving an adequate semantics with matching granularity.

Example: syntax and semantics

In sum, both exemplary paradigms of high-granularity syntax (cartography and distributed morphology) require some effort in achieving an adequate semantics with matching granularity.

Both cartography and DM are about **tree growth**. Does the same sort of concern also arise in **feature expansion**? (Yes.)

Features

How do individual features together describe lexical items? The usual format is a *record-like data structure* (to borrow a term from HPSG foundations; Ait-Kaci 1984).

- *he*: [PER:3RD, NUM:SG, GEN:MASC]
- *runs*: [CATEGORY:V, PER:3RD, NUM:SG]

Features

How do individual features together describe lexical items? The usual format is a *record-like data structure* (to borrow a term from HPSG foundations; Ait-Kaci 1984).

- *he*: [PER:3RD, NUM:SG, GEN:MASC]
- *runs*: [CATEGORY:V, PER:3RD, NUM:SG]

How are such featural descriptions interpreted? The usual mode of composition is simply *conjunction*.

- $[\![\text{PER:3RD, NUM:SG, GEN:MASC}]\!] = [\![\text{PER:3RD}]\!] \wedge [\![\text{NUM:SG}]\!] \wedge [\![\text{GEN:MASC}]\!]$
- $[\![\text{CATEGORY:V, PER:3RD, NUM:SG}]\!] =$
  $[\![\text{CATEGORY:V}]\!] \wedge [\![\text{PER:3RD}]\!] \wedge [\![\text{NUM:SG}]\!]$

Features

But the record-and-conjunction-based feature interpretation has two prerequisites:

1. All features in a record must be of the same type (e.g., first-order predicate), because conjunction requires type matching.
2. All features in a record must have parallel status (i.e., no hierarchical structure), because conjunction is commutative.

Features

But the record-and-conjunction-based feature interpretation has two prerequisites:

1. All features in a record must be of the same type (e.g., first-order predicate), because conjunction requires type matching.
2. All features in a record must have parallel status (i.e., no hierarchical structure), because conjunction is commutative.

This is at odds with the kind of tree structure familiar in Chomskyan syntax, especially with the branch that argues for "syntax (aka Merge) all the way down" (see Tsoulas 2017 for feature-related formulation). It is more suitable for a unification-based syntactic framework instead.

Two types of granularity lifting

Record-and-conjunction-based feature expansion is of a different type of granularity lifting from that in cartography/DM. I call them **paradigmatic granularity** and **syntagmatic granularity** respectively (Song 2019).

- ↑Paradigmatic granularity: redefine a feature by a list of other features
  e.g., [SG] → [+ATOMIC, −AUGMENTED]
- ↑Syntagmatic granularity: redefine a category by a subtree of other categories
  e.g., IP → AgrP-TP, CP → TopP-FocP-FinP

A minimalist complication

Feature integration does not always fall in a record-and-conjunction style.

- Some features are not designed to denote (first-order) predicates
- Sometimes the ordering of features matters (i.e., noncommutative integration)

A minimalist complication

Feature integration does not always fall in a record-and-conjunction style.

- Some features are not designed to denote (first-order) predicates
- Sometimes the ordering of features matters (i.e., noncommutative integration)

For example, Harbour's (2016) lattice-theoretic person features:

- $[\![+\textsc{author}(\pi)]\!] = [\![\textsc{author}]\!] \oplus [\![\pi]\!] = \{\mathfrak{a} \sqcup \mathfrak{b} : \mathfrak{a} \in \mathcal{L}_{\mathfrak{au}}, \mathfrak{b} \in \mathcal{L}_{\pi}\}$
- $[\![-\textsc{author}(\pi)]\!] = [\![\textsc{author}]\!] \ominus [\![\pi]\!] = \{\mathfrak{b} \setminus \mathfrak{max}(\mathcal{L}_{\mathfrak{au}}) : \mathfrak{b} \in \mathcal{L}_{\pi}\}$

where person features denote lattice-theoretic structures and $+/-$ denote actions on those structures.

Features

So, at least in minimalism, feature expansion (i.e., paradigmatic granularity lifting) also requires some effort in achieving a semantics with matching granularity, because

- there's no fixed denotation pattern for features, and
- there's no fixed mode of composition for features.

Interim summary

Granularity $\triangleq$ level of detailedness

Two types of granularity in general: descriptive and explanatory.

Two types of granularity matching: (i) descriptive-explanatory, (ii) different aspects of an explanatory theory (group).

It is not easy to maintain granularity matching, as illustrated by the various thorny issues arising from the syntax-semantics example. . . which may partly explain why both syntacticians and semanticists prefer focusing on one side of the story.

Progress

Why does granularity matter?

1. Granularity is **omnipresent** in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.

Why does granularity matter?

1. Granularity is **omnipresent** in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.
2. Granularity is **flexible** in generative syntax: Different analyses may assume different levels of granularity.
   e.g., C-T-v-V vs. cartography

Why does granularity matter?

1. Granularity is **omnipresent** in generative syntax: Every analysis assumes some level of granularity. ☜ Bear this in mind.
2. Granularity is **flexible** in generative syntax: Different analyses may assume different levels of granularity.
   e.g., C-T-v-V vs. cartography
3. Granularity is **usually left implicit** in generative syntax: It's sometimes a matter of trend and sometimes to-each-his-own.
   e.g., C-Agr-T-Asp-v-V vs. C-T-Asp-Voice-Appl-V

Why does granularity matter?

4. Analyses of different granularity levels often overlap in labels—though the granularity difference means that the overlapping labels cannot have identical definitions.
   e.g., "v" in C-T-v-V $\neq$ "v" in C-Agr-T-Asp-Voice-v-Appl-V

Why does granularity matter?

4. Analyses of different granularity levels often overlap in labels—though the granularity difference means that the overlapping labels cannot have identical definitions.
   e.g., "v" in C-T-v-V $\neq$ "v" in C-Agr-T-Asp-Voice-v-Appl-V
5. Given 1–4, granularity mismatches may lead to fundamental misunderstandings of terms/concepts, and a lack of awareness of granularity is detrimental to theoretical comparison and integration.
   e.g., if my "v" isn't your "v" how do we know we are arguing about the same thing? Short answer: We don't know and that's confusing!
   (see, e.g., D'Alessandro, Franco & Gallego 2017)

Why does granularity matter?

4. Analyses of different granularity levels often overlap in labels—though the granularity difference means that the overlapping labels cannot have identical definitions.
   e.g., "v" in C-T-v-V $\neq$ "v" in C-Agr-T-Asp-Voice-v-Appl-V
5. Given 1–4, granularity mismatches may lead to fundamental misunderstandings of terms/concepts, and a lack of awareness of granularity is detrimental to theoretical comparison and integration.
   e.g., if my "v" isn't your "v" how do we know we are arguing about the same thing? Short answer: We don't know and that's confusing!
   (see, e.g., D'Alessandro, Franco & Gallego 2017)

The above reasons are mainly methodological (next we'll see a more conceptual reason).

Progress

Why does granularity matter? (Continued)

6. Granularity can be abstracted into a more general notion.

Why does granularity matter? (Continued)

6. Granularity can be abstracted into a more general notion.

Recall: Granularity is omnipresent in generative syntax.

Why does granularity matter? (Continued)

6. Granularity can be abstracted into a more general notion.

Recall: Granularity is omnipresent in generative syntax.

A granularity level Γ can be viewed as the **ambient categorial context** for a derivational analysis $\mathcal{A}$.

$$\boxed{\Gamma \vdash \mathcal{A}}^{4}$$

[4] I use this notation because this conception of granularity is somewhat similar to the *typing context* in type theory.

Why does granularity matter? (Continued)

6. Granularity can be abstracted into a more general notion.

Recall: Granularity is omnipresent in generative syntax.

A granularity level Γ can be viewed as the **ambient categorial context** for a derivational analysis $\mathcal{A}$.

$$\boxed{\Gamma \vdash \mathcal{A}}\ ^{4}$$

In other words, a granularity level in the abstract sense is just a *background categorial setting*. A granularity level is **completely defined** by the **syntactic categories** it consists of as well as their **individual definitions**.

[4] I use this notation because this conception of granularity is somewhat similar to the *typing context* in type theory.

Why does granularity matter? (Continued)

6. Granularity can be abstracted into a more general notion.

Recall: Granularity is omnipresent in generative syntax.

