Evaluating Domain-General Learning of Parametric Stress Typology

Gaja Jarosz (University of Massachusetts Amherst)
Aleksei Nazarov (University of Toronto)

Society for Computation in Linguistics
January 6 2019

Overview

- Are learners for Principles & Parameters grammars different from OT learners?
  - Do they require access to the content of linguistic parameters (Pearl 2007, 2011)?
  - In other words, are they domain-specific (and must they be specified in UG)?

- Nazarov & Jarosz (2017): sophisticated inference in a domain-general learner may be sufficient
- Current work: typological test of this learner

Domain-generality vs. domain-specificity

Stress in OT vs. P&P

- All learners for **regular** stress patterns in OT are domain-general (have no specific knowledge of grammar)
  - Only refer to match/mismatch, settings/probabilities in the model, degrees of certainty/uncertainty
- However, for stress in Principles and Parameters (e.g., Dresher and Kaye 1990), domain-specific learners appear to be required (Pearl 2007, 2011)
  - Learners that refer to names of specific parameters, and to the content of specific parameters
  - Cannot be general cognitive strategy, must be at least partially specified in UG

Hypothesis

- Our hypothesis (see also Nazarov and Jarosz 2017): P&P word stress grammars can be learned with domain-general mechanisms (i.e., those without access to information on specific parameters)
  - Established domain-general learner (Yang 2002): not very effective (Pearl 2011, Nazarov and Jarosz 2017)
  - Propose new domain-general learner (Nazarov and Jarosz 2017) that has a better chance of dealing with the ambiguity in the learning problem

Previous results and overview

- Inspired by Tesar and Smolensky (2000), Nazarov and Jarosz (2017) tested their learner on a subset of stress systems possible in Dresher and Kaye (1990)
  - Nazarov and Jarosz’s learner highly successful, Yang’s learner learns only 1 stress system

- Here: our existing learner + 3 versions of Yang’s learner tested on ALL 280 stress systems possible in Dresher and Kaye (1990)

- EDPL successful on 96% of these stress systems

The learning challenge

Ambiguity in stress systems

- A single data point is often ambiguous between many different analyses

a. $\acute{\sigma}\ \sigma\ \sigma\ \sigma\ \sigma\ \sigma$

b. Trochee = On, MainStressLeft = On, **SilentSecondaryStress = On**

$(\underline{\acute{\sigma}}\ \sigma)\ (\underline{\sigma}\ \sigma)\ (\underline{\sigma}\ \sigma)$ (multiple trochaic feet are built over the word, but only the leftmost one, the head foot, receives stress)

c. Trochee = On, **Bounded = Off**

$(\underline{\acute{\sigma}}\ \sigma\ \sigma\ \sigma\ \sigma\ \sigma)$ (a single, unbounded trochaic foot is constructed over the word)

d. MainStressLeft = On, **QS = Off**, **NoLightFootHead = On**

$\acute{\sigma}\ \sigma\ \sigma\ \sigma\ \sigma\ \sigma$ (feet are constructed, then destroyed, since their heads are light; instead, stress comes on the leftmost element: the 1st syllable)

Credit/Blame problem

- This leads to a Credit/Blame problem (cf. Dresher and Kaye 1990)
  - success in correctly generating a data point with certain parameter settings makes it unclear which of these to *credit* for the victory
  - failure to correctly generate a data point with certain parameter settings makes it unclear which of these to *blame* for the defeat
- Leads to a potential stalemate for the learner

Approaches

- Proposed approaches to Credit/Blame problem:
  - Learning only from data points that provide **unambiguous evidence** for a particular parameter setting
    (Dresher and Kaye 1990, Fodor 1998, Pearl 2007)
  - Learning parameters in a **pre-specified order**
    (Dresher and Kaye 1990, Pearl 2007)
  - Adjusting each parameter's setting in proportion to the strength of support from each data point (use **statistical inference**)
    (Gould 2015, Nazarov and Jarosz 2017)

Domain-general learners

Domain-general proposals

- Yang (2002): Naïve Parameter Learner (NPL)
  - Maximally simple statistical learner for parameters
  - Shows good initial results for simple syntactic parameter setting problems
- Nazarov and Jarosz (2017): Expectation-Driven Parameter Learner (EDPL)
  - Based on the Expectation-Driven Learning algorithm for OT (Jarosz 2015)
  - Unifies hidden structure learning for OT and parameters

Domain-general proposals

- Share:
  - Probabilistic parameter grammar
  - Online linear update rule

- Differ in:
  - Use of statistical inference
  - Nature of the values input to the update rule (categorical or probabilistic)

Probabilistic parameter grammar

- Principles:
  - A parametrized word stress generator that assigns foot and PrWd structure

- Parameters:
  - The setting of each parameter is represented by a Bernoulli distribution

  - For each instance of generation, sample an On/Off value from each Bernoulli distribution

Probabilistic parameter grammar

- Parameters:
  - The setting of each parameter is represented by a Bernoulli distribution

$$G = \left\{ \begin{matrix} P(FootHead: L) = 0.6 \\ P(FootHead: R) = 0.4 \end{matrix}, \begin{matrix} P(Footing: L \rightarrow R) = 0.3 \\ P(Footing: R \rightarrow L) = 0.7 \end{matrix}, \dots \right\}$$

  - For each instance of generation, sample an On/Off value from each Bernoulli distribution

/tatama/	*sample FootHead: L and Footing: R → L*	*ta(ˈta.ma)*
/tatama/	*sample FootHead: L and Footing: L → R*	*(ˈta.ta)ma*

Yang (2002)

- Naïve Parameter Learner (NPL) assigns a Reward value of 1 or 0 for each parameter setting

- For each data point, generates stress pattern

  - If **match** (observed = predicted), **R = 1** for all parameter settings utilized
    (R = 0 for all other settings)

  - If **mismatch** (observed ≠ predicted), **R = 0** for all parameter settings utilized
    (R = 1 for all other settings)

Update rule

- Linear Reward-Penalty Rule (Bush and Mosteller 1951) responds to each data point by adjusting parameter probabilities up or down:

$$\hat{P}(\psi_i \mid G_{t+1}) = \lambda \times R(\psi_i) + (1 - \lambda) \times P(\psi_i \mid G_t)$$

New probability

Reward value

Old probability

Learning rate

Nazarov and Jarosz (2017)

- Each parameter setting's reward (R) is a probability
  - R($\boldsymbol{\psi}_i$) = p($\boldsymbol{\psi}_i$|data point) – How useful is this parameter setting for successfully analyzing this data point?
  - Approximates E-step in Expectation Maximization
- Easily computed through Bayesian reformulation (Jarosz 2015):

$$R(\psi_i) = p(\psi_i | data\,point) = \frac{p(data\,point|\psi_i) * \; p(\psi_i)_{old}}{p(data\,point)}$$

(by Bayes' Rule)

Nazarov and Jarosz (2017)

- Each parameter setting's reward (R) is a probability
  - R(ψ_i) = p(ψ_i|data point) – How useful is this parameter setting for successfully analyzing this data point?
  - Approximates E-step in Expectation Maximization
- Easily computed through Bayesian reformulation (Jarosz 2015):

$$R(\psi_i) = p(\psi_i | data\, point) = \frac{p(data\, point | \psi_i) * \; p(\psi_i)_{old}}{p(data\, point)}$$

Prob. of choosing this parameter setting (look up in current grammar)

Nazarov and Jarosz (2017)

- Each parameter setting's reward (R) is a probability
  - $R(\psi_i) = p(\psi_i | \text{data point})$ – How useful is this parameter setting for successfully analyzing this data point?
  - Approximates E-step in Expectation Maximization
- Easily computed through Bayesian reformulation (Jarosz 2015):

$$R(\psi_i) = p(\psi_i | data\,point) = \frac{p(data\,point | \psi_i) * \; p(\psi_i)_{old}}{p(data\,point)}$$

Estimated by sampling
(see below)

Nazarov and Jarosz (2017)

- Each parameter setting's reward (R) is a probability
  - $R(\psi_i) = p(\psi_i|\text{data point})$ – How useful is this parameter setting for successfully analyzing this data point?
  - Approximates E-step in Expectation Maximization
- Easily computed through Bayesian reformulation (Jarosz 2015):

$$R(\psi_i) = p(\psi_i|data\ point) = \frac{p(data\ point|\psi_i) *\ p(\psi_i)_{old}}{p(data\ point)}$$

Derived from both previously mentioned quantities

Estimating p(data point$|\psi_i$)

- **p(data point$|\psi_i$)** estimated as proportion of matches out of a sample of productions, assuming the setting ψ_i (constrained sampling)
  - Temporarily set p(ψ_i) to 1 in current grammar (e.g., set FootHead to L, keep all other parameters probabilistic)
  - Produce data point r times and assess match/mismatch (we chose *r* = 50)
  - $p(data\ point|\psi_i) \approx \frac{number\ of\ matches}{r}$

Comparison

- Reward computation for NPL: based on a single guess as to hidden structure
- Reward computation for EDPL: based on statistical inference of hidden structure
- Computation time for both learners: linear
  - NPL: Number of match/mismatch trials per data point = 1
  - EDPL: Number of match/mismatch trials per data point = # of parameters x # of settings x sample size (in our case: 1100)

Simulations + results

Simulation setup

- All 280 unique stress systems possible in Dresher and Kaye (1990) generated
  - As in Nazarov and Jarosz (2017), presented on 1080 words of 3 to 6 syllables
    - Every possible combination of CV, CVC, CVV syllables represented exactly once
  - All words equally likely to occur ("parent" samples from uniform distribution)

- For each of these 280 stress systems, NPL (batch size = 0, 5, 10) and EDPL run 10 times
  - NPL: maximum of 10,000,000 iterations
  - EDPL: maximum of 100,000 iterations

Random baseline

- In addition, random baseline learner run 10 times for each stress system ("how fast can you learn a system by just guessing"):
  - Choose a random parameter grammar (non-probabilistic), and generate each incoming data point's stress with this grammar
  - At each mismatch, choose another random grammar
  - Convergence: when no more mismatches are detected
    - Convergence guaranteed: number of grammars is finite

Results

	EDPL	NPL, no batch	NPL, batch = 5	NPL, batch = 10	Random baseline
# of runs that converge (% of 2800)	2644 (94.4%)	21 (0.8%)	176 (6.3%)	148 (5.3%)	
# of stress systems that converge at ≥1 run (% of 280)	268 (95.7%)	3 (1.1%)	25 (8.9%)	24 (8.6%)	
# of stress systems that converge at all 10 runs (% of 280)	255 (91.1%)	2 (0.7%)	10 (3.6%)	12 (4.3%)	
Median # of iterations/data points till convergence (range)	200 (100–15,700)	200,000 (4,400–9,999,900)	70,000 (400–9,000,000)	4,100 (700–9,999,900)	700 (100-30,000)

Results

> 90%	EDPL	NPL, no batch	NPL, batch = 5	NPL, batch = 10	Random baseline
# of runs that converge (% of 2800)	2644 (94.4%)	21 (0.8%)	176 (6.3%)	148 (5.3%)	
# of stress systems that converge at ≥1 run (% of 280)	268 (95.7%)	3 (1.1%)	25 (8.9%)	24 (8.6%)	
# of stress systems that converge at all 10 runs (% of 280)	255 (91.1%)	2 (0.7%)	10 (3.6%)	12 (4.3%)	
Median # of iterations/data points till convergence (range)	200 (100–15,700)	200,000 (4,400–9,999,900)	70,000 (400–9,000,000)	4,100 (700–9,999,900)	700 (100-30,000)

Faster than baseline

Results

> 90%	EDPL	NPL, no batch < 10%	NPL, batch = 5	NPL, batch = 10	Random baseline
# of runs that converge (% of 2800)	2644 (94.4%)	21 (0.8%)	176 (6.3%)	148 (5.3%)	
# of stress systems that converge at ≥1 run (% of 280)	268 (95.7%)	3 (1.1%)	25 (8.9%)	24 (8.6%)	
# of stress systems that converge at all 10 runs (% of 280)	255 (91.1%)	2 (0.7%)	10 (3.6%)	12 (4.3%)	
Median # of iterations/data points till convergence (range)	200 (100–15,700)	200,000 (4,400–9,999,900)	70,000 (400–9,000,000)	4,100 (700–9,999,900)	700 (100-30,000)

Faster than baseline

Slower than baseline

Relation to typology

Stress patterns (not) learned

- NPL (batch = 0) only learns initial/final stress

- NPL (batch = 5, 10) learns:
  - initial/final stress, penult/peninitial stress
  - a selective range of quantity-sensitive patterns

- EDPL learns the overwhelming majority of all patