More than abstract nonsense: A Category-theoretic sketch of the syntactic category system

Chenchen Julio Song, cs791@cam.ac.uk

Theoretical and Applied Linguistics
University of Cambridge

SyntaxLab, 29 January 2019

Overview

Introduction

Methodology

Category Theory
- Category
- Functor and Natural Transformation
- Adjunction

Syntactic Category System
- Cross-functional-hierarchy parallelism
- Global interconnection
- Entire SCS

Conclusion

The Syntactic Category System (SCS)

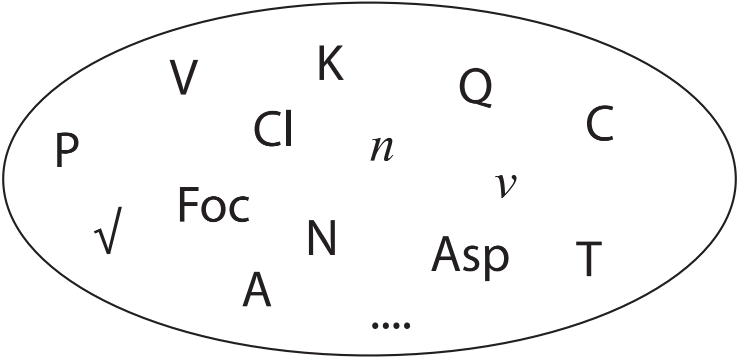


Figure 1: The universe of syntactic categories.

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.
- Parallel hierarchies:
  V–v–T–C
  N–n–Num–D. . .

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.
- Parallel hierarchies:
  V–v–T–C
  N–n–Num–D...
- Stacked hierarchies:
  v–C $>>$
  V–v–T–C $>>$
  V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top $>>$ ...

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.
- Parallel hierarchies:
  V–v–T–C
  N–n–Num–D...
- Stacked hierarchies:
  v–C $>>$
  V–v–T–C $>>$
  V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top $>> \ldots$
- Flexible hierarchies: N–n–Num–D $vs.$ N–n–Cl–D
  V–v–T–C $vs.$ V–v–Asp–C

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.
- Parallel hierarchies:
  V–v–T–C
  N–n–Num–D...
- Stacked hierarchies:
  v–C $>>$
  V–v–T–C $>>$
  V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top $>> \dots$
- Flexible hierarchies: N–n–Num–D $vs.$ N–n–Cl–D
  V–v–T–C $vs.$ V–v–Asp–C

☞ SCS has an intuitively rich **ontological structure**.

The SCS is not just a set

- Functional hierarchies
  aka. extended projections, hierarchies of projections, etc.
- Parallel hierarchies: #parallelism
  V–v–T–C
  N–n–Num–D...
- Stacked hierarchies: #granularity
  v–C $>>$
  V–v–T–C $>>$
  V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top $>> \ldots$
- Flexible hierarchies: N–n–Num–D *vs.* N–n–Cl–D
  V–v–T–C *vs.* V–v–Asp–C

☞ SCS has an intuitively rich **ontological structure**.

Ontological structure of the SCS

- Independent of concrete derivations (hence part of lexicon)

Ontological structure of the SCS

- Independent of concrete derivations (hence part of lexicon)
- More subtle than first impression

Ontological structure of the SCS

- Independent of concrete derivations (hence part of lexicon)
- More subtle than first impression
  - What category is parallel with what? #parallelism
    V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top
    N–Gen–n–Cl–Num–Q–Det–K

Ontological structure of the SCS

- Independent of concrete derivations (hence part of lexicon)
- More subtle than first impression
  - What category is parallel with what? #parallelism
    V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top
    N–Gen–n–Cl–Num–Q–Det–K
  - Which hierarchy is stacked on which? #granularity
    Res–Proc–Init–v–T–C $>>$ V–v–Asp–Tns–Mod–C *vs.* V–v–Asp–Tns–Mod–C $>>$ Res–Proc–Init–v–T–C

Ontological structure of the SCS

- Independent of concrete derivations (hence part of lexicon)
- More subtle than first impression
  - What category is parallel with what? #parallelism
    V–Appl–Voice–Asp–Tns–Mod–Fin–Foc–Top
    N–Gen–n–Cl–Num–Q–Det–K
  - Which hierarchy is stacked on which? #granularity
    Res–Proc–Init–v–T–C $>>$ V–v–Asp–Tns–Mod–C *vs.* V–v–Asp–Tns–Mod–C $>>$ Res–Proc–Init–v–T–C

☞ **Cross-hierarchy relations** are rather intricate.

More abstract aspects of the SCS ontological structure

- Parallelism and granularity stacking are **complementary**

More abstract aspects of the SCS ontological structure

- Parallelism and granularity stacking are **complementary**
- The ontological structure is **layered**:
  individual category $\rightarrow$
  individual hierarchy $\rightarrow$
  individual granularity level $\rightarrow$
  multiple granularity levels $\rightarrow$
  entire SCS

More abstract aspects of the SCS ontological structure

- Parallelism and granularity stacking are **complementary**
- The ontological structure is **layered**:
  individual category $\rightarrow$
  individual hierarchy $\rightarrow$
  individual granularity level $\rightarrow$
  multiple granularity levels $\rightarrow$
  entire SCS #“ladder of abstraction”

More abstract aspects of the SCS ontological structure

- Parallelism and granularity stacking are **complementary**
- The ontological structure is **layered**:
  individual category $\rightarrow$
  individual hierarchy $\rightarrow$
  individual granularity level $\rightarrow$
  multiple granularity levels $\rightarrow$
  entire SCS #"ladder of abstraction"
- A crucial concept: **order relation**

Why is the SCS ontological structure worth studying?

- It underlies **derivation**: first-Merge position
  T–C on f-hierarchy $\Rightarrow$ T–C in concrete derivation

Why is the SCS ontological structure worth studying?

- It underlies **derivation**: first-Merge position
  T–C on f-hierarchy ⇒ T–C in concrete derivation
- It reflects **acquisition**: (Biberauer & Roberts 2015)

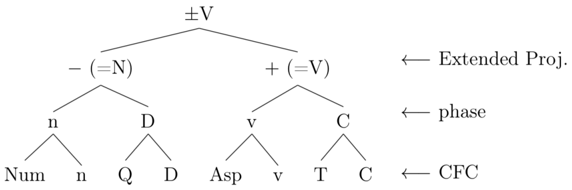


Figure 2: Successive category division results in stacked hierarchies.

Why is the SCS ontological structure worth studying?

- It underlies **derivation**: first-Merge position
  T–C on f-hierarchy ⇒ T–C in concrete derivation
- It reflects **acquisition**: (Biberauer & Roberts 2015)

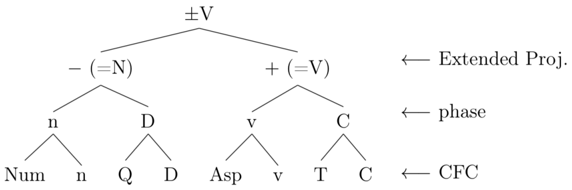


Figure 2: Successive category division results in stacked hierarchies.

- It links **cognitive domains**: "universal spine"
  Classification–PoV–Anchoring–Linking (Wiltschko 2014)

In this talk, I will

- Explore the SCS ontological structure
- Formalize hierarchies and cross-hierarchy relations

Mathematics is the professional tool to study structures

- It is rigorous and can make intuitions explicit

Mathematics is the professional tool to study structures

- It is rigorous and can make intuitions explicit
- It suits our task: functional hierarchies are ordered sets

Mathematics is the professional tool to study structures

- It is rigorous and can make intuitions explicit
- It suits our task: functional hierarchies are ordered sets
- A branch of math is dedicated to studying abstract structures:

> *Category theory is [...] unmatched in its ability to organize and layer abstractions, to find commonalities between structures of all sorts, and [...] it has also been branching out into science, informatics, and industry. We believe that it has the potential to be a major cohesive force in the world, building rigorous bridges between disparate worlds, both theoretical and practical.*
>
> *(Fong & Spivak 2018)*

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

☞ NB the layered levels of abstraction.

Category Theory Roadmap

"Big 3": Category, Functor, and Natural Transformation.

- A Category is like a universe of discourse.
- A Functor connects two such universes.
- A Natural Transformation connects two such Functors.

☞NB the layered levels of abstraction.

Level 4: Adjunction (Category comparison).

