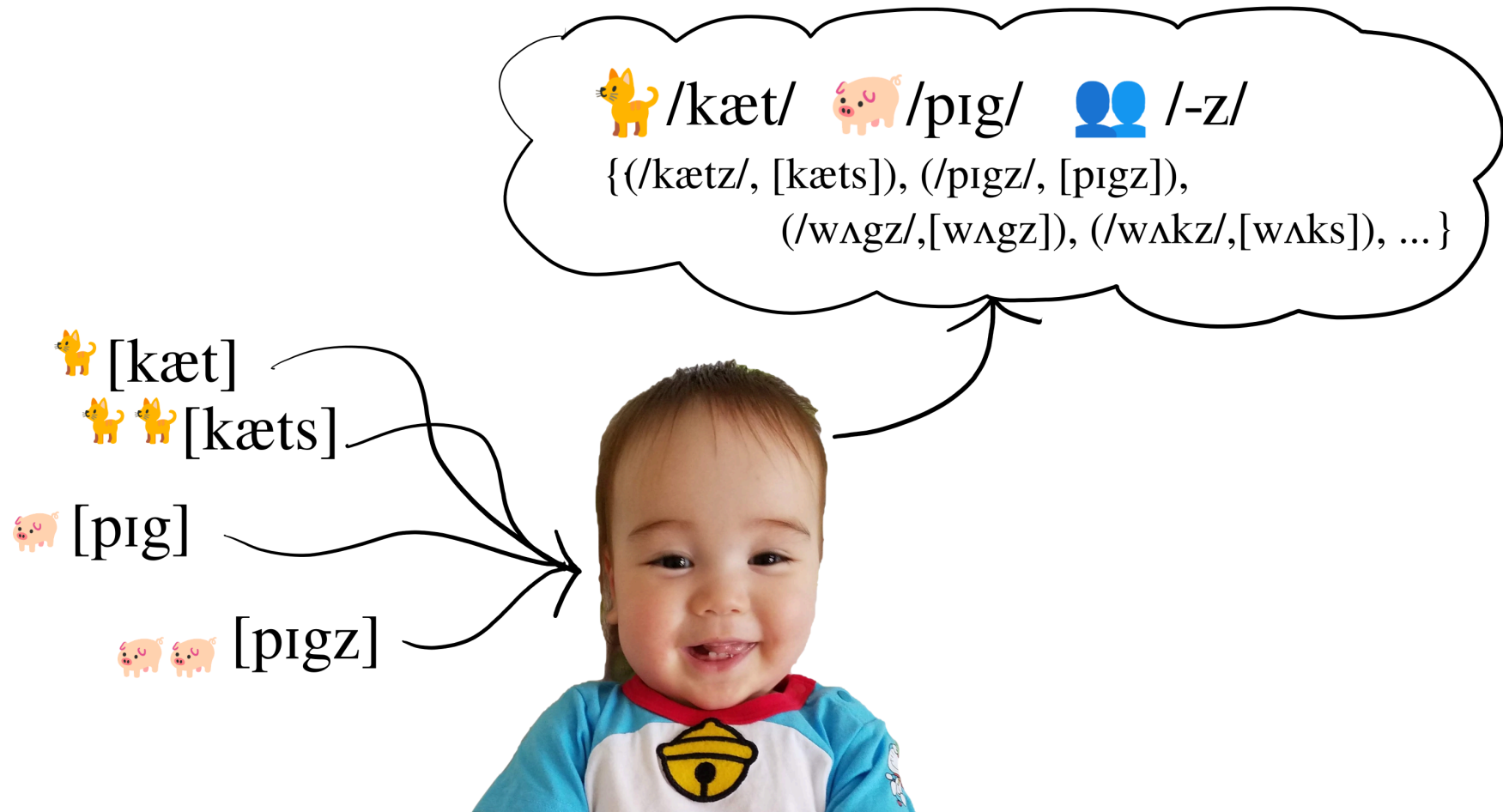


COMPUTATIONAL SIMPLICITY IS FUNDAMENTAL TO LANGUAGE LEARNING

Adam Jardine

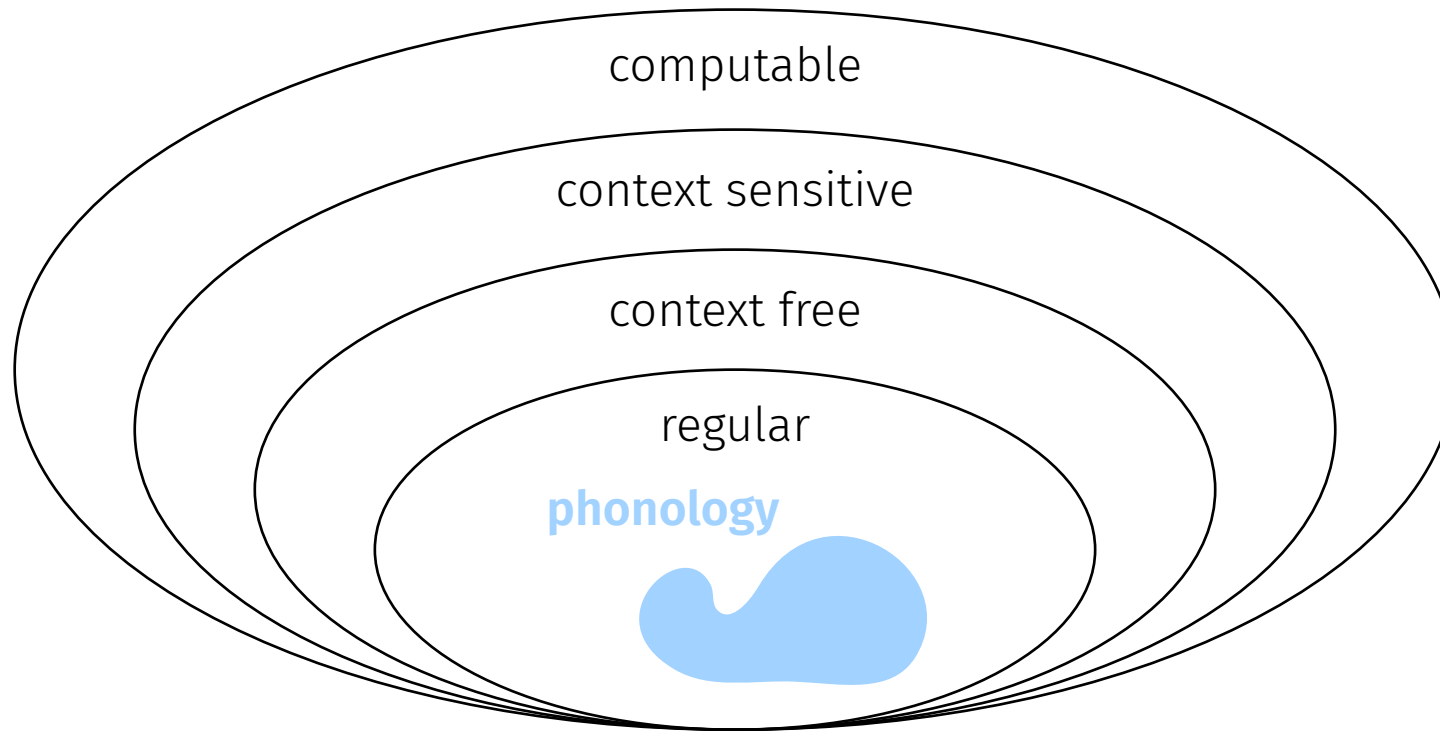
RuCCS Faculty Double Feature

October 13, 2026



Input

λ		φ		Finite sample of $\varphi \circ \lambda$		
	$\mapsto$	/z/	/kæt/	$\mapsto$	[kæt]	(, [kæt])
	$\mapsto$	/kæt/	/kætz/	$\mapsto$	[kæts]	(· , [kæts])
·	$\mapsto$	/kætz/	/pig/	$\mapsto$	[pig]	(, [pig])
	$\mapsto$	/pig/	/pigz/	$\mapsto$	[pigz]	(· , [pigz])
