Tutorial: Logic and Model Theory for Phonology

Jane Chandlee (Haverford College), Adam Jardine (Rutgers University), and Scott Nelson (Stony Brook University)

LSA 2024 | NYC

6 January 2024

Overview

- Two things that are important to phonologists are:
  - **Representations**
    Features
    Autosegments
    Gestures
    ...
  - **Maps**
    E.g., final devoicing:
    /bɛd/ ↦ [bɛt]
    /akab/ ↦ [akap]
    /bɛn/ ↦ [bɛn]
    /azaz/ ↦ [azas]
    ⋮

This Tutorial

1. **Model Theory**

2. **First-order Logic**

3. **Interpretations**

This Tutorial

1. Model Theory

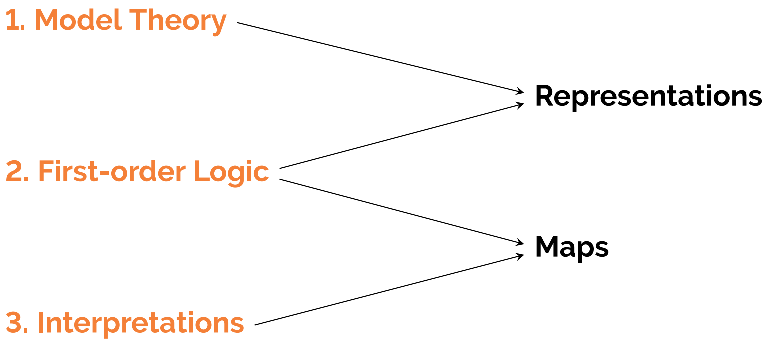
Representations

2. First-order Logic

Maps

3. Interpretations

This Tutorial



Model Theory

- What is the difference between these two words?

  $[\mathrm{aaa}]$ and $[\mathrm{aaaa}]$

- What is the difference between these two words?

  $[\mathrm{barp}]$ and $[\mathrm{brap}]$

- What is the difference between these two words?

  $[\mathrm{barp}]$ and $[\mathrm{parp}]$

Model Theory

- What is the difference between these two words?

  $[\mathrm{aaa}]$ and $[\mathrm{aaaa}]$

  elements of the structure

- What is the difference between these two words?

  $[\mathrm{barp}]$ and $[\mathrm{brap}]$

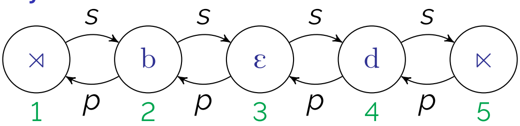
  order of elements

- What is the difference between these two words?

  $[\mathrm{barp}]$ and $[\mathrm{parp}]$

  properties of the elements

Model Theory



- **indices**
- **order functions** *p* and *s*
- **properties** of the indices
- A *model signature* is a collection of functions and relations that are used to describe structures:

$$\{\mathbf{p}, \mathbf{s}, \mathbf{S_1} \ldots \mathbf{S_n}\}$$

- A *model* is a structure in some signature:

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{S_1} \ldots \mathbf{S_n} \rangle$$

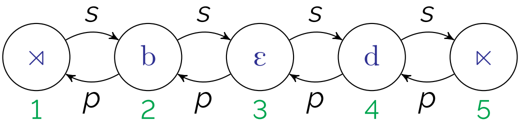
Model Theory

- Model Signatures for Phonological Representations

Segment strings	$\{\mathbf{p}, \mathbf{s}, \mathbf{S_1} \ldots \mathbf{S_n}\}$
Feature strings	$\{\mathbf{p}, \mathbf{s}, \mathbf{F_1} \ldots \mathbf{F_n}\}$
Autosegmental structures	$\{\mathbf{p}, \mathbf{s}, \mathbf{A}, \mathbf{F_1} \ldots \mathbf{F_n}\}$
Syllable trees	$\{\mathbf{p}, \mathbf{s}, \mathbf{parent}, \mathbf{ons}, \mathbf{nuc}, \mathbf{cod}, \sigma\}$
Sign language structures	$\{\mathbf{p}, \mathbf{s}, \mathbf{A}, \mathbf{loc}, \mathbf{L}, \mathbf{M}, \mathbf{H_i}, \mathbf{P_i}\}$
Articulatory structures	$\{\mathbf{0}, \mathbf{30}, \mathbf{60}, \mathbf{180}, \mathbf{G_1} \ldots \mathbf{G_n}\}$

Jardine (2017); Chandlee and Jardine (2019); Strother-Garcia (2019); Jardine et al. (2021); Oakden (2020); Rawski (2020); Chadwick (2021); Nelson (2022, 2023)

Segment strings



D = {1, 2, 3, 4, 5}

p(1) = 2, p(2) = 3, p(3) = 4, etc.

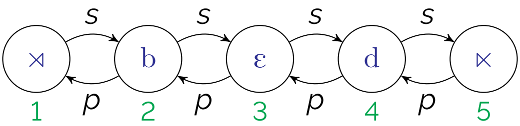
s(5) = 4, s(4) = 3, s(3) = 2, etc.

b = {2}

ε = {3}

d = {4}

Feature strings



D = {1, 2, 3, 4, 5}

⋮

voice = {2,3,4}

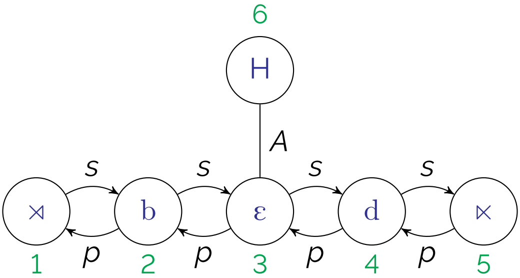
obs = {2,4}

syll = {3}

cor = {4}

etc.

Autosegmental structures

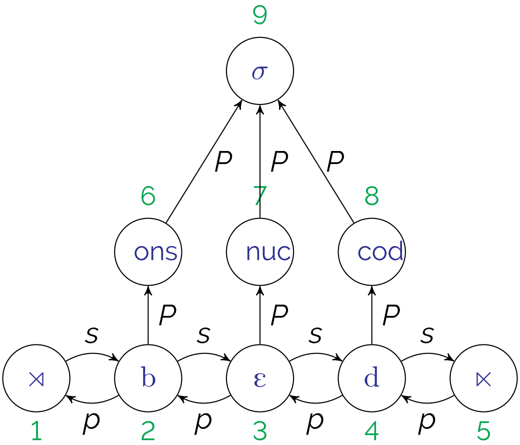


D = {1, 2, 3, 4, 5, 6}

⋮

A(6) = 3, A(3) = 6

Syllable trees



D = {1, 2, 3, 4, 5, 6, 7, 8, 9}

⋮

P(arent)(2) = 6, P(3) = 7, P(4) = 8, P(6) = 9, P(7) = 9, P(8) = 9

First-order Logic

- Why logic?
  - Logic allows us to formalize our grammars/theories as sets of axioms that we can use to formally analyze and compare the types of structures that comply with a given theory.
  - The computational complexity of logics are well known (McNaughton and Papert, 1971; Simon, 1975; Immerman, 1980; Rogers et al., 2013, et seq.)
  - We can study the interaction of complexity and representation by changing the model while keeping the power of the logic fixed.
  - Logical formalisms make for strong hypotheses about the complexity of phonology (Rogers et al., 2013; Heinz, 2018)

First-order Logic

- First-order logic describes truth conditions of structures

	Name	Meaning
x, y, z	variables	**Elements**
$R_1 ... R_n$	relations	**Order**/**Properties**
$F_1 ... F_n$	functions	**Order**/**Properties**
$\wedge$	conjunction	“And”
$\vee$	disjunction	“Or”
$\neg$	negation	“Not”
$\rightarrow$	implication	“If...then”
$\leftrightarrow$	bi-direction	“Same”
$\exists$	existential quantifier	“There Exists”
$\forall$	universal quantifier	“For All”

First-order logic

- For any signature Σ, a Σ-formula is a logical formula where all the non-logical symbols are drawn from Σ.

- Suppose $\Sigma = \langle \mathbf{p}, \mathbf{s}, \mathbf{C}, \mathbf{V}, \rtimes, \ltimes \rangle$, which of the following are Σ-formulas?

$$\mathbf{V}(x) \wedge \ltimes(\mathbf{s}(x))$$
$$\mathbf{G}(x) \wedge \ltimes(\mathbf{s}(x))$$
$$\mathbf{G}(x) \wedge \ltimes(\mathbf{A}(x))$$

First-order logic

- For any signature Σ, a Σ-formula is a logical formula where all the non-logical symbols are drawn from Σ.

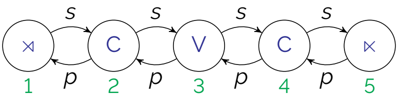
- Suppose $\Sigma = \langle \mathbf{p}, \mathbf{s}, \mathbf{C}, \mathbf{V}, \rtimes, \ltimes \rangle$, which of the following are Σ-formulas?

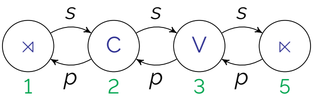
$\mathbf{V}(x) \land \ltimes(\mathbf{s}(x))$	✓
$\mathbf{G}(x) \land \ltimes(\mathbf{s}(x))$	✗
$\mathbf{G}(x) \land \ltimes(\mathbf{A}(x))$	✗

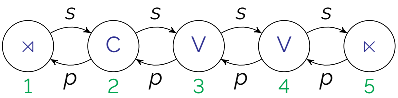
First-order logic

- If A is a structure built from Σ and φ is a Σ-formula, then we write $A \models \varphi$ if $\varphi(A)$ evaluates to `true` and say A satisfies (or *models*) φ. Otherwise, A does not satisfy/model φ.

- Which of the following structures satisfy $\mathbf{V}(x) \wedge \ltimes(\mathbf{s}(x))$?



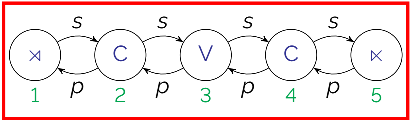


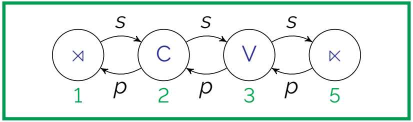


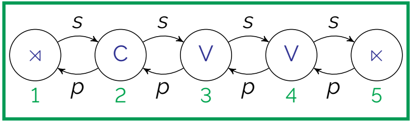
First-order logic

- If A is a structure built from Σ and φ is a Σ-formula, then we write $A \models \varphi$ if $\varphi(A)$ evaluates to `true` and say A satisfies (or *models*) φ. Otherwise, A does not satisfy/model φ.

- Which of the following structures satisfy $\mathbf{V}(x) \wedge \ltimes(\mathbf{s}(x))$?







Maps as interpretations

Phonologists care about maps!

$$[-\text{son}] \;\rightarrow\; [-\text{voi}] \;/\; \underline{\quad\quad} \;\#$$

$$\textsc{Faith}, {}^{*}\text{D}\# \gg \text{ID(voi)}$$

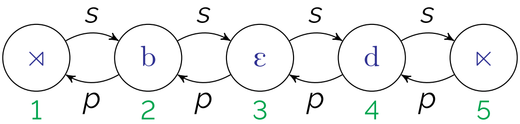
$$\Downarrow$$

$$/\text{b}\varepsilon\text{d}/ \mapsto [\text{b}\varepsilon\text{t}]$$
$$/\text{akab}/ \mapsto [\text{akap}]$$
$$/\text{b}\varepsilon\text{n}/ \mapsto [\text{b}\varepsilon\text{n}]$$
$$/\text{azaz}/ \mapsto [\text{azas}]$$
$$\vdots$$

Maps as interpretations

Defining new relations

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$

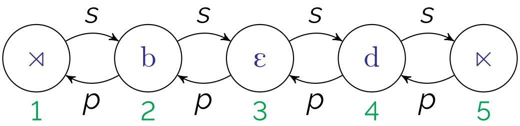


$$\mathbf{wdfinalobs}(x) \equiv \neg\mathbf{son}(x) \wedge \ltimes(s(x))$$

Maps as interpretations

Defining new relations

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$



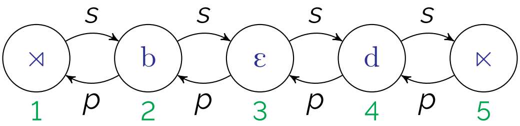
$$\mathbf{wdfinalobs}(x) \equiv \neg\mathbf{son}(x) \wedge \ltimes(s(x))$$

	1	2	3	4	5
$\mathbf{son}(x)$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$\mathbf{voi}(x)$	$\bot$	$\top$	$\top$	$\top$	$\bot$
$\mathbf{wdfinalobs}(x)$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$

Maps as interpretations

Defining new relations

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$



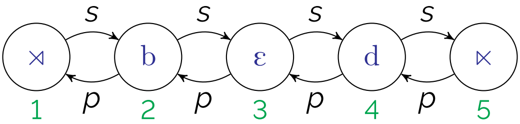
$$\mathbf{wdfinalobs}(x) \equiv \neg\mathbf{son}(x) \wedge \ltimes(s(x))$$

	1	2	3	4	5
$\mathbf{son}(x)$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$\mathbf{voi}(x)$	$\bot$	$\top$	$\top$	$\top$	$\bot$
$\mathbf{wdfinalobs}(x)$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
$\neg\mathbf{wdfinalobs}(x)$	$\top$	$\top$	$\top$	$\bot$	$\top$

Maps as interpretations

Defining new structures

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$

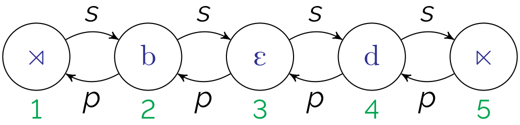


$\mathbf{son}'(x) \equiv \ldots$
$\mathbf{voi}'(x) \equiv \ldots$

Maps as interpretations

Defining new structures

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$

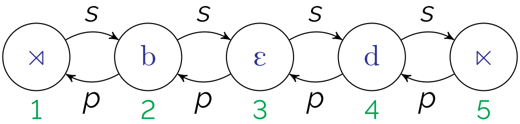


$$\mathbf{son}'(x) \equiv \mathbf{son}(x)$$
$$\mathbf{voi}'(x) \equiv \ldots$$

Maps as interpretations

Defining new structures

$$\langle \mathbf{D}; \mathbf{p}, \mathbf{s}, \mathbf{voi}, \mathbf{son}, \rtimes, \ltimes \rangle$$



$$\mathbf{son}'(x) \equiv \mathbf{son}(x)$$
$$\mathbf{voi}'(x) \equiv \mathbf{voi}(x) \wedge \neg\mathbf{wdfinalobs}(x)$$

Maps as interpretations

Defining new structures

$$\textbf{son}'(x) \equiv \textbf{son}(x)$$
$$\textbf{voi}'(x) \equiv \textbf{voi}(x) \wedge \neg\textbf{wdfinalobs}(x)$$

	⋊	b	ɛ	d	⋉
	1	2	3	4	5
son′(*x*)	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
voi′(*x*)	$\bot$	$\top$	$\top$	$\bot$	$\bot$

Maps as interpretations

Defining new structures

$$\textbf{son}'(x) \equiv \textbf{son}(x)$$
$$\textbf{voi}'(x) \equiv \textbf{voi}(x) \wedge \neg\textbf{wdfinalobs}(x)$$

	$\rtimes$	b	ɛ	d	$\ltimes$
	1	2	3	4	5
son′(x)	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
voi′(x)	$\bot$	$\top$	$\top$	$\bot$	$\bot$
	1′	2′	3′	4′	5′
	$\rtimes$	b	ɛ	t	$\ltimes$

Maps as interpretations

Defining new structures

$$\textbf{son}'(x) \equiv \textbf{son}(x)$$
$$\textbf{voi}'(x) \equiv \textbf{voi}(x) \wedge \neg\textbf{wdfinalobs}(x)$$

	$\rtimes$	a	k	a	b	$\ltimes$
	1	2	3	4	5	6
$\textbf{son}'(x)$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
$\textbf{voi}'(x)$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
	1′	2′	3′	4′	5′	6′
	$\rtimes$	a	k	a	p	$\ltimes$

Maps as interpretations

- Maps so defined are **local** (Chandlee and Lindell, forthcoming)

Maps as interpretations

Recursive definitions

Iterative stress

$$\begin{array}{lcl} \sigma & \mapsto & \acute{\sigma} \\ \sigma\sigma & \mapsto & \acute{\sigma}\sigma \\ \sigma\sigma\sigma & \mapsto & \acute{\sigma}\sigma\acute{\sigma} \\ \sigma\sigma\sigma\sigma & \mapsto & \acute{\sigma}\sigma\acute{\sigma}\sigma \\ \sigma\sigma\sigma\sigma\sigma & \mapsto & \acute{\sigma}\sigma\acute{\sigma}\sigma\acute{\sigma} \\ \sigma\sigma\sigma\sigma\sigma\sigma & \mapsto & \acute{\sigma}\sigma\acute{\sigma}\sigma\acute{\sigma}\sigma \\ & \vdots & \end{array}$$

Maps as interpretations

Recursive definitions

Iterative, non-final stress (e.g., Pintupi)

$$\textbf{stress}'(x) \equiv \sigma(x) \wedge \rtimes(p(x))$$

	$\rtimes$	σ	σ	σ	σ	σ	σ	$\ltimes$
	1	2	3	4	5	6	7	8
$\textbf{stress}'(x)$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
	$1'$	$2'$	$3'$	$4'$	$5'$	$6'$	$7'$	$8'$
	$\rtimes$	$\acute{\sigma}$	σ	σ	σ	σ	σ	$\ltimes$

Maps as interpretations

Recursive definitions

Iterative, non-final stress (e.g., Pintupi)

$$\textbf{stress}'(x) \equiv \sigma(x) \wedge \big(\rtimes(p(x)) \vee \textbf{stress}'(p(p(x)))\big)$$

	$\rtimes$	σ	σ	σ	σ	σ	σ	$\ltimes$
	1	2	3	4	5	6	7	8
stress$'(x)$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
	$1'$	$2'$	$3'$	$4'$	$5'$	$6'$	$7'$	$8'$
	$\rtimes$	$\acute{\sigma}$	σ	$\acute{\sigma}$	σ	$\acute{\sigma}$	σ	$\ltimes$

Maps as interpretations

- Recursive, quantifier-free definitions are called **boolean monadic recursive schemes** (BMRS; Bhaskar et al., 2020; Chandlee and Jardine, 2021)
- Maps so defined are **subsequential** (Bhaskar et al., 2020), meaning that they are myopic (Wilson, 2003; Jardine, 2019)

Next Steps

- How do we learn logical grammars?
- What does a tertiary feature system look like in BMRS?
- BMRS captures elsewhere condition-type effects well. What about non-derived environment blocking?
- What is the status of intermediate representations?
- How does BMRS capture the typology of stress patterns?
- ... and many more!

References I

Bhaskar, S., Chandlee, J., Jardine, A., and Oakden, C. (2020). Boolean monadic recursive schemes as a logical characterization of the subsequential functions. In *Language and Automata Theory and Applications - LATA 2020*, Lecture Notes in Computer Science, pages 157–169. Springer.

Chadwick, S. (2021). A computational analysis for articulatory phonology: A duel example with the phonotactic properties of rhotic consonants in english. Master's thesis, Stony Brook University.

Chandlee, J. and Jardine, A. (2019). Autosegmental input-strictly local functions. *Transactions of the Association for Computational Linguistics*, 7:157–168.

Chandlee, J. and Jardine, A. (2021). Computational universals in linguistic theory: Using recursive programs for phonological analysis. *Language*, 93:485–519.

Chandlee, J. and Lindell, S. (forthcoming). A logical characterization of strictly local functions. In Heinz, J., editor, *Doing Computational Phonology*. OUP.

Heinz, J. (2018). The computational nature of phonological generalizations. In Hyman, L. and Plank, F., editors, *Phonological Typology*, Phonetics and Phonology, chapter 5, pages 126–195. De Gruyter Mouton.

Immerman, N. (1980). *First-order expressibility as a New Complexity Measure*. PhD thesis, Cornell University.

Jardine, A. (2017). On the logical complexity of autosegmental representations. In Kanazawa, M., de Groote, P., and Sadrzadeh, M., editors, *Proceedings of the 15th Meeting on the Mathematics of Language*, pages 22–35, London, UK. Association for Computational Linguistics.

Jardine, A. (2019). Computation also matters: a response to pater (2018). *Phonology*, 36(2):341–350.

Jardine, A., Danis, N., and Iacoponi, L. (2021). A formal investigation of q-theory in comparison to autosegmental representations. *Linguistic Inquiry*, 52(2):333–358.

McNaughton, R. and Papert, S. (1971). *Counter-Free Automata*. MIT Press.

Nelson, S. (2022). A model theoretic perspective on phonological feature systems. *Proceedings of the Society for Computation in Linguistics*, 5(1):1–10.

Nelson, S. (2023). Model theoretic phonology and theory comparison: Segments, gestures, and coupling graphs. Presentation at North American Phonology Conference 12.

Oakden, C. (2020). Notational equivalence in tonal geometry. *Phonology*, 37(2):257–296.

Rawski, J. (2020). The logical nature of phonology across speech and sign. Unpublished Manuscript.

Rogers, J., Heinz, J., Fero, M., Hurst, J., Lambert, D., and Wibel, S. (2013). Cognitive and sub-regular complexity. In *Formal Grammar*, volume 8036 of *Lecture Notes in Computer Science*, pages 90–108. Springer.