Category Theory “Big 3”

A Category $\mathcal{C}$ has **objects** and **arrows** (aka. **morphisms**), where

- Each object C has an identity arrow id_C
- Arrows compose
- Composition obeys two *coherence conditions*
  - **Associativity**: $h \circ (g \circ f) = (h \circ g) \circ f$
  - **Unit law**: $\mathrm{id}_B \circ f = f$, $g \circ \mathrm{id}_B = g$ #commutative diagram

$$
\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, shift right, "(h\circ g)\circ f"'] \arrow[d, shift left, "h\circ(g\circ f)"] \arrow[dr, "g\circ f"'] & B \arrow[d, "g"] \arrow[dl, "h\circ g"'] \\
D & C \arrow[l, "h"]
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[dr, "f"'] & B \arrow[d, "\mathrm{id}_B"] \arrow[dr, "g"] & \\
& B \arrow[r, "g"] & C
\end{tikzcd}
$$

Category Theory “Big 3”

A Category $\mathcal{C}$ has **objects** and **arrows** (aka. **morphisms**), where

- Each object C has an identity arrow id_C
- Arrows compose
- Composition obeys two *coherence conditions*
  - **Associativity**: $h \circ (g \circ f) = (h \circ g) \circ f$
  - **Unit law**: $id_B \circ f = f$, $g \circ id_B = g$ #commutative diagram

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, shift right, "(h\circ g)\circ f"'] \arrow[d, shift left, "h\circ(g\circ f)"] \arrow[dr, "g\circ f"] & B \arrow[d, "g"] \arrow[dl, "h\circ g"'] \\ D & C \arrow[l, "h"] \end{tikzcd} \qquad \begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "f"'] & B \arrow[d, "id_B"] \arrow[dr, "g"] & \\ & B \arrow[r, "g"'] & C \end{tikzcd}$$

☞ **Anything** that satisfies this definition is a Category.

Examples

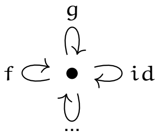
A one-object-one-arrow Category $\mathbf{1}$

$$\bullet \circlearrowleft \mathrm{id}$$

Examples

A one-object-one-arrow Category $\mathbf{1}$

A one-object-many-arrow Category $\mathbf{M}$



Examples

A one-object-one-arrow Category $\mathbf{1}$

$$\begin{tikzcd} \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

A one-object-many-arrow Category $\mathbf{M}$

$$\begin{tikzcd} \bullet \arrow[loop right, "\mathrm{id}"] \arrow[loop left, "f"] \arrow[loop above, "g"] \arrow[loop below, "\cdots"] \end{tikzcd}$$

A two-object-three-arrow Category $\mathbf{2}$

$$\begin{tikzcd} \bullet \arrow[loop left, "\mathrm{id}"] \arrow[r] & \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

Examples

A one-object-one-arrow Category $\mathbf{1}$

$$\begin{tikzcd} \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

A one-object-many-arrow Category $\mathbf{M}$ #monoid

$$\begin{tikzcd} \bullet \arrow[loop above, "g"] \arrow[loop left, "f"] \arrow[loop right, "\mathrm{id}"] \arrow[loop below, "\cdots"] \end{tikzcd}$$

A two-object-three-arrow Category $\mathbf{2}$ #poset

$$\begin{tikzcd} \bullet \arrow[loop left, "\mathrm{id}"] \arrow[r] & \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

Examples

A one-object-one-arrow Category $\mathbf{1}$

$$\begin{tikzcd} \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

Partially ordered set (poset)

A set equipped with a partial order relation $\sqsubseteq$

- reflexive, transitive, antisymmetric

$\cdots$

A two-object-three-arrow Category $\mathbf{2}$ #poset

$$\begin{tikzcd} \bullet \arrow[loop left, "\mathrm{id}"] \arrow[r] & \bullet \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

Examples

The Category $\mathbf{Set}$ of all (small) sets and functions

Examples

The Category **Set** of all (small) sets and functions

- Every set is also a Category

$$
\begin{array}{llll}
\mathrm{id} \circlearrowleft \bullet & \mathrm{id} \circlearrowleft \bullet & \mathrm{id} \circlearrowleft \bullet & \\
\mathrm{id} \circlearrowleft \bullet & \mathrm{id} \circlearrowleft \bullet & \mathrm{id} \circlearrowleft \bullet & \dots
\end{array}
$$

Examples

The Category $\mathbf{Set}$ of all (small) sets and functions

- Every set is also a Category #discrete Category

$$\begin{array}{llll} \mathrm{id} \circlearrowright \bullet & \mathrm{id} \circlearrowright \bullet & \mathrm{id} \circlearrowright \bullet & \\ \mathrm{id} \circlearrowright \bullet & \mathrm{id} \circlearrowright \bullet & \mathrm{id} \circlearrowright \bullet & \ldots \end{array}$$

Examples

The Category $\mathbf{Set}$ of all (small) sets and functions

- Every set is also a Category #discrete Category

$$\boxed{\begin{tikzcd} \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \\ \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \dots \end{tikzcd}}$$

The Category $\mathbf{Pos}$ of all posets and monotone functions

$$\begin{tikzcd} \bullet \arrow[loop above, "\mathtt{id}"] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] & \dots \\ \mathbf{1} & \mathbf{2} & & & \mathbf{3} & & \dots \end{tikzcd}$$

Examples

The Category **Set** of all (small) sets and functions

- Every set is also a Category #discrete Category

$$\begin{tikzcd}
\bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \\
\bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \bullet \arrow[loop left, "\mathtt{id}"] & \ldots
\end{tikzcd}$$

The Category **Pos** of all posets and monotone functions

$$\begin{tikzcd}
\bullet \arrow[loop above, "\mathtt{id}"] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] \arrow[r] & \bullet \arrow[loop above, "\mathtt{id}"] & \ldots \\
\mathbf{1} & \mathbf{2} & & & \mathbf{3} & & \ldots
\end{tikzcd}$$

- $\boxed{\text{Every poset is also a Category}}$ #poset Category

Examples

The Category $\mathbf{Set}$ of all (small) sets and functions

- Every set is also a Category #discrete Category

$$\mathrm{id} \circlearrowright \bullet \quad \mathrm{id} \circlearrowright \bullet \quad \mathrm{id} \circlearrowright \bullet$$

$$\mathrm{id} \circlearrowright \bullet \quad \mathrm{id} \circlearrowright \bullet \quad \mathrm{id} \circlearrowright \bullet \qquad \ldots$$

The Category $\mathbf{Pos}$ of all posets and monotone functions

Monotone function

A function $f\colon A \to B$ that is order-preserving

- $\forall x, y \in A, x \sqsubseteq y \Rightarrow f(x) \sqsubseteq f(y)$

- $\boxed{\text{Every poset is also a Category}}$ #poset Category

A functional hierarchy is a poset Category

- Objects: individual functional categories
- Arrows: instances of partial order relation
- Composition: by transitivity
- Identities: by reflexivity

$$\begin{tikzcd} \mathrm{V} \arrow[r] \arrow[loop above, "\mathtt{id}"] & v \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{T} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{C} \arrow[loop above, "\mathtt{id}"] \end{tikzcd} \qquad \begin{tikzcd} \mathrm{N} \arrow[r] \arrow[loop above, "\mathtt{id}"] & n \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Num} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{D} \arrow[loop above, "\mathtt{id}"] \end{tikzcd}$$

$$\begin{tikzcd} \mathrm{V} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Appl} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Voice} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Asp} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Tns} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Foc} \arrow[r] \arrow[loop above, "\mathtt{id}"] & \mathrm{Top} \arrow[loop above, "\mathtt{id}"] \end{tikzcd}$$

Several f-hierarchies form a Category of posets

NB this in itself is not a poset Category (but a *preorder Category*).

- Objects: individual f-hierarchies $A_{\mathbf{N}}$, $A_{\mathbf{V}}$, etc.
- Arrows: monotone functions (tbc)
- Composition: by monotone function composition
- Identities: identity monotone functions

$$\begin{tikzcd}
A_{\mathbf{N}} \arrow[loop left, "\mathrm{id}"] \arrow[r, shift left] \arrow[d, shift left] \arrow[dr, shift left] & A_{\mathbf{V}} \arrow[loop right, "\mathrm{id}"] \arrow[l, shift left] \arrow[d, shift left] \arrow[dl, shift left] & \\
A_{\mathbf{P}} \arrow[loop left, "\mathrm{id}"] \arrow[u, shift left] \arrow[r, shift left] \arrow[ur, shift left] & A_{\mathbf{A}} \arrow[loop right, "\mathrm{id}"] \arrow[u, shift left] \arrow[l, shift left] \arrow[ul, shift left] & \cdots
\end{tikzcd}$$

Several f-hierarchies form a Category of posets

NB this in itself is not a poset Category (but a *preorder Category*).

- Objects: individual f-hierarchies $A_{\mathbf{N}}$, $A_{\mathbf{V}}$, etc.
- Arrows: monotone functions (tbc)
- Composition: by monotone function composition
- Identities: identity monotone functions

$$\begin{tikzcd}
\mathrm{id} \circlearrowright A_{\mathbf{N}} \arrow[r, shift left] \arrow[d, shift right] \arrow[dr, shift left] & A_{\mathbf{V}} \circlearrowleft \mathrm{id} \arrow[l, shift left] \arrow[d, shift right] \arrow[dl, shift left] & \\
\mathrm{id} \circlearrowright A_{\mathbf{P}} \arrow[r, shift left] \arrow[u, shift right] \arrow[ur, shift left] & A_{\mathbf{A}} \circlearrowleft \mathrm{id} \arrow[l, shift left] \arrow[u, shift right] \arrow[ul, shift left] & \ldots
\end{tikzcd}$$

☞ This describes a speaker's **functional category inventory**.

Several f-hierarchies form a Category of posets

NB this in itself is not a poset Category (but a *preorder Category*).

- Objects: individual f-hierarchies $A_{\mathbf{N}}$, $A_{\mathbf{V}}$, etc.
- Arrows: monotone functions (tbc)
- Composition: by monotone function composition
- Identities: identity monotone functions

$$\begin{tikzcd}
A_{\mathbf{N}} \arrow[loop left, "\mathtt{id}"] \arrow[r, shift left] \arrow[d, shift left] \arrow[dr, shift left] & A_{\mathbf{V}} \arrow[loop right, "\mathtt{id}"] \arrow[l, shift left] \arrow[d, shift left] \arrow[dl, shift left] \\
A_{\mathbf{P}} \arrow[loop left, "\mathtt{id}"] \arrow[r, shift left] \arrow[u, shift left] \arrow[ur, shift left] & A_{\mathbf{A}} \arrow[loop right, "\mathtt{id}"] \arrow[l, shift left] \arrow[u, shift left] \arrow[ul, shift left] & \ldots
\end{tikzcd}$$

☞This describes a speaker's **functional category inventory**.

Several f-hierarchies form a Category of posets

NB this in itself is not a poset Category (but a *preorder Category*).

Preordered set (preorder)

A set equipped with a preorder relation $\preceq$

- reflexive and transtive
- partial order without antisymmetry

$$
\begin{tikzcd}
A_{\mathbf{N}} \arrow[loop left, "\mathrm{id}"] \arrow[r, shift left] \arrow[d, shift left] \arrow[dr, shift left] & A_{\mathbf{V}} \arrow[loop right, "\mathrm{id}"] \arrow[l, shift left] \arrow[d, shift left] \arrow[dl, shift left] & \\
A_{\mathbf{P}} \arrow[loop left, "\mathrm{id}"] \arrow[r, shift left] \arrow[u, shift left] \arrow[ur, shift left] & A_{\mathbf{A}} \arrow[loop right, "\mathrm{id}"] \arrow[l, shift left] \arrow[u, shift left] \arrow[ul, shift left] & \ldots
\end{tikzcd}
$$

☞ This describes a speaker's **functional category inventory**.

Several f-hierarchies form a Category of posets

NB this in itself is not a poset Category (but a *preorder Category*).

Preordered set (preorder)

A set equipped with a preorder relation $\preceq$

- reflexive and transtive
- partial order without antisymmetry

$$\begin{tikzcd}
A_{\mathbf{N}} \arrow[loop left, "\mathrm{id}"] \arrow[r, shift left] \arrow[d, shift right] \arrow[dr, shift left] & A_{\mathbf{V}} \arrow[loop right, "\mathrm{id}"] \arrow[l, shift left] \arrow[d, shift right] \arrow[dl, shift left] & \\
A_{\mathbf{P}} \arrow[loop left, "\mathrm{id}"] \arrow[u, shift right] \arrow[r, shift left] \arrow[ur, shift left] & A_{\mathbf{A}} \arrow[loop right, "\mathrm{id}"] \arrow[u, shift right] \arrow[l, shift left] \arrow[ul, shift left] & \ldots
\end{tikzcd}$$

☜ merely expository

☞ This describes a speaker's **functional category inventory**.

Category Theory “Big 3”

A Functor $F\colon \mathcal{C} \to \mathcal{D}$ is an arrow between two Categories.

- It maps **objects to objects** and **arrows to arrows**
- It preserves composition and identities

$$\mathcal{C} \xrightarrow{F} \mathcal{D}$$

$$
\begin{tikzcd}
A \arrow[loop left, "id_A"] \arrow[r, "f"] \arrow[dr, "h"'] & B \arrow[loop right, "id_B"] \arrow[d, "g"] \\
& C \arrow[loop right, "id_C"]
\end{tikzcd}
\qquad
\begin{tikzcd}
F(A) \arrow[loop left, "id_{F(A)}"] \arrow[r, "F(f)"] \arrow[dr, "F(h)"'] & F(B) \arrow[loop right, "id_{F(B)}"] \arrow[d, "F(g)"] \\
& F(C) \arrow[loop right, "id_{F(C)}"]
\end{tikzcd}
$$

☞ A Functor produces an **image** of one Category in another.

Examples

- The Functor $\mathbb{U}\colon \mathbf{Pos} \to \mathbf{Set}$ sends posets to their underlying sets by forgetting the partial orders. #forgetful Functor

Examples

- The Functor $\mathrm{U}\colon \mathbf{Pos} \to \mathbf{Set}$ sends posets to their underlying sets by forgetting the partial orders. #forgetful Functor
- The Functor $\mathrm{F}\colon \mathbf{Set} \to \mathbf{Pos}$ sends sets to the smallest posets built on them (ordered by $=$). #free Functor

Examples

- The Functor $\mathrm{U}\colon \mathbf{Pos} \to \mathbf{Set}$ sends posets to their underlying sets by forgetting the partial orders. #forgetful Functor
- The Functor $\mathrm{F}\colon \mathbf{Set} \to \mathbf{Pos}$ sends sets to the smallest posets built on them (ordered by $=$). #free Functor
- The Functor $\mathrm{M}[\![\cdot]\!]\colon \mathbf{Syn} \to \mathbf{Sem}$ sends syntactic expressions (atomic or phrasal) to their meanings. #interpretation Functor

Monotone functions qua Functors (identities omitted)

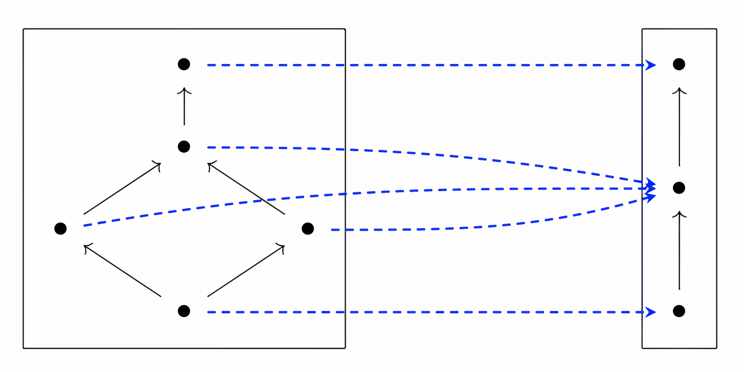


Figure 3: A random monotone function (Fong & Spivak 2018).

Monotone functions qua Functors

The Functor R: $\mathbf{Ani} \to \mathbf{Tax}$ sends animals to taxonomic ranks.

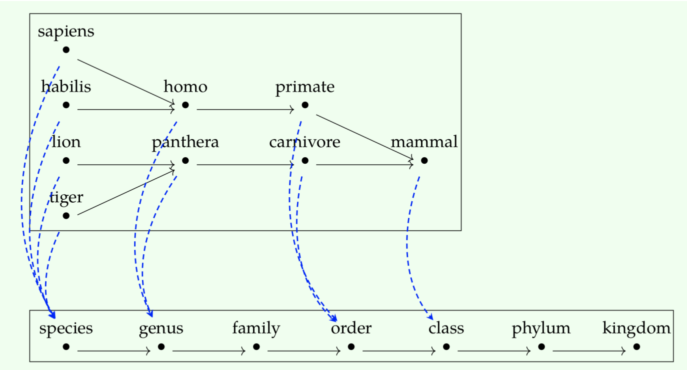


Figure 4: A Functor for biological taxonomy (Fong & Spivak 2018).

Functors between f-hierarchy Categories

The Functor $\mathrm{F}\colon \mathcal{A}_{\mathbf{V}} \to \mathcal{A}_{\mathbf{N}}$ maps verbal categories to nominal ones

$$
\begin{tikzcd}
\mathcal{A}_{\mathbf{V}} \arrow[d, "\mathrm{F}"'] & \mathsf{V} \arrow[r] \arrow[d, mapsto, dashed] & v \arrow[r] \arrow[d, mapsto, dashed] & \mathsf{T} \arrow[r] \arrow[d, mapsto, dashed] & \mathsf{C} \arrow[d, mapsto, dashed] \\
\mathcal{A}_{\mathbf{N}} & \mathsf{N} \arrow[r] & n \arrow[r] & \mathsf{Num} \arrow[r] & \mathsf{D}
\end{tikzcd}
$$

⚠ Don't read too much into the specific mapping (yet).

Functors between f-hierarchy Categories

Another Functor between $\mathcal{A}_{\mathbf{V}}$ and $\mathcal{A}_{\mathbf{N}}$

$$\begin{tikzcd}
\mathcal{A}_{\mathbf{V}} \arrow[d, "G"'] & \mathsf{V} \arrow[r] \arrow[d, mapsto, dashed] & v \arrow[r] \arrow[dl, mapsto, dashed] & \mathsf{T} \arrow[r] \arrow[dll, mapsto, dashed] & \mathsf{C} \arrow[dlll, mapsto, dashed] \\
\mathcal{A}_{\mathbf{N}} & \mathsf{N} \arrow[r] & n \arrow[r] & \mathsf{Num} \arrow[r] & \mathsf{D}
\end{tikzcd}$$

Functors between f-hierarchy Categories

Another Functor between $\mathcal{A}_{\mathbf{V}}$ and $\mathcal{A}_{\mathbf{N}}$

$$
\begin{tikzcd}
\mathcal{A}_{\mathbf{V}} \arrow[d, "G"'] & \mathsf{V} \arrow[r] \arrow[d, dashed, mapsto] & v \arrow[r] \arrow[dl, dashed] & \mathsf{T} \arrow[r] \arrow[dll, dashed] & \mathsf{C} \arrow[dlll, dashed] \\
\mathcal{A}_{\mathbf{N}} & \mathsf{N} \arrow[r] & n \arrow[r] & \mathsf{Num} \arrow[r] & \mathsf{D}
\end{tikzcd}
$$

#constant Functor

Functors between f-hierarchy Categories

Another Functor between $\mathcal{A}_{\mathbf{V}}$ and $\mathcal{A}_{\mathbf{N}}$

$$
\begin{tikzcd}
\mathcal{A}_{\mathbf{V}} \arrow[d, "G"'] & \mathsf{V} \arrow[r] \arrow[d, mapsto, dashed] & v \arrow[r] \arrow[dl, mapsto, dashed] & \mathsf{T} \arrow[r] \arrow[dll, mapsto, dashed] & \mathsf{C} \arrow[dlll, mapsto, dashed] \\
\mathcal{A}_{\mathbf{N}} & \mathsf{N} \arrow[r] & n \arrow[r] & \mathsf{Num} \arrow[r] & \mathsf{D}
\end{tikzcd}
$$

#constant Functor

What is a **meaningful** Functor between f-hierarchies? (tbc)

Functors between f-hierarchy Categories

And this one?

$$\begin{array}{ll} \mathcal{B}_{\mathbf{V}} & V \rightarrow \text{Appl} \rightarrow v \rightarrow \text{Asp} \rightarrow \text{Tns} \rightarrow \text{Mod} \rightarrow \text{Fin} \rightarrow \text{Foc} \rightarrow \text{Top} \\ \downarrow ? & \\ \mathcal{B}_{\mathbf{N}} & N \rightarrow \text{Gen} \rightarrow n \rightarrow \text{Cl} \rightarrow \text{Num} \longrightarrow \text{Q} \longrightarrow \text{D} \end{array}$$

Functors between f-hierarchy Categories

And this one?

$$
\begin{array}{ll}
\mathcal{B}_{\mathbf{V}} & \mathrm{V} \to \mathrm{Appl} \to v \to \mathrm{Asp} \to \mathrm{Tns} \to \mathrm{Mod} \to \mathrm{Fin} \to \mathrm{Foc} \to \mathrm{Top} \\
\downarrow {\scriptstyle ?} & \\
\mathcal{B}_{\mathbf{N}} & \mathrm{N} \to \mathrm{Gen} \to n \to \mathrm{Cl} \to \mathrm{Num} \longrightarrow \mathrm{Q} \longrightarrow \mathrm{D}
\end{array}
$$

What is a **stably meaningful** Functor between f-hierarchies?

Functors between f-hierarchy Categories

And this one?

$$\begin{array}{ll} \mathcal{B}_{\mathbf{V}} & \text{V} \to \text{Appl} \to v \to \text{Asp} \to \text{Tns} \to \text{Mod} \to \text{Fin} \to \text{Foc} \to \text{Top} \\ \big\downarrow ? & \\ \mathcal{B}_{\mathbf{N}} & \text{N} \to \text{Gen} \to n \to \text{Cl} \to \text{Num} \longrightarrow \text{Q} \longrightarrow \text{D} \end{array}$$

What is a **stably meaningful** Functor between f-hierarchies?

☞ Essentially a question about **cross-hierarchy parallelism**.

Category Theory “Big 3”

A Natural Transformation $\alpha\colon F \Rightarrow G$ is an arrow between Functors.

- It is a transformation between two Functorial images of $\mathcal{C}$
- It is a **family of maps** α_A in $\mathcal{D}$
- For any $f\colon A \to A'$ in $\mathcal{C}$ there is a **naturality square** in $\mathcal{D}$

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, bend right, "G"'] \arrow[r, phantom, "\Downarrow\alpha"] & \mathcal{D} \end{tikzcd}$$

$$A \xrightarrow{f} A'$$

$$\begin{tikzcd} F(A) \arrow[r, "F(f)"] \arrow[d, "\alpha_A"'] & F(A') \arrow[d, "\alpha_{A'}"] \\ G(A) \arrow[r, "G(f)"] & G(A') \end{tikzcd}$$

Adjunction (aka. Adjointness or Adjoint Situation)

In the configuration $\mathcal{A} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{B}$, we say that F is **left adjoint** to G and G is **right adjoint** to F, and write $F \dashv G$, if

$$\mathcal{B}(F(A), B) \cong \mathcal{A}(A, G(B)) \qquad (\theta)$$

naturally in $A \in \mathcal{A}$ and $B \in \mathcal{B}$.

Adjunction (aka. Adjointness or Adjoint Situation)

In the configuration $\mathcal{A} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{B}$, we say that F is **left adjoint** to G and G is **right adjoint** to F, and write $F \dashv G$, if

$$\mathcal{B}(F(A), B) \cong \mathcal{A}(A, G(B)) \qquad (\theta)$$

naturally in $A \in \mathcal{A}$ and $B \in \mathcal{B}$.

$$\boxed{\text{there is a } \mathcal{B}\text{-arrow } F(A) \to B \textit{ iff } \text{there is an } \mathcal{A}\text{-arrow } A \to G(B)}$$

Adjunction (aka. Adjointness or Adjoint Situation)

In the configuration $\mathcal{A} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{B}$, we say that F is **left adjoint** to G and G is **right adjoint** to F, and write $F \dashv G$, if

$$\mathcal{B}(F(A), B) \cong \mathcal{A}(A, G(B)) \qquad (\theta)$$

naturally in $A \in \mathcal{A}$ and $B \in \mathcal{B}$.

$$\boxed{\text{there is a } \mathcal{B}\text{-arrow } F(A) \to B \textit{ iff } \text{there is an } \mathcal{A}\text{-arrow } A \to G(B)}$$

☞ This describes a weak similarity between Categories.

Adjunction (aka. Adjointness or Adjoint Situation)

Adjunction via hom-set isomorphism

$$\begin{tikzcd} \mathcal{A} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{B} \arrow[ll, bend left, "G"] \end{tikzcd}$$

$$\begin{tikzcd} A \arrow[rr] \arrow[d, "f"'] \arrow[drr, phantom, "\theta \wr\|"] & & F(A) \arrow[d, "g"] \\ G(B) & & B \arrow[ll] \end{tikzcd}$$

$$\downarrow\theta \; \frac{F(A) \xrightarrow{g} B}{A \xrightarrow{f} G(B)} \; \uparrow\theta^{-1}$$

- Hom-set $\mathcal{C}(X, Y) \triangleq$ **set** of all $\mathcal{C}$-arrows $X \to Y$

$\Rightarrow$ Isomorphism $\equiv$ **bijection**

Adjunction (aka. Adjointness or Adjoint Situation)

Since the isomorphism (θ) is natural in A and B, we can

- let $A = G(B)$ and get

$$\frac{F(G(B)) \to B}{G(B) \xrightarrow{f = id_{G(B)}} G(B)}$$

$$\downarrow\theta \; \frac{F(A) \xrightarrow{g} B}{A \xrightarrow{f} G(B)} \; \uparrow\theta^{-1}$$

- let $B = F(A)$ and get

$$\frac{F(A) \xrightarrow{g = id_{F(A)}} F(A)}{A \to G(F(A))}$$

i.e. we use (θ) to map **identity arrows**.

Adjunction (aka. Adjointness or Adjoint Situation)

Since the isomorphism (θ) is natural in A and B, we can

Adjunction via unit and co-unit

$$\downarrow \frac{F(A) \xrightarrow{g=id_{F(A)}} F(A)}{A \xrightarrow{\eta_A} G(F(A))}$$

$$\begin{tikzcd} \mathcal{A} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{B} \arrow[ll, bend left, "G"] \end{tikzcd}$$

$$\frac{F(G(B)) \xrightarrow{\varepsilon_B} B}{G(B) \xrightarrow{f=id_{G(B)}} G(B)} \uparrow$$

$$\begin{tikzcd} & A \arrow[dl, "\eta_A"'] \arrow[d, "f"] & \\ G(F(A)) \arrow[r, "G(g)"'] & G(B) \arrow[loop right, "id_{G(B)}"] & \end{tikzcd}$$

$$\begin{tikzcd} F(A) \arrow[loop left, "id_{F(A)}"] \arrow[r, "F(f)"] \arrow[d, "g"] & F(G(B)) \arrow[dl, "\varepsilon_B"] \\ B & \end{tikzcd}$$

i.e. we use (θ) to map **identity arrows**.