A granularity level Γ can be viewed as the **ambient categorial context** for a derivational analysis A.

$$\boxed{\Gamma \vdash A}\ ^{4}$$

In other words, a granularity level in the abstract sense is just a *background categorial setting*. A granularity level is **completely defined** by the **syntactic categories** it consists of as well as their **individual definitions**.

This abstract notion of granularity encompasses syntagmatic and paradigmatic granularity.

[4] I use this notation because this conception of granularity is somewhat similar to the *typing context* in type theory.

Typing context (Nederpelt & Geuvers 2014)

In type theory, a *typing context* (or simply *context*) is a (possibly empty) list of typing statements for variables.
e.g., $\Gamma \triangleq x_1 : \alpha, x_2 : \alpha \to \beta, x_3 : (\beta \to \alpha) \to \beta$

A *judgment* $\Gamma \vdash M : \sigma$ is derivable iff M has type σ in context Γ.
e.g., given the above context, $\Gamma \vdash x_2 x_1 : \beta$ is derivable

Usually when one says "M has type σ" a context is always assumed in the background.

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Example:

- $\Gamma \vdash x_2 x_1 : \beta$ ($x_2 x_1 : \beta$ is derivable in context Γ)
- $\Gamma \vdash$ *I saw a dog* :
  $[_{CP}$ C $[_{TP}$ $[_{DP}$ I] $[_{TP}$ see$_i$-T$_{Past}$ $[_{vP}$ v $[_{VP}$ $[_{V}$ t$_i$] $[_{DP}$ a dog]]]]]]
  (this structure is derivable in granularity Γ)

Granularity level vs. typing context

Similarities:

1. Both are omnipresent.
2. Both are flexible.
3. Both are often left implicit.

Example:

- $\Gamma \vdash x_2 x_1 : \beta$ ($x_2 x_1 : \beta$ is derivable in context Γ)
- $\Gamma \vdash$ *I saw a dog* :
  $[_{CP}$ C $[_{TP}$ $[_{DP}$ I] $[_{TP}$ see$_i$-T$_{Past}$ $[_{vP}$ v $[_{VP}$ $[_{V}$ t$_i$] $[_{DP}$ a dog]]]]]]
  (this structure is derivable in granularity Γ)

Differences:

1. Label overlapping is less of a problem in type theory, where type symbols are purely formal (e.g., there's no "interface condition").
2. Logicians are very aware of the existence and flexibility of typing contexts (linguists should learn from them).

Granularity level redefined

A granularity level is a **set** of well-defined syntactic categories that can serve as the "typing context" of a syntactic derivation system.

Granularity level redefined

A granularity level is a **set** of well-defined syntactic categories that can serve as the "typing context" of a syntactic derivation system.

In other words, a granularity level is just a **functioning inventory** of syntactic categories (where *functioning* means that a usable[5] derivation system can be built based on the given inventory). For example:

- The inventory used in GB.
- The inventory used in standard minimalist program (Chomsky 1995).
- The inventory used in cartography.

[5] An ideal derivation system should be sound and complete, but considering derivational theories proposed for natural languages are often imperfect or under constant improvement, we can relax the criterion of "functioning" to being usable.

Granularity level redefined

A granularity level is a **set** of well-defined syntactic categories that can serve as the "typing context" of a syntactic derivation system.

In other words, a granularity level is just a **functioning inventory** of syntactic categories (where *functioning* means that a usable[5] derivation system can be built based on the given inventory). For example:

- The inventory used in GB.
- The inventory used in standard minimalist program (Chomsky 1995).
- The inventory used in cartography.

All such granularity levels are "the same" in that they all serve to describe the same object (i.e., natural language syntax).

[5] An ideal derivation system should be sound and complete, but considering derivational theories proposed for natural languages are often imperfect or under constant improvement, we can relax the criterion of "functioning" to being usable.

Generalization

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

Generalization

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

So, not only GB, standard MP, and cartography but also various categorial inventories of intermediate sizes (e.g., those between standard MP and cartography) can define granularity levels.

- Top-Foc-Fin-Mood-T-Asp-v-V...
- C-T-Asp-Voice-v-Appl-V...
- C-T-Init-Proc-Res...

Generalization

Our redefinition of granularity is a rather broad one. **Any** functioning inventory of syntactic categories can define a granularity level.

So, not only GB, standard MP, and cartography but also various categorial inventories of intermediate sizes (e.g., those between standard MP and cartography) can define granularity levels.

- Top-Foc-Fin-Mood-T-Asp-v-V...
- C-T-Asp-Voice-v-Appl-V...
- C-T-Init-Proc-Res...

And syntacticians can indeed freely choose whichever granularity level they like as the background categorial setting of their analyses.
e.g., someone studying the C-domain may choose to only split CP, and someone studying the V-domain may choose to only split VP.

Granularity level space (GLS)

In theory there can be a lot of granularity levels for natural language syntax, whose totality can be conceived as a *granularity level space*.

The granularity level space can be defined as the **set** of all possible granularity levels for human language syntax. This is more precisely a *partially ordered set* or *poset*.

Granularity level space (GLS)

In theory there can be a lot of granularity levels for natural language syntax, whose totality can be conceived as a *granularity level space*.

The granularity level space can be defined as the **set** of all possible granularity levels for human language syntax. This is more precisely a *partially ordered set* or *poset*.

If a granularity level Γ is less fine-grained than or equal to another granularity level Γ' we can write $\Gamma \leqslant \Gamma'$.
e.g., the standard MP granularity $\leqslant$ the cartographic granularity

Progress

Granularity in I-language

So far we have only looked at granularity variation from an analyst's perspective. Does granularity also have a place in I-language, where speakers do not have the kind of "cross-granularity" perspective that linguists have?

Granularity in I-language

So far we have only looked at granularity variation from an analyst's perspective. Does granularity also have a place in I-language, where speakers do not have the kind of "cross-granularity" perspective that linguists have?

Yes. At a specific point (or period) in time, a speaker's I-language only has a particular granularity level, because it only has a particular inventory of categories.

The granularity level of a speaker's I-language may change **over time**, especially during its development/maturing process.

Granularity in I-language

The change of granularity in I-language is usually **increasing**.[6]

[6] Can it ever be decreasing (e.g., in language pathology)?

Granularity in I-language

The change of granularity in I-language is usually **increasing**.[6]

Consider Biberauer & Roberts' (2015) category subtyping hierarchy:

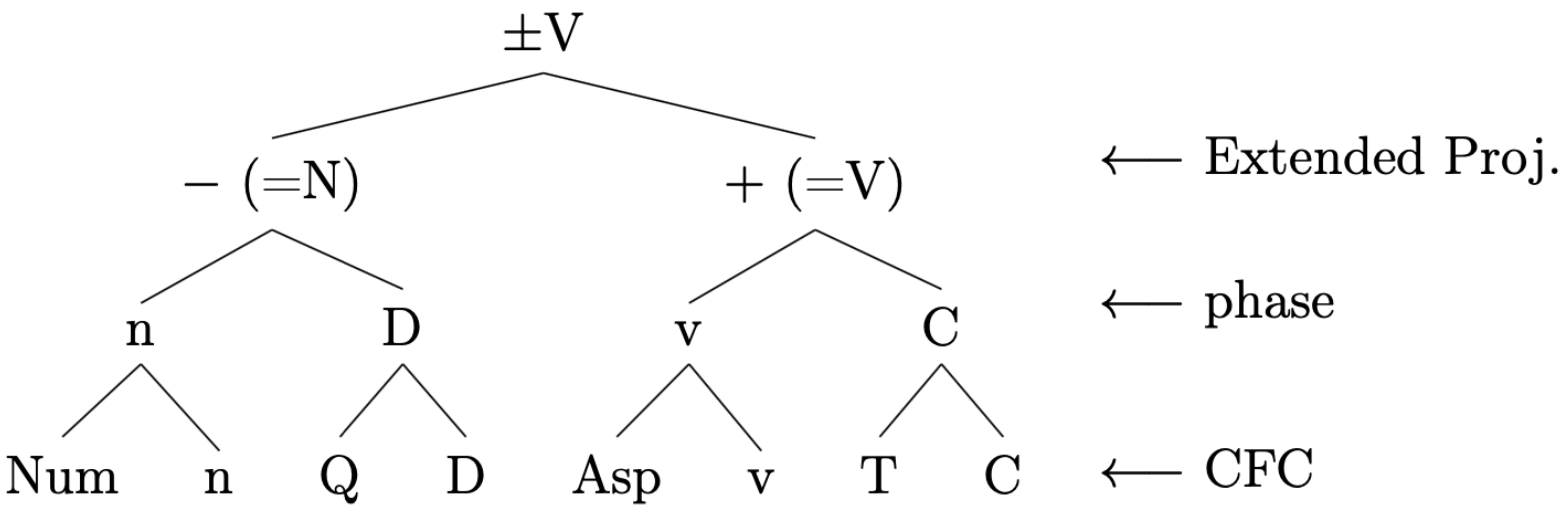


Figure 2: B&R's (2015) different levels of "magnification" for syntactic analyses

[6] Can it ever be decreasing (e.g., in language pathology)?

Granularity in I-language

The change of granularity in I-language is usually **increasing**.[6]

Consider Biberauer & Roberts' (2015) category subtyping hierarchy:

$$\Gamma_{EP} \leqslant \Gamma_{Ph} \leqslant \Gamma_{CFC}$$

Each Γ corresponds to a row in B&R's hierarchy, so (an abstracted form of) the B&R hierarchy is a corner in the GLS.

[6] Can it ever be decreasing (e.g., in language pathology)?

Broad vs. narrow granularity

NB: The generalized notion of granularity level **no longer depends on** a cross-granularity comparative viewpoint; it is just a synonym for "categorial inventory" instead.

So it makes sense to talk about granularity in an I-language discourse even though speakers lack the "linguist's view."

Broad vs. narrow granularity

NB: The generalized notion of granularity level **no longer depends on** a cross-granularity comparative viewpoint; it is just a synonym for "categorial inventory" instead.

So it makes sense to talk about granularity in an I-language discourse even though speakers lack the "linguist's view."

The term *granularity* merely highlights the fact that the various categorial inventories for natural languages are **interconnected** to one another (e.g., they fall in an overarching partial order).

Broad vs. narrow granularity

NB: The generalized notion of granularity level **no longer depends on** a cross-granularity comparative viewpoint; it is just a synonym for "categorial inventory" instead.

So it makes sense to talk about granularity in an I-language discourse even though speakers lack the "linguist's view."

The term *granularity* merely highlights the fact that the various categorial inventories for natural languages are **interconnected** to one another (e.g., they fall in an overarching partial order).

To avoid misunderstanding, I call this generalized granularity **granularity in the broad sense**; accordingly, the originally conceived, comparison-based granularity is **granularity in the narrow sense**.

Granularity in I-language

Individual speakers are naturally equipped with granularity levels in the **broad** sense, while linguists can reason about all sorts of granularity levels, either broad or narrow.

It's not always easy to distinguish the two senses of granularity.

Granularity in I-language

Consider the "domain categories" in Chomskyan syntax (e.g., C, I, V). Is a granularity level Γ_{Dom} made up of such categories broad or narrow?

Granularity in I-language

Consider the "domain categories" in Chomskyan syntax (e.g., C, I, V). Is a granularity level Γ_{Dom} made up of such categories broad or narrow?

It depends.

1. If these categories are all there have formed in a speaker's I-language, then as long as the speaker can already make use of them, Γ_{Dom} (being a functioning inventory of syntactic categories) is granularity in the broad sense.
2. If these categories are just cover terms used by linguists to gloss over spans of categories they don't want to bother spelling out, then Γ_{Dom} is granularity in the narrow sense (because it entails the existence of and is built on another "magnified" granularity level).

Granularity in I-language

Consider the "domain categories" in Chomskyan syntax (e.g., C, I, V). Is a granularity level Γ_{Dom} made up of such categories broad or narrow?

In the spirit of the B&R hierarchy (i.e., category subtyping), Γ_{Dom} is either a coarse granularity level that hasn't gone through further splitting or a coarse granularity level that has gone through further splitting but can be referenced for analytical purposes. Of course, this kind of "back-referencing" can only be done by linguists (including amateurs[7]), not by speakers.

[7] Sometimes nonlinguists can actively reason about grammars too, but then they can't be considered "naive speakers."

Granularity in I-language

Consider the "domain categories" in Chomskyan syntax (e.g., C, I, V). Is a granularity level Γ_{Dom} made up of such categories broad or narrow?

In the spirit of the B&R hierarchy (i.e., category subtyping), Γ_{Dom} is either a coarse granularity level that hasn't gone through further splitting or a coarse granularity level that has gone through further splitting but can be referenced for analytical purposes. Of course, this kind of "back-referencing" can only be done by linguists (including amateurs[7]), not by speakers.

The two senses of Γ_{Dom} are formally identical (i.e., they have identical extensions) but conceptually different (i.e., they have different intensions).

We can call granularity in the speaker's mind **mental granularity** and granularity in linguists' practice **analytical granularity**.

[7] Sometimes nonlinguists can actively reason about grammars too, but then they can't be considered "naive speakers."

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

It's tricky and depends on our definition of "phase." In particular, at what granularity level does phase-based derivation become relevant?

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

It's tricky and depends on our definition of "phase." In particular, at what granularity level does phase-based derivation become relevant?

It's surely already at work in the standard MP system (Chomsky 1995), let alone in cartography, but what about the granularity level where the phase-categories-to-be have just been formed and haven't further split yet?

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

It's tricky and depends on our definition of "phase." In particular, at what granularity level does phase-based derivation become relevant?

It's surely already at work in the standard MP system (Chomsky 1995), let alone in cartography, but what about the granularity level where the phase-categories-to-be have just been formed and haven't further split yet?

If phases are active at Γ_{Ph}, then each merger there would be a phase. . . How desirable is that?

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

It's tricky and depends on our definition of "phase." In particular, at what granularity level does phase-based derivation become relevant?

It's surely already at work in the standard MP system (Chomsky 1995), let alone in cartography, but what about the granularity level where the phase-categories-to-be have just been formed and haven't further split yet?

If phases are active at Γ_{Ph}, then each merger there would be a phase. . . How desirable is that?

A more sensible take seems to be that phases only become significant when the categorial inventory becomes fine-grained enough.

Granularity in I-language

What about phase categories (e.g., v, C)? Is a granularity level Γ_{Ph} solely made up of such categories broad or narrow? Mental or analytical?

- It's broad/mental if it represents a complete I-language categorial inventory (and in this case the term "phase" perhaps makes no sense).
- It's narrow if it's being back-referenced by linguists.
- It's analytical if it's used in linguistic reasoning, either in the broad or the narrow sense.

Granularity in I-language

NB: "C" occurs in both Γ_{Ph} and Γ_{Dom} but has crucially different analytical senses (recall that such label overlapping is common in generative syntax).

Granularity in I-language

NB: "C" occurs in both Γ_{Ph} and Γ_{Dom} but has crucially different analytical senses (recall that such label overlapping is common in generative syntax).

In the narrow analytical sense:

- C_{Ph} is a specific category signaling cyclic spell-out (it can participate in derivation)
- C_{Doc} is a cover term for a set of categories (it can't participate in derivation as such because derivational input can only be individual categories or "first-order" but not sets of categories or "second-order").

Granularity in I-language

NB: "C" occurs in both Γ_{Ph} and Γ_{Dom} but has crucially different analytical senses (recall that such label overlapping is common in generative syntax).

In the narrow analytical sense:

- C_{Ph} is a specific category signaling cyclic spell-out (it can participate in derivation)
- C_{Doc} is a cover term for a set of categories (it can't participate in derivation as such because derivational input can only be individual categories or "first-order" but not sets of categories or "second-order").

So what do people mean when they write "C"? It depends on the context (i.e., granularity)—hence the importance of a granularity awareness!

Granularity level space organization

Recall: the totality of granularity levels for natural language syntax form the *granularity level space* (GLS).

The space has a highly organized structure that can be made mathematically explicit (see Song 2019). In particular, there are multiple perspectives to organize granularity levels:

- poset based on fine-graineness (e.g., "less fine-grained or equal to")
- correspondences induced by category subtyping (e.g., granularity levels of a single "subtyping line" [i.e., a single B&R hierarchy] are all "isomorphic" at a certain level of abstraction)

Granularity level space organization

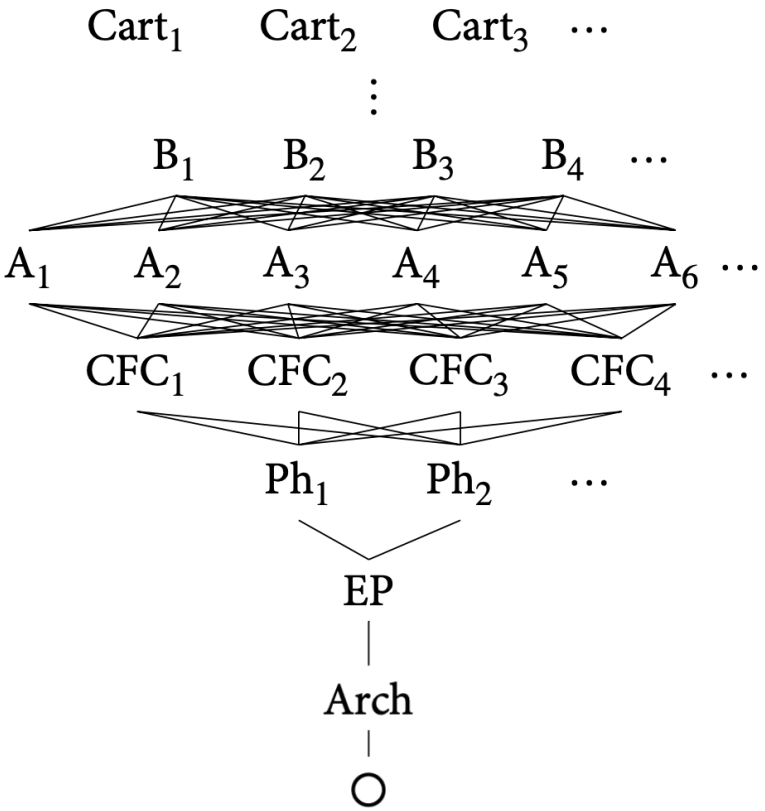


Figure 2: GLS partially ordered by coarser-than-or-equal-to (Song 2019:194)[8]

[8] Each node represents a granularity level.

Granularity level space organization

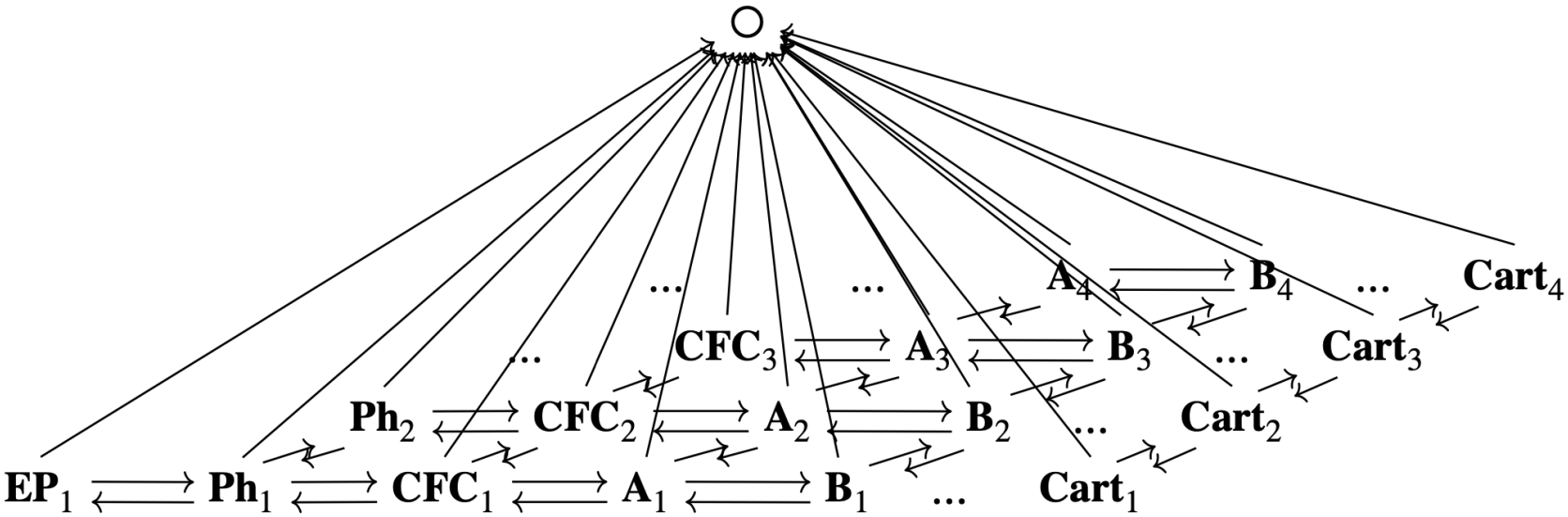


Figure 3: GLS organized by inheritance-induced isomorphism (Song 2019:224)[9]

[9]Each node represents a granularity level.

Granularity level space organization

And clearly, when we talk about granularity level space organization, we are talking as linguists and hence about analytical granularity.

The GLS is an extremely macro-level issue (i.e., of a very high level of abstraction). When reasoning about inter-granularity-level connection, we simply ignore the internal details of each granularity level.

Granularity level space organization

If we take a granularity level (say CFC_1) and put it under our "magnifying glass," we can see those details again.

Granularity level space organization

If we take a granularity level (say CFC_1) and put it under our "magnifying glass," we can see those details again.

- CFC_1 →Q→ {**V**, **N**}[10] →Q→ {{V, v, T, C}, {N, n, Num, D}}

One of the conclusions in Song (2019) is that many such granularity-level-internal details can be reformulated as cross-granularity-level connections (e.g., functions).

[10] These boldfaced labels represent part-of-speech domains (i.e., names of sets).

Progress

Summary

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in the granularity of syntactic analyses.

Summary

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in the granularity of syntactic analyses.
2. There are two ways to define granularity: (1) level of detailedness (narrow), (2) inventory of syntactic categories (broad).

Summary

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in the granularity of syntactic analyses.
2. There are two ways to define granularity: (1) level of detailedness (narrow), (2) inventory of syntactic categories (broad).
3. There are different ways to understand granularity (i.e., granularity is a multifaceted notion): (1) broad vs. narrow, (2) descriptive vs. explanatory, (3) syntagmatic vs. paradigmatic, (4) mental vs. analytical.

Summary

1. Two basic observations about generative syntax: (1) trees grow, (2) features expand. These reflect changes in the granularity of syntactic analyses.
2. There are two ways to define granularity: (1) level of detailedness (narrow), (2) inventory of syntactic categories (broad).
3. There are different ways to understand granularity (i.e., granularity is a multifaceted notion): (1) broad vs. narrow, (2) descriptive vs. explanatory, (3) syntagmatic vs. paradigmatic, (4) mental vs. analytical.
4. Granularity lifting seems to be developing faster in syntax than in other areas of linguistics (as evidenced by the granularity mismatch at the syntax-semantic interface).

Summary

5. In generative syntax granularity is omnipresent, flexible, and usually left implicit. Its role is analogous to that of a typing context in type theory.

Summary

5. In generative syntax granularity is omnipresent, flexible, and usually left implicit. Its role is analogous to that of a typing context in type theory.
6. Granularity matters for both methodological and conceptual reasons.

Summary

5. In generative syntax granularity is omnipresent, flexible, and usually left implicit. Its role is analogous to that of a typing context in type theory.
6. Granularity matters for both methodological and conceptual reasons.
7. The totality of possible granularity levels for human language syntax together form a hypothetical granularity level space (i.e., a highly abstracted set of categorial inventories), which has a rich and formalizable organization.

Summary

5. In generative syntax granularity is omnipresent, flexible, and usually left implicit. Its role is analogous to that of a typing context in type theory.
6. Granularity matters for both methodological and conceptual reasons.
7. The totality of possible granularity levels for human language syntax together form a hypothetical granularity level space (i.e., a highly abstracted set of categorial inventories), which has a rich and formalizable organization.
8. It is high time that linguists developed an awareness of granularity!


Thank you!

Selected references

Nederpelt, R. & H. Geuvers
Type Theory and Formal Proof: An Introduction.
CUP, 2014.

Biberauer, T. & I. Roberts
Rethinking formal hierarchies: A proposed unification.
COPiL7, 2015.

Kelly, J.
The syntax-semantics interface in distributed morphology.
Georgetown University PhD dissertation, 2013.

Song, C.
On the formal flexibility of syntactic categories.
University of Cambridge PhD dissertation, 2019.