·	$\mapsto$	/pigz/	/kaũ/	$\mapsto$	[kaũ]	(, [kaũ])
	$\mapsto$	/kaũ/	/kaũz/	$\mapsto$	[kaũz]	(· , [kaũz])
·	$\mapsto$	/kaũz/	/wʌgz/	$\mapsto$	[wʌgz]	
	$\mapsto$	/wʌkz/	$\mapsto$	[wʌks]		
·						
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		



- This talk based on Chandlee & Jardine (2025)
- NSF Grant #2416184 with

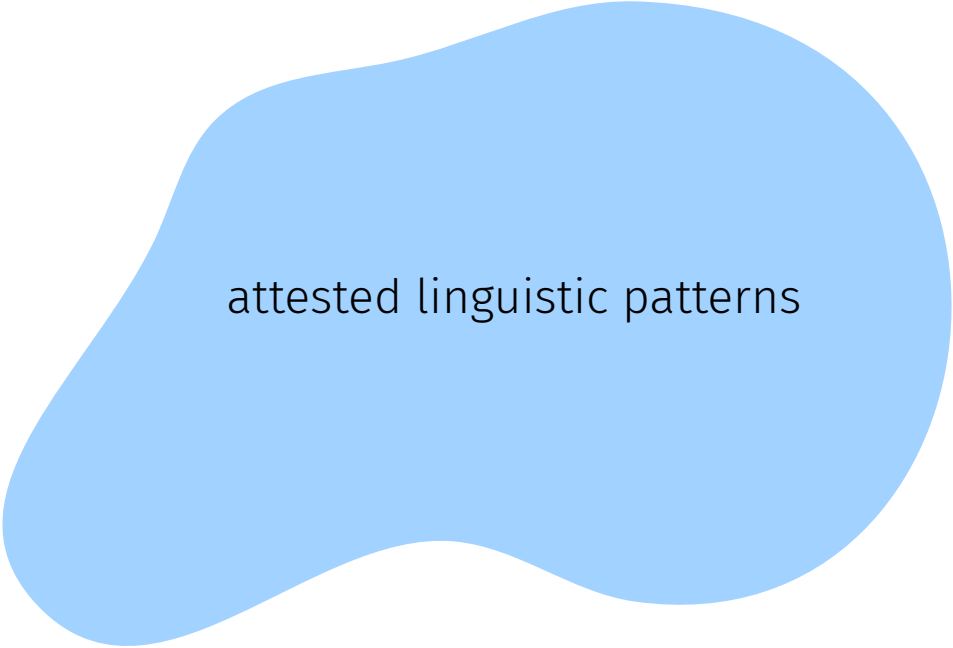


Jane Chandlee (Haverford) **Jeffrey Heinz** (Stony Brook)

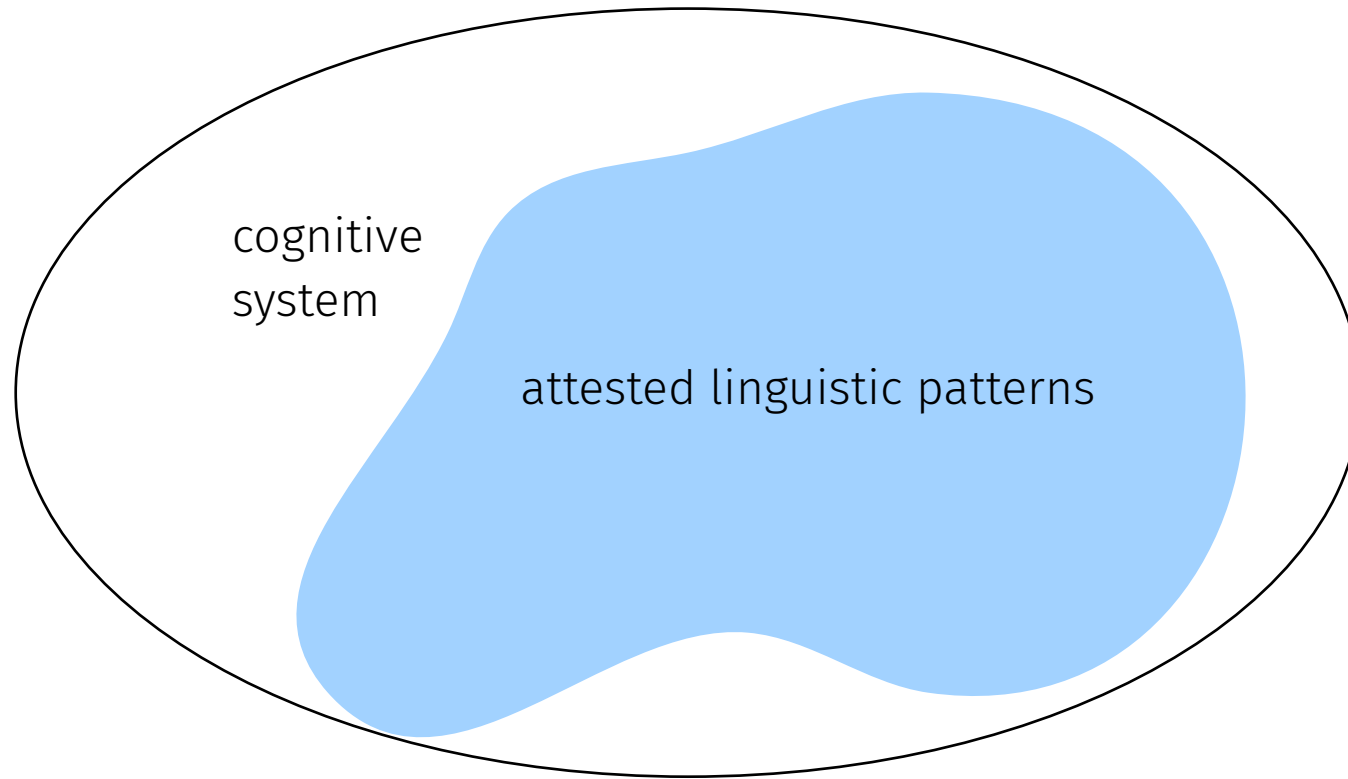


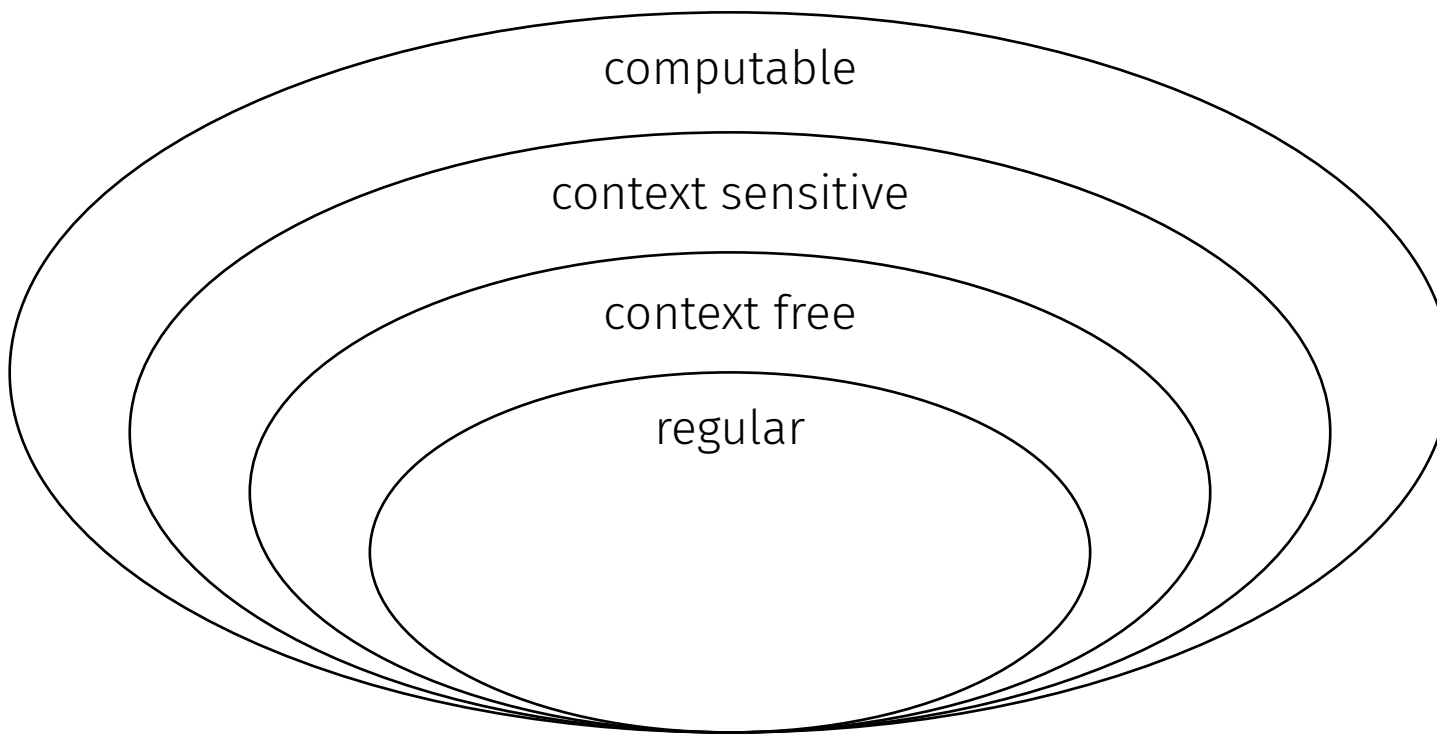
- **Grads:** Vincent Czarnecki Chenli Wang Eli Franz Zachary Helfand
Hyunjung Joo (now at Hanyang Univ.)
Huteng Dai (now at Univ. of Michigan)
Wenyue Hua (Ling MA, CS PhD, now at Microsoft Research)
- **UGs:** Ezra Anthony Neha Jeyaram Manish Kandepalli
Ben Evans (now at Stony Brook)

Linguistic universals and computation

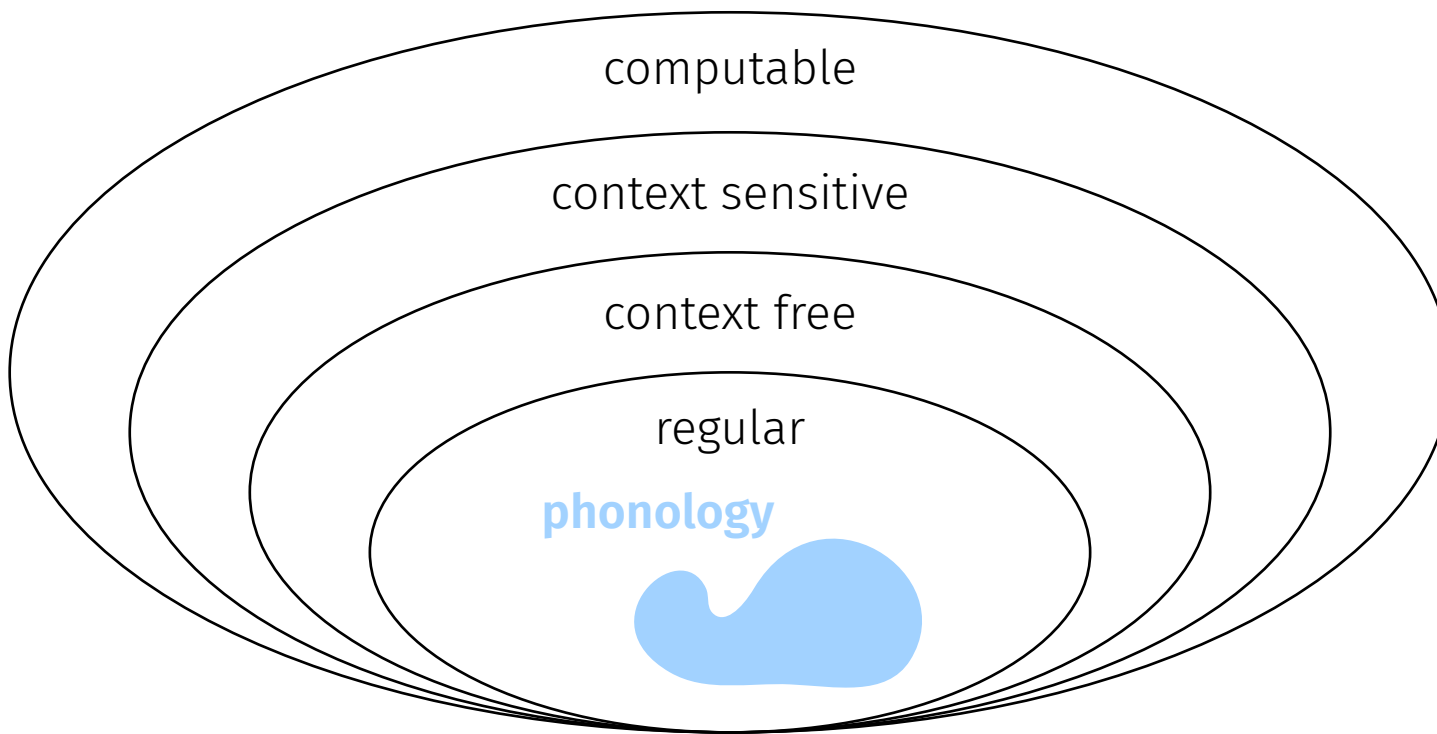


attested linguistic patterns



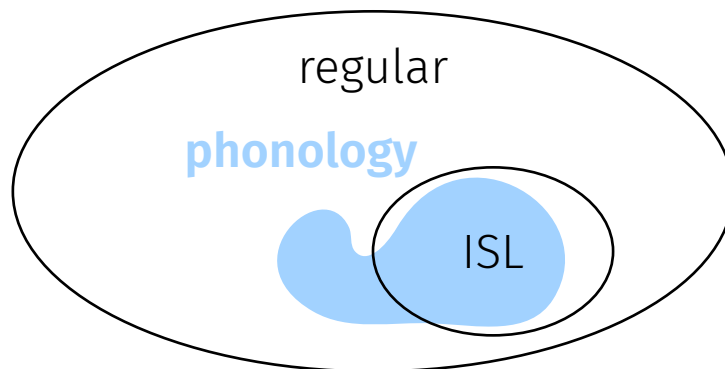


The Chomsky-Schützenberger Hierarchy (Chomsky 1957, Chomsky and Schützenberger 1963)

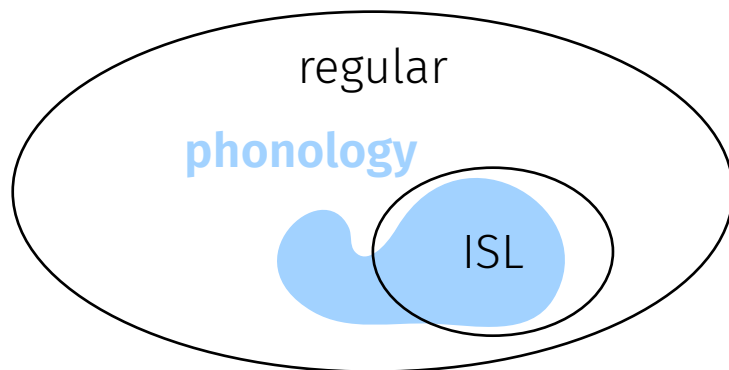


The Chomsky-Schützenberger Hierarchy (Chomsky 1957, Chomsky and Schützenberger 1963)
(Johnson 1972, Kaplan and Kay 1994, Heinz 2007, et seq.)

Phonology and local functions



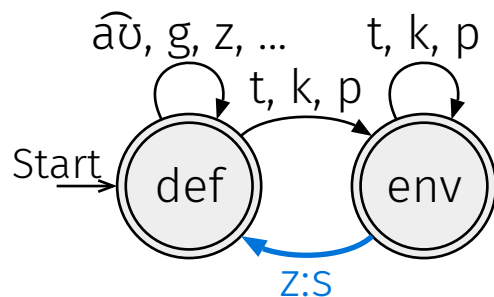
Input Strictly Local (ISL) functions
(Chandlee 2014, Chandlee and Heinz 2018)



ISL functions

(Chandlee 2014, Chandlee and Heinz 2018)

φ		
/kæt/	$\mapsto$	[kæt]
/kæt ^z /	$\mapsto$	[kæt ^s]
/pɪg/	$\mapsto$	[pɪg]
/pɪg ^z /	$\mapsto$	[pɪg ^z]
/kâũ/	$\mapsto$	[kâũ]
/kâũ ^z /	$\mapsto$	[kâũ ^z]
/wʌgz/	$\mapsto$	[wʌgz ^s]
/wʌk ^z /	$\mapsto$	[wʌk ^s]
⋮	⋮	⋮

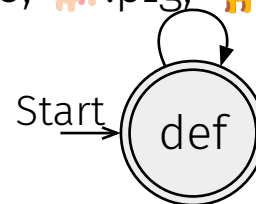


Finite-state machine for φ

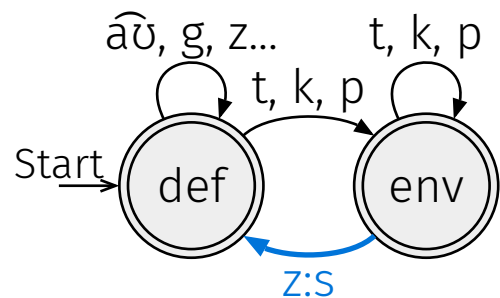
φ		
/kæt/	$\mapsto$	[kæt]
/kæt $\mathbf{z}$ /	$\mapsto$	[kæt $\mathbf{s}$]
/pɪg/	$\mapsto$	[pɪg]
/pɪg $\mathbf{z}$ /	$\mapsto$	[pɪg $\mathbf{z}$]
/kâû/	$\mapsto$	[kâû]
/kâû $\mathbf{z}$ /	$\mapsto$	[kâû $\mathbf{z}$]
/wʌg $\mathbf{z}$ /	$\mapsto$	[wʌg $\mathbf{z}$]
/wʌk $\mathbf{z}$ /	$\mapsto$	[wʌk $\mathbf{s}$]
⋮	⋮	⋮

λ		
	$\mapsto$	/z/
	$\mapsto$	/kæt/
 	$\mapsto$	/kætz/
	$\mapsto$	/pɪg/
 	$\mapsto$	/pɪgz/
	$\mapsto$	/kâũ/
 	$\mapsto$	/kâũz/
$\vdots$	$\vdots$	$\vdots$

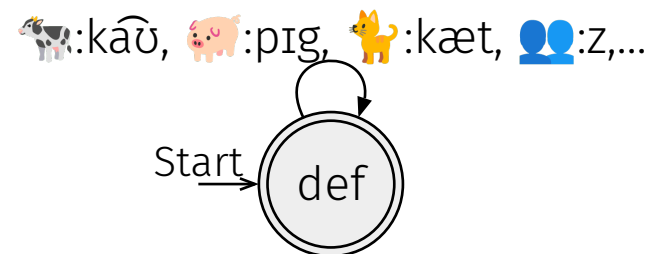
:kâũ, :pɪg, :kæt, :z,...



Finite-state machine for λ



Finite-state machine for φ

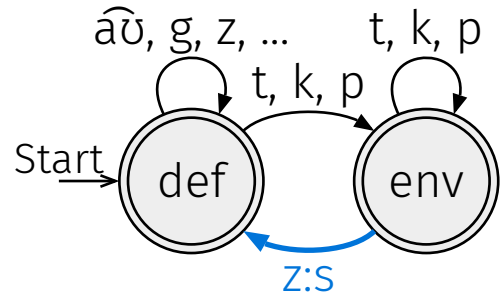


Finite-state machine for λ

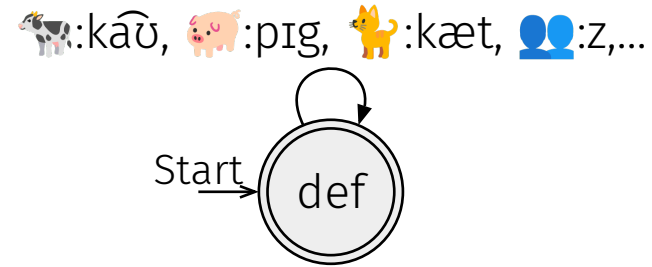
Towards a solution

Input

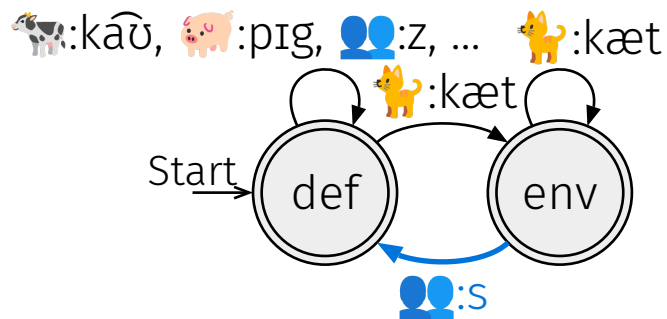
λ		φ		Finite sample of $\varphi \circ \lambda$		
	$\mapsto$	/z/	/kæt/	$\mapsto$	[kæt]	(, [kæt])
	$\mapsto$	/kæt/	/kætz/	$\mapsto$	[kæts]	(· , [kæts])
·	$\mapsto$	/kætz/	/pig/	$\mapsto$	[pig]	(, [pig])
	$\mapsto$	/pig/	/pigz/	$\mapsto$	[pigz]	(· , [pigz])
·	$\mapsto$	/kaũ/	/kaũz/	$\mapsto$	[kaũz]	(, [kaũ])
	$\mapsto$	/kaũz/	/wʌgz/	$\mapsto$	[wʌgz]	(· , [kaũz])
·	$\mapsto$	/wʌks/				
$\vdots$		$\vdots$	$\vdots$			



Finite-state machine for φ



Finite-state machine for λ



Finite-state machine for $\varphi \circ \lambda$

Input

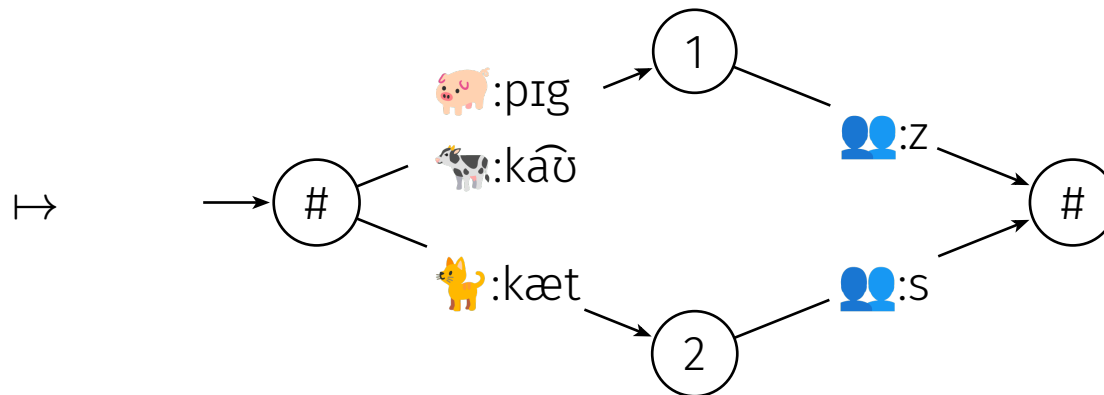
Finite sample of $\varphi \circ \lambda$

- | | |
|---|---------|
| ( , | [kæt]) |
| ( .  , | [kæts]) |
| ( , | [pɪg]) |
| ( .  , | [pɪgz]) |
| ( , | [kəʊ]) |
| ( .  , | [kəʊz]) |

Input

Finite sample of $\varphi \circ \lambda$

( ,	[kæt])
( .  ,	[kæts])
( ,	[pig])
( .  ,	[pigz])
( ,	[kɑ̃])
( .  ,	[kɑ̃z])



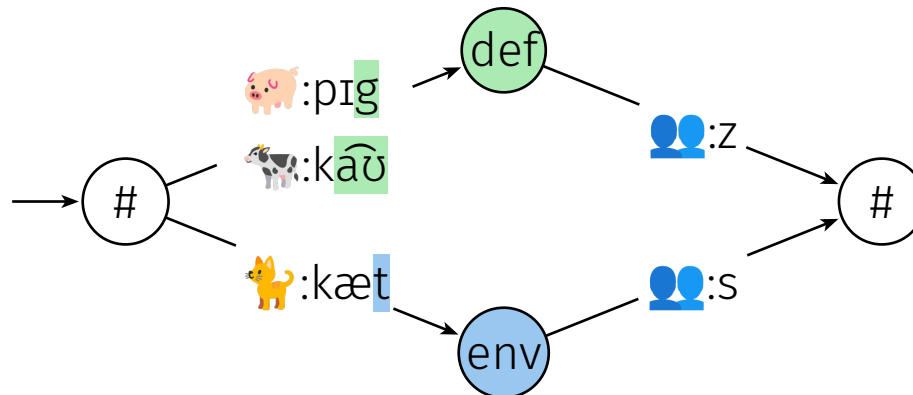
OSTIA-D (Oncina et al., 1993),(Oncina & Varò, 1996)

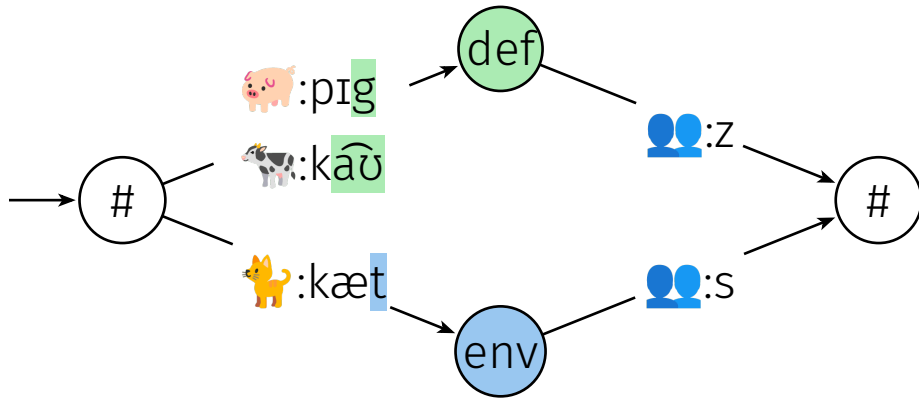
Input

Finite sample of $\varphi \circ \lambda$

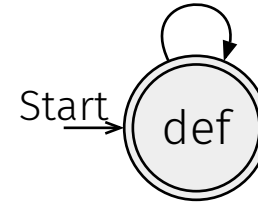
(🐱,	[kæt])
(🐱.👤,	[kæts])
(🐷,	[pig])
(🐷.👤,	[pigz])
(🐮,	[kɑ̃])
(🐮.👤,	[kɑ̃z])

$\mapsto$

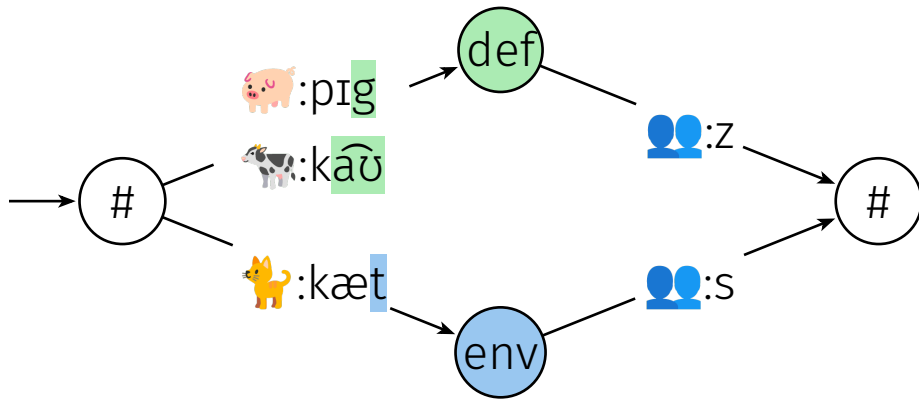




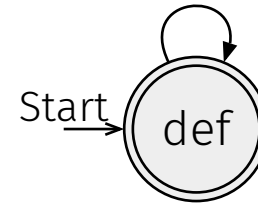
🐮:kæu, 🐷:pig, 🐱:kæt, 👤:z,...



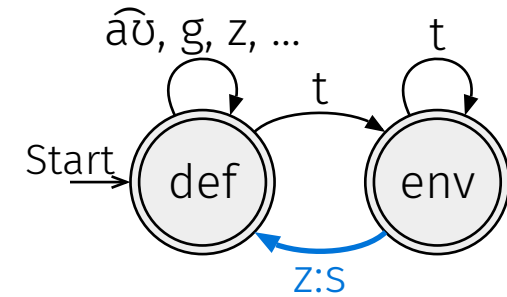
Finite-state machine for λ



🐮:kâũ, 🐷:pig, 🐱:kæt, 👤:z,...



Finite-state machine for λ



Finite-state machine for φ

(Chandlee & Jardine, 2025): Dataset from Johore Malay (Onn, 1976)

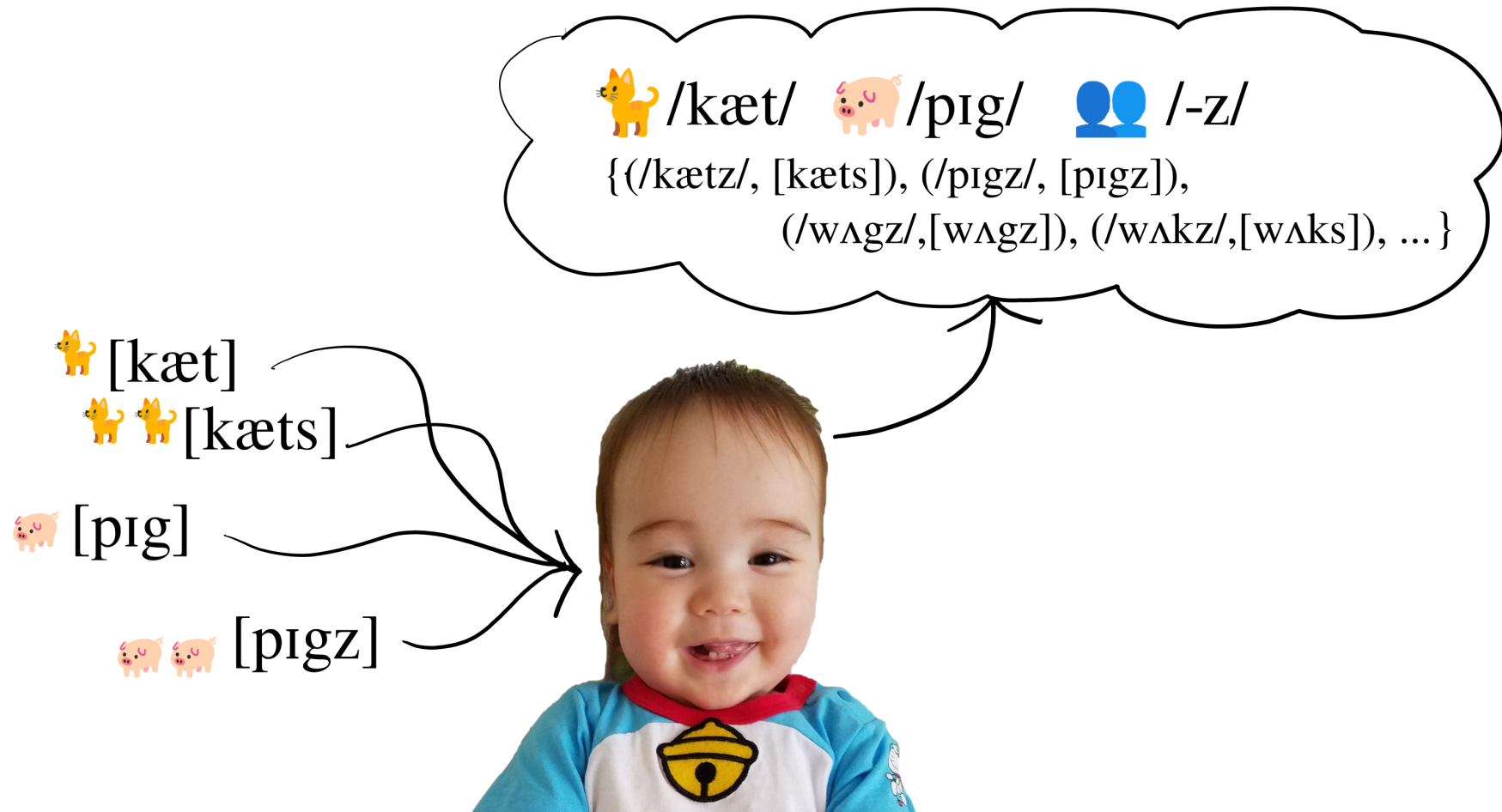
- Input: 14 stems × 5 affixes
- Output: Correct lexicon and grammar representing five processes
- Generalizes theoretical results of (Hua et al., 2021), (Hua & Jardine, 2021)
- Uses notion of **natural classes**

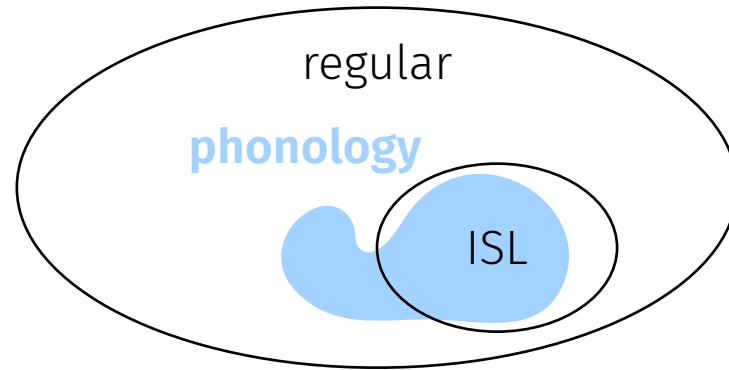
Other solutions

- Constraint-based grammars (Tesar 2014; Jarosz 2006, 2013; Wang and Hayes 2022)
- Minimum-description length (Rasin et al., 2020, 2021)
- Bayesian approaches (Barke et al., 2019; Ellis et al. 2022)
- All require some **restriction** on hypothesis space

Future work

- Interacting processes (Wang)
- Two-sided contexts (Anthony, Jeyaram, Kandepalli)
- Long-distance processes (Czarnecki, Franz)
- Abstract lexical entries (Franz)
- Non-concatenative morphology (Helfand)
- Testing on large corpus of datasets (Heinz et al.)
- ...





Thank you!