Adjunction (aka. Adjointness or Adjoint Situation)

The two special arrows

$$\eta_A : A \to G(F(A))$$
$$\varepsilon_B : F(G(B)) \to B$$

extend to two Natural Transformations (between two *endofunctors*)

$$\eta : \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F$$
$$\varepsilon : F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}}$$

Adjunction (aka. Adjointness or Adjoint Situation)

The two special arrows

$$\eta_A : A \to G(F(A))$$
$$\varepsilon_B : F(G(B)) \to B$$

extend to two Natural Transformations (between two *endofunctors*)

$$\begin{tikzcd} \mathcal{A} \arrow[r, bend left, "\mathrm{Id}_{\mathcal{A}}"] \arrow[r, bend right, "G \circ F"'] \arrow[r, phantom, "\Downarrow\eta"] & \mathcal{A} \end{tikzcd} \qquad \eta : \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F$$

$$\varepsilon : F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}} \qquad \begin{tikzcd} \mathcal{B} \arrow[r, bend left, "F \circ G"] \arrow[r, bend right, "\mathrm{Id}_{\mathcal{B}}"'] \arrow[r, phantom, "\Downarrow\varepsilon"] & \mathcal{B} \end{tikzcd}$$

Adjunction (aka. Adjointness or Adjoint Situation)

The two special arrows

$$\eta_A : A \to G(F(A))$$
$$\varepsilon_B : F(G(B)) \to B$$

extend to two Natural Transformations (between two *endofunctors*)

$$\begin{tikzcd} \mathcal{A} \arrow[r, bend left, "\mathrm{Id}_{\mathcal{A}}"] \arrow[r, bend right, "G \circ F"'] \arrow[r, phantom, "\Downarrow\eta"] & \mathcal{A} \end{tikzcd} \quad \eta : \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F$$

$$\varepsilon : F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}} \quad \begin{tikzcd} \mathcal{B} \arrow[r, bend left, "F \circ G"] \arrow[r, bend right, "\mathrm{Id}_{\mathcal{B}}"'] \arrow[r, phantom, "\Downarrow\varepsilon"] & \mathcal{B} \end{tikzcd}$$

☞ These are the **unit** and **co-unit** of an Adjunction.

Category comparison via unit and co-unit

Three levels of similarity between Categories

Category comparison via unit and co-unit

Three levels of similarity between Categories

- Two natural transformations: weak (Adjunction)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F, \quad \varepsilon\colon F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}}$$

Category comparison via unit and co-unit

Three levels of similarity between Categories

- Two natural transformations: weak (Adjunction)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F,\ \ \varepsilon\colon F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}}$$

- Two natural isomorphisms: intermediate (equivalence)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} \Leftrightarrow G \circ F,\ \ \varepsilon\colon F \circ G \Leftrightarrow \mathrm{Id}_{\mathcal{B}}$$

Category comparison via unit and co-unit

Three levels of similarity between Categories

- Two natural transformations: weak (Adjunction)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} \Rightarrow G \circ F, \;\; \varepsilon\colon F \circ G \Rightarrow \mathrm{Id}_{\mathcal{B}}$$

- Two natural isomorphisms: intermediate (equivalence)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} \Leftrightarrow G \circ F, \;\; \varepsilon\colon F \circ G \Leftrightarrow \mathrm{Id}_{\mathcal{B}}$$

- Two equalities: strong (isomorphism)

$$\eta\colon \mathrm{Id}_{\mathcal{A}} = G \circ F, \;\; \varepsilon\colon F \circ G = \mathrm{Id}_{\mathcal{B}}$$

Category comparison via unit and co-unit

Three levels of similarity between Categories

Adjunction (weak similarity)

For poset Categories

After a round-trip A grows and B shrinks, in the depicted way.

$$\begin{tikzcd} \mathcal{A} \arrow[r, bend left, "F"] \arrow[r, phantom, "\perp"] & \mathcal{B} \arrow[l, bend left, "G"] \end{tikzcd}$$

$$\begin{tikzcd} & A \arrow[dl, "\eta_A"'] \arrow[d, "f"] \\ G(F(A)) \arrow[r, "G(g)"'] & G(B) \arrow[loop right, "id_{G(B)}"] \end{tikzcd}
\qquad
\begin{tikzcd} F(A) \arrow[loop left, "id_{F(A)}"] \arrow[r, "F(f)"] \arrow[d, "g"] & F(G(B)) \arrow[dl, "\varepsilon_B"] \\ B & \end{tikzcd}$$

Category comparison via unit and co-unit

Three levels of similarity between Categories

Equivalence (intermediate similarity)

For poset Categories

After a round-trip A and B land somewhere isomorphic to themselves.

$$\begin{tikzcd} \mathcal{A} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{B} \arrow[ll, bend left, "G"] \end{tikzcd}$$

$$\begin{tikzcd} & A \arrow[d, "f"] \arrow[dl, leftrightarrow, "\eta_A"'] \\ G(F(A)) \arrow[r, "G(g)"'] & G(B) \arrow[loop right, "id_{G(B)}"] \end{tikzcd}
\qquad
\begin{tikzcd} F(A) \arrow[loop left, "id_{F(A)}"] \arrow[r, "F(f)"] \arrow[d, "g"] & F(G(B)) \arrow[dl, leftrightarrow, "\varepsilon_B"] \\ B & \end{tikzcd}$$

Category comparison via unit and co-unit

Three levels of similarity between Categories

Isomorphism (strong similarity)

For poset Categories

After a round-trip A and B go back to themselves.

$$\begin{tikzcd}
\mathcal{A} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{B} \arrow[ll, bend left, "G"]
\end{tikzcd}$$

$$\begin{tikzcd}
 & A \arrow[d, "f"'] \arrow[dl, equal, "\eta_A"'] \\
G(F(A)) \arrow[r, "G(g)"'] & G(B) \arrow[loop right, "id_{G(B)}"]
\end{tikzcd}
\qquad
\begin{tikzcd}
F(A) \arrow[loop left, "id_{F(A)}"] \arrow[r, "F(f)"] \arrow[d, "g"] & F(G(B)) \arrow[dl, equal, "\varepsilon_B"] \\
B &
\end{tikzcd}$$

Adjunction for poset Categories: Galois connection

Typical scenario: given a monotone function f, its inverse f^{-1}, if existent, will allow us to say the f-paired elements are **parallel**. When there is no such f^{-1} yet we still want some reasonably "parallel" pairing, a Galois connection, if existent, is our friend.

Adjunction for poset Categories: Galois connection

Typical scenario: given a monotone function f, its inverse f^{-1}, if existent, will allow us to say the f-paired elements are **parallel**. When there is no such f^{-1} yet we still want some reasonably "parallel" pairing, a Galois connection, if existent, is our friend.

Example

Given an *inclusion function* $\iota\colon \mathbb{Z} \to \mathbb{R}$ from integers to reals. Its intuitive inverse is another inclusion $\iota^{-1}\colon \mathbb{R} \to \mathbb{Z}$, which does not exist (e.g. $\iota(2.5) \notin \mathbb{Z}$). But we have a potential **second-best** solution: the *ceiling function* $\lceil - \rceil\colon \mathbb{R} \to \mathbb{Z}$ which maps a real number to the smallest integer above (or equal to) it. We can verify that $\lceil - \rceil$ and ι form a Galois connection.

Adjunction for poset Categories: Galois connection

$$\begin{tikzcd} \mathbb{R} \arrow[r, bend left, "C \triangleq \lceil - \rceil"] & \mathbb{Z} \arrow[l, bend left, "\iota"] \end{tikzcd}$$

$$\begin{tikzcd} & x \arrow[dl, "\eta_x"'] \arrow[d, "f"] & C(x) \arrow[loop left, "\mathrm{id}_{C(x)}"] \arrow[r, "C(f)"] \arrow[d, "g"'] & C(\iota(n)) \arrow[dl, equal, "\varepsilon_n"] \\ \iota(C(x)) \arrow[r, "\iota(g)"'] & \iota(n) \arrow[loop right, "\mathrm{id}_{\iota(n)}"] & n & \end{tikzcd}$$

In the left configuration:

- (i) Since ι is monotone, $C(x) \leqslant n \Rightarrow \iota(C(x)) \leqslant \iota(n)$, but $x \leqslant \iota(C(x))$ because $\iota(C(x)) = \lceil x \rceil$, therefore $x \leqslant \iota(n)$ and so $g \Rightarrow f$.

Adjunction for poset Categories: Galois connection

$$\begin{tikzcd} \mathbb{R} \arrow[r, bend left, "C \triangleq \lceil - \rceil"] & \mathbb{Z} \arrow[l, bend left, "\iota"] \end{tikzcd}$$

$$\begin{tikzcd} & x \arrow[dl, blue, "\eta_x"'] \arrow[d, red, "f"] & C(x) \arrow[loop left, "id_{C(x)}"] \arrow[r, "C(f)"] \arrow[d, red, "g"'] & C(\iota(n)) \arrow[dl, blue, equal, "\varepsilon_n"] \\ \iota(C(x)) \arrow[r, "\iota(g)"'] & \iota(n) \arrow[loop right, "id_{\iota(n)}"] & n & \end{tikzcd}$$

In the left configuration:

- (ii) Since C is monotone, $x \leqslant \iota(n) \Rightarrow C(x) \leqslant C(\iota(n))$, but $C(\iota(n)) = \lceil n \rceil = n$, therefore $C(x) \leqslant n$ and so $f \Rightarrow g$.

Adjunction for poset Categories: Galois connection

$$\begin{tikzcd} \mathbb{R} \arrow[r, bend left, "C \triangleq \lceil - \rceil"] & \mathbb{Z} \arrow[l, bend left, "\iota"] \end{tikzcd}$$

$$\begin{tikzcd} & x \arrow[dl, "\eta_x"'] \arrow[d, "f"] & C(x) \arrow[loop left, "id_{C(x)}"] \arrow[r, "C(f)"] \arrow[d, "g"'] & C(\iota(n)) \arrow[dl, equal, "\varepsilon_n"] \\ \iota(C(x)) \arrow[r, "\iota(g)"'] & \iota(n) \arrow[loop right, "id_{\iota(n)}"] & n & \end{tikzcd}$$

In the left configuration:

- Combining (i) and (ii), we obtain $f \Leftrightarrow g$, whence $C \dashv \iota$, i.e. C is **left adjoint** to ι and ι is **right adjoint** to C.

Adjunction for poset Categories: Galois connection

$$\begin{tikzcd}
\mathbb{R} \arrow[r, bend left, "C \triangleq \lceil - \rceil"] & \mathbb{Z} \arrow[l, bend left, "\iota"]
\end{tikzcd}$$

$$\begin{tikzcd}
 & x \arrow[dl, "\eta_x"'] \arrow[d, "f"] & C(x) \arrow[loop left, "\mathrm{id}_{C(x)}"] \arrow[r, "C(f)"] \arrow[d, "g"] & C(\iota(n)) \arrow[dl, equal, "\varepsilon_n"] \\
\iota(C(x)) \arrow[r, "\iota(g)"'] & \iota(n) \arrow[loop right, "\mathrm{id}_{\iota(n)}"] & n &
\end{tikzcd}$$

In the left configuration:

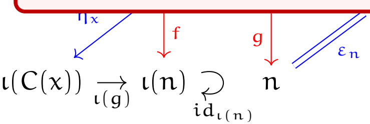
- Combining (i) and (ii), we obtain $f \Leftrightarrow g$, whence $C \dashv \iota$, i.e. C is **left adjoint** to ι and ι is **right adjoint** to C.

☞ Here ε is an equality, so we have an “enhanced” Adjunction.

Adjunction for poset Categories: Galois connection

Enhanced weak similarity (cf. Erné 2004)

In general, if the co-unit of an Adjunction is an *isomorphism* (whereof equality is a special case), typically when an inclusion Functor has a left adjoint, the situation is called an **epi-Adjunction** (or **right perfect Galois connection** for posets).



$\dashv \iota$,
i.e. C is **left adjoint** to ι and ι is **right adjoint** to C.

☞ Here ε is an equality, so we have an “enhanced” Adjunction.

Adjunction for poset Categories: Galois connection

Enhanced weak similarity (cf. Erné 2004)

In general, if the co-unit of an Adjunction is an *isomorphism* (whereof equality is a special case), typically when an inclusion Functor has a left adjoint, the situation is called an **epi-Adjunction** (or **right perfect Galois connection** for posets).

Alternatively, in this situation the "smaller" of the two Categories (here $\mathbb{Z}$) is called a **reflective Subcategory** of the "bigger" one (here $\mathbb{R}$) and the left adjoint (here $\lceil - \rceil$) called a **reflector**.

☞ Here ε is an equality, so we have an "enhanced" Adjunction.

Cross-f-hierarchy parallelism via epi-Adjunction

What is a **stably meaningful** Functor between f-hierarchies?

Cross-f-hierarchy parallelism via epi-Adjunction

What is a **stably meaningful** Functor between f-hierarchies?

- Direct Functors between f-hierarchies are either unstable or not meaningful.

Cross-f-hierarchy parallelism via epi-Adjunction

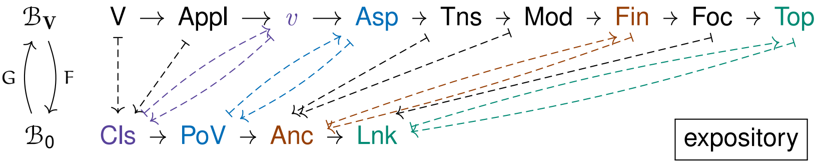
What is a **stably meaningful** Functor between f-hierarchies?

- Direct Functors between f-hierarchies are either unstable or not meaningful.
- But a stably meaningful Functorial connection exists between **each** f-hierarchy and a “universal spine” (Wiltschko 2014), e.g.

Cross-f-hierarchy parallelism via epi-Adjunction

What is a **stably meaningful** Functor between f-hierarchies?

- Direct Functors between f-hierarchies are either unstable or not meaningful.
- But a stably meaningful Functorial connection exists between **each** f-hierarchy and a “universal spine” (Wiltschko 2014), e.g.



Cross-f-hierarchy parallelism via epi-Adjunction

What is a **stably meaningful** Functor between f-hierarchies?

- Direct Functors between f-hierarchies are either unstable or not meaningful.
- But a stably meaningful Functorial connection exists between **each** f-hierarchy and a “universal spine” (Wiltschko 2014), e.g.

$$\begin{array}{ll} \mathcal{B}_{\mathbf{V}} & \mathrm{V} \to \mathrm{Appl} \to v \to \mathrm{Asp} \to \mathrm{Tns} \to \mathrm{Mod} \to \mathrm{Fin} \to \mathrm{Foc} \to \mathrm{Top} \\ \mathrm{G} \dashv \mathrm{F} & \\ \mathcal{B}_0 & \mathrm{Cls} \to \mathrm{PoV} \to \mathrm{Anc} \to \mathrm{Lnk} \end{array}$$

expository

- This is in addition is an epi-Adjunction.

Cross-f-hierarchy parallelism via epi-Adjunction

expository

Let's examine the f-hierarchy epi-Adjunction more closely:

$$\mathcal{B}_{\mathbf{V}} \quad \mathrm{V} \rightarrow \mathrm{Appl} \rightarrow v \rightarrow \mathrm{Asp} \rightarrow \mathrm{Tns} \rightarrow \mathrm{Mod} \rightarrow \mathrm{Fin} \rightarrow \mathrm{Foc} \rightarrow \mathrm{Top}$$

$$G \dashv F$$

$$\mathcal{B}_{0} \quad \mathrm{Cls} \rightarrow \mathrm{PoV} \rightarrow \mathrm{Anc} \rightarrow \mathrm{Lnk}$$

Cross-f-hierarchy parallelism via epi-Adjunction

Let's examine the f-hierarchy epi-Adjunction more closely: $\boxed{\text{expository}}$

$$\mathcal{B}_{\mathbf{V}} \quad \mathrm{V} \to \mathrm{Appl} \to v \to \mathrm{Asp} \to \mathrm{Tns} \to \mathrm{Mod} \to \mathrm{Fin} \to \mathrm{Foc} \to \mathrm{Top}$$

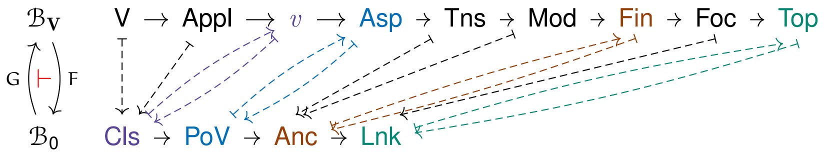
$$\mathrm{G} \quad \vdash \quad \mathrm{F}$$

$$\mathcal{B}_0 \quad \mathrm{Cls} \to \mathrm{PoV} \to \mathrm{Anc} \to \mathrm{Lnk}$$

- The left adjoint $\mathrm{F}\colon \mathcal{B}_{\mathbf{V}} \to \mathcal{B}_0$ is **unique** for any f-hierarchy $\mathcal{B}$.

Cross-f-hierarchy parallelism via epi-Adjunction

Let's examine the f-hierarchy epi-Adjunction more closely: expository



- The left adjoint $\mathrm{F}\colon \mathcal{B}_\mathbf{V} \to \mathcal{B}_0$ is **unique** for any f-hierarchy $\mathcal{B}$.
- Hence, the right adjoint $\mathrm{G}\colon \mathcal{B}_0 \to \mathcal{B}_\mathbf{V}$ is also unique, à la Freyd Adjoint Functor Theorem (FAFT).

Cross-f-hierarchy parallelism via epi-Adjunction

tory

G

Freyd Adjoint Functor Theorem (poset version)

In a Galois connection $\mathcal{P}_{\leqslant} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{Q}_{\sqsubseteq}$ where $\mathcal{P}_{\leqslant}$ has all joins and $\mathcal{Q}_{\sqsubseteq}$ has all meets, adjoint Functors in a pair $F \dashv G$ uniquely determine each other by the formulae:

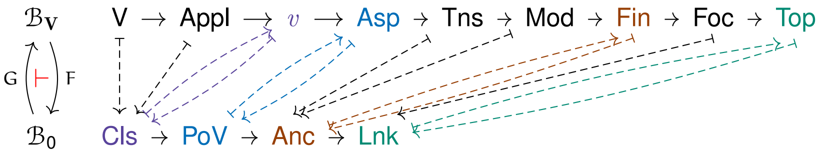
$$F(p) = \min\{q \in \mathcal{Q}_{\sqsubseteq} \mid p \leqslant G(q)\}$$

$$\boxed{G(q) = \max\{p \in \mathcal{P}_{\leqslant} \mid F(p) \sqsubseteq q\}}$$

for any $p \in \mathcal{P}_{\leqslant}$, $q \in \mathcal{Q}_{\sqsubseteq}$.

Cross-f-hierarchy parallelism via epi-Adjunction

Let's examine the f-hierarchy epi-Adjunction more closely: $\boxed{\text{expository}}$



- The right adjoint chooses a "representative" for each f-domain. FAFT says this is the **highest** category in each f-domain.

Cross-f-hierarchy parallelism via epi-Adjunction

Let's examine the f-hierarchy epi-Adjunction more closely:

$\boxed{\text{expository}}$

$$\begin{tikzcd} \mathcal{B}_{\mathbf{V}} \arrow[d, bend left, "F"] \arrow[d, phantom, "\vdash"] \\ \mathcal{B}_0 \arrow[u, bend left, "G"] \end{tikzcd}$$

$$\mathrm{V} \rightarrow \mathrm{Appl} \rightarrow v \rightarrow \mathrm{Asp} \rightarrow \mathrm{Tns} \rightarrow \mathrm{Mod} \rightarrow \mathrm{Fin} \rightarrow \mathrm{Foc} \rightarrow \mathrm{Top}$$

$$\mathrm{Cls} \rightarrow \mathrm{PoV} \rightarrow \mathrm{Anc} \rightarrow \mathrm{Lnk}$$

- The right adjoint chooses a "representative" for each f-domain. FAFT says this is the **highest** category in each f-domain.
- The mathematically determined representatives coincide with our linguistically special categories, i.e. **core functional** or **phase** categories.

Cross-f-hierarchy parallelism via epi-Adjunction

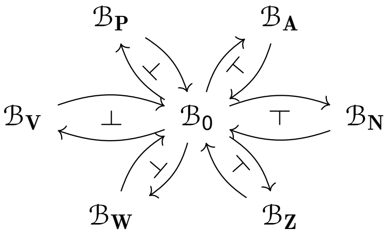
Let $\mathcal{B}_{\mathbf{V}}$ vary (e.g. $\mathcal{B}_{\mathbf{N}}$, $\mathcal{B}_{\mathbf{P}}$, or any other f-hierarchy a language variety may have) and we obtain a **flower-shaped** configuration

$$
\begin{tikzcd}
 & \mathcal{B}_{\mathbf{P}} \arrow[dr, bend left] \arrow[dr, phantom, "\vdash"] & & \mathcal{B}_{\mathbf{A}} \arrow[dl, bend left] & \\
\mathcal{B}_{\mathbf{V}} \arrow[rr, bend left] \arrow[rr, phantom, "\perp"] & & \mathcal{B}_{0} \arrow[ll, bend left] \arrow[rr, bend left] \arrow[rr, phantom, "\top"] \arrow[ul, bend left] \arrow[ur, bend left] \arrow[ur, phantom, "\dashv"] \arrow[dl, bend left] \arrow[dr, bend left] & & \mathcal{B}_{\mathbf{N}} \arrow[ll, bend left] \\
 & \mathcal{B}_{\mathbf{W}} \arrow[ur, bend left] \arrow[ur, phantom, "\vdash"] & & \mathcal{B}_{\mathbf{Z}} \arrow[ul, bend left] \arrow[ul, phantom, "\dashv"] &
\end{tikzcd}
$$

where the center is the u-spine and the petals are the f-hierarchies.

Cross-f-hierarchy parallelism via epi-Adjunction

Let $\mathcal{B}_{\mathbf{V}}$ vary (e.g. $\mathcal{B}_{\mathbf{N}}$, $\mathcal{B}_{\mathbf{P}}$, or any other f-hierarchy a language variety may have) and we obtain a **flower-shaped** configuration



$\boxed{\text{1 flower = 1 f-category inventory}}$

where the center is the u-spine and the petals are the f-hierarchies.

Cross-f-hierarchy parallelism via epi-Adjunction

Depending on one’s assumption about the **major parts of speech**, the “categorial flower” may have more or fewer petals

$$\begin{tikzcd}
& \mathcal{B}_{\mathbf{P}} \arrow[d, bend left] \arrow[d, phantom, "\vdash"] & \\
\mathcal{B}_{\mathbf{V}} \arrow[r, bend left] \arrow[r, phantom, "\perp"] & \mathcal{B}_0 \arrow[l, bend left] \arrow[u, bend left] \arrow[r, bend left] \arrow[d, bend left] \arrow[r, phantom, "\top"] & \mathcal{B}_{\mathbf{N}} \arrow[l, bend left] \\
& \mathcal{B}_{\mathbf{A}} \arrow[u, bend left] \arrow[u, phantom, "\dashv"] &
\end{tikzcd}$$

$$\begin{tikzcd}
& \mathcal{B}_{\mathbf{P}} \arrow[d, bend left] \arrow[d, phantom, "\vdash"] & \\
& \mathcal{B}_0 \arrow[u, bend left] \arrow[dl, bend left] \arrow[dr, bend left] & \\
\mathcal{B}_{\mathbf{V}} \arrow[ur, bend left] \arrow[ur, phantom, "\perp"] & & \mathcal{B}_{\mathbf{N}} \arrow[ul, bend left] \arrow[ul, phantom, "\top"]
\end{tikzcd}$$

$$\begin{tikzcd}
\mathcal{B}_{\mathbf{V}} \arrow[r, bend left] \arrow[r, phantom, "\perp"] & \mathcal{B}_0 \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] & \mathcal{B}_{\mathbf{N}} \arrow[l, bend left]
\end{tikzcd}$$

but all such flowers are constructed by **joint epi-Adjunction**.

Cross-f-hierarchy parallelism via epi-Adjunction

We can also change $\mathcal{B}$ to $\mathcal{P}$, $\mathcal{Q}$, CFC, Cart . . . and obtain a garden of categorial flowers, e.g.

$$\mathcal{P}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathcal{P}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathcal{P}_{\mathbf{N}} \qquad \mathcal{Q}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathcal{Q}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathcal{Q}_{\mathbf{N}} \qquad \mathcal{S}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathcal{S}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathcal{S}_{\mathbf{N}}$$

$$\mathrm{Ph}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathrm{Ph}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathrm{Ph}_{\mathbf{N}} \quad \mathrm{CFC}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathrm{CFC}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathrm{CFC}_{\mathbf{N}} \quad \mathrm{Cart}_{\mathbf{V}} \underset{\longleftarrow}{\overset{\longrightarrow}{\perp}} \mathrm{Cart}_0 \underset{\longleftarrow}{\overset{\longrightarrow}{\top}} \mathrm{Cart}_{\mathbf{N}}$$

There is further **global interconnection** at the garden level.

Global interconnection

Recall:

- Category division hierarchy
- Granularity level stacking

Global interconnection

Recall:

- Category division hierarchy
- Granularity level stacking

$\boxed{\text{1 flower = 1 granularity level}}$

Both are beyond individual f-hierarchies and even individual adult speakers, since the f-category inventory may

- change over time in an individual
- vary across language varieties

Global interconnection

Recall:

- Category division hierarchy
- Granularity level stacking

$\boxed{\text{1 flower = 1 granularity level}}$

Both are beyond individual f-hierarchies and even individual adult speakers, since the f-category inventory may

- change over time in an individual
- vary across language varieties

The garden is a collection of **all theoretically possible** f-category inventories, call it the Granularity Level Space (GLS).

Global interconnection

Three global relations between f-hierarchies X and Y

Global interconnection

Three global relations between f-hierarchies X and Y

- Parallel: via universal spine (as we have seen)

Global interconnection

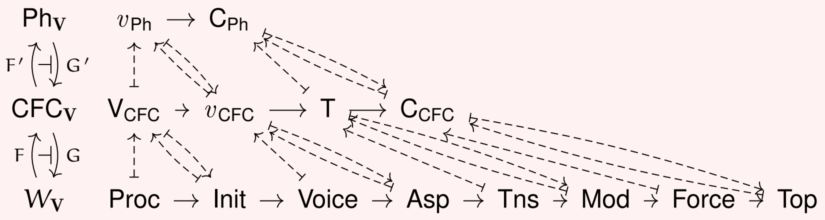
Three global relations between f-hierarchies X and Y

- Parallel: via universal spine (as we have seen)
- Stackable: **direct** and **composable** epi-Adjunctions, e.g.

Global interconnection

Epi-Adjunctions across granularity levels

$$\mathrm{F} \dashv \mathrm{G} \wedge \mathrm{F}' \dashv \mathrm{G}' \Rightarrow \mathrm{F}' \circ \mathrm{F} \dashv \mathrm{G} \circ \mathrm{G}'$$



Global interconnection

Three global relations between f-hierarchies X and Y

- Parallel: via universal spine (as we have seen)
- Stackable: **direct** and **composable** epi-Adjunctions, e.g.

Global interconnection

Three global relations between f-hierarchies X and Y

- Parallel: via universal spine (as we have seen)
- Stackable: **direct** and **composable** epi-Adjunctions, e.g.
- Incomparable: neither parallel nor stackable, e.g.

$$\mathrm{V}_{\mathrm{CFC}} \to v_{\mathrm{CFC}} \to \mathrm{C}_{\mathrm{Ph}} \qquad v_{\mathrm{Ph}} \to \mathrm{T} \to \mathrm{C}_{\mathrm{CFC}}$$

Global interconnection

Parallel + stackable $\Rightarrow$ a fully connected corner in the GLS

$$
\begin{tikzcd}
\mathrm{Ph}_{\mathbf{V}} \arrow[r, bend left] \arrow[r, phantom, "\perp"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathrm{Ph}_0 \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathrm{Ph}_{\mathbf{N}} \arrow[l, bend left] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] \\
\mathrm{CFC}_{\mathbf{V}} \arrow[u, bend left] \arrow[r, bend left] \arrow[r, phantom, "\perp"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathrm{CFC}_0 \arrow[u, bend left] \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathrm{CFC}_{\mathbf{N}} \arrow[u, bend left] \arrow[l, bend left] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] \\
\mathrm{Cart}_{\mathbf{V}} \arrow[u, bend left] \arrow[r, bend left] \arrow[r, phantom, "\perp"] & \mathrm{Cart}_0 \arrow[u, bend left] \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] & \mathrm{Cart}_{\mathbf{N}} \arrow[u, bend left] \arrow[l, bend left]
\end{tikzcd}
$$

Global interconnection

Abstract away from the internal details of flowers

$$\begin{tikzcd}
\mathsf{Ph}_{\mathbf{V}} \arrow[r, bend left] \arrow[r, phantom, "\perp"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathsf{Ph}_0 \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathsf{Ph}_{\mathbf{N}} \arrow[l, bend left] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathbf{Ph} \arrow[d, bend left, "f"] \\
\mathsf{CFC}_{\mathbf{V}} \arrow[u, bend left] \arrow[r, bend left] \arrow[r, phantom, "\perp"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathsf{CFC}_0 \arrow[u, bend left] \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathsf{CFC}_{\mathbf{N}} \arrow[u, bend left] \arrow[l, bend left] \arrow[d, bend left] \arrow[d, phantom, "\dashv"] & \mathbf{CFC} \arrow[u, bend left, "f^{-1}"] \arrow[d, bend left, "f'"] \\
\mathsf{Cart}_{\mathbf{V}} \arrow[u, bend left] \arrow[r, bend left] \arrow[r, phantom, "\perp"] & \mathsf{Cart}_0 \arrow[u, bend left] \arrow[l, bend left] \arrow[r, bend left] \arrow[r, phantom, "\top"] & \mathsf{Cart}_{\mathbf{N}} \arrow[u, bend left] \arrow[l, bend left] & \mathbf{Cart} \arrow[u, bend left, "f'^{-1}"]
\end{tikzcd}$$

and we get an isomorphism $\mathbf{Ph} \cong \mathbf{CFC} \cong \mathbf{Cart}$.

Global interconnection

Parallel + stackable + incomparable $\xRightarrow{\text{zoom out}}$ GLS poset

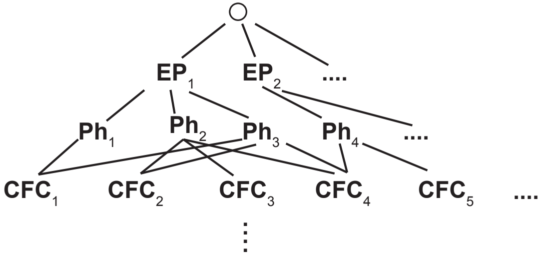


Figure 5: Granularity Level Space (GLS) ordered by inheritance.

☞ This is the highest abstraction layer for functional hierarchies.

Global interconnection

Parallel + stackable + incomparable $\xRightarrow{\text{zoom out}}$ GLS poset

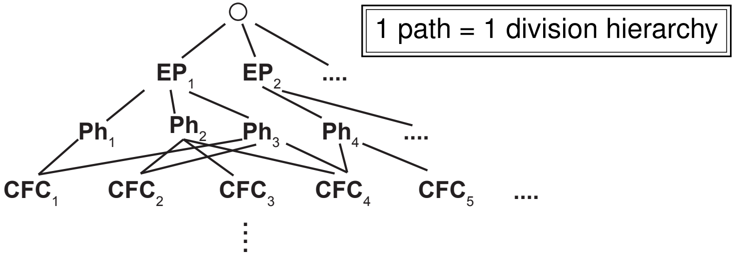


Figure 5: Granularity Level Space (GLS) ordered by inheritance.

☞ This is the highest abstraction layer for functional hierarchies.

Entire SCS

Finally we add in acategorial categories

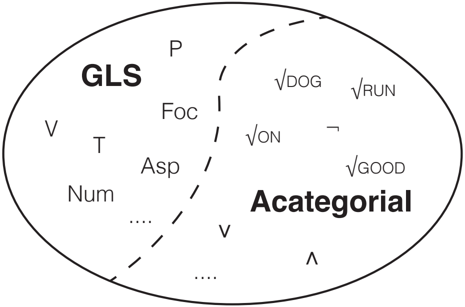


Figure 6: The entire Syntactic Category System (SCS).

Entire SCS

And put the SCS in a larger universe. . .

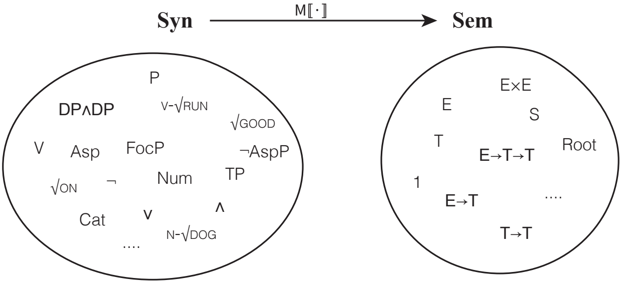


Figure 7: Syntax and Semantics connected by interpretation Functor.

Entire SCS

And put the SCS in a larger universe. . .

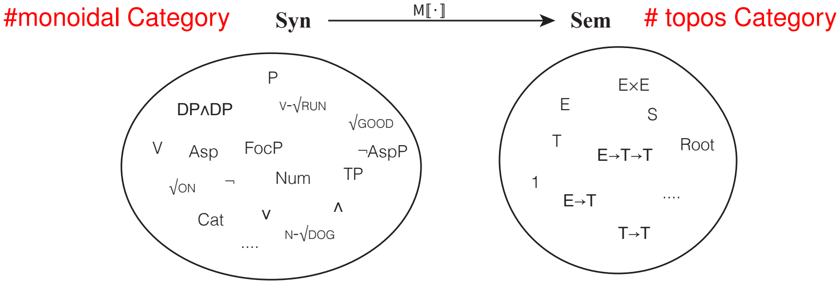
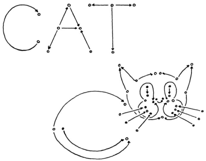


Figure 7: Syntax and Semantics connected by interpretation Functor.

☞ **Syn** as a Category has more structures than partial orders.

Conclusion



Category Theory has been called “abstract nonsense”, but it provides a very sensible metalanguage to describe the **ontological** organization of the Syntactic Category System (SCS).

- A functional hierarchy is a poset Category.
- A f-category inventory is a tiny Category of posets.
- All possible f-category inventories form a huge poset.

Conclusion

An important Categorical relation **epi-Adjunction**

- indirectly formalizes cross-functional-hierarchy parallelism (via universal spine)
- directly formalizes cross-granularity-level inheritance (via Adjunction composition)

(More details are in my dissertation, Song 2019).

$$
\begin{tikzcd}
 & \mathcal{B}_{\mathbf{P}} \arrow[dr, bend left] & & \mathcal{B}_{\mathbf{A}} \arrow[dl, bend left] & \\
\mathcal{B}_{\mathbf{V}} \arrow[rr, bend left] & & \mathcal{B}_{0} \arrow[ll, bend left] \arrow[rr, bend left] \arrow[ul, bend left] \arrow[ur, bend left] \arrow[dl, bend left] \arrow[dr, bend left] \arrow[ll, phantom, "\perp"] \arrow[rr, phantom, "\top"] & & \mathcal{B}_{\mathbf{N}} \arrow[ll, bend left] \\
 & \mathcal{B}_{\mathbf{W}} \arrow[ur, bend left] & & \mathcal{B}_{\mathbf{Z}} \arrow[ul, bend left] &
\end{tikzcd}
$$

References I

Fong, B. & D. Spivak
Seven Sketches in Compositionality: An Invitation to Applied Category Theory.
MIT, 2018.

Wiltschko, M.
The Universal Structure of Categories: Towards a Formal Typology.
CUP, 2014.

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

References II

Erné, M.
Adjunctions and Galois connections: Origins, history and development.
Galois Connections and Applications.
Springer Science+Business Media, B.V., 2004.

Song, C.
Formal flexibility of syntactic categories.
PhD dissertation, 2019 (in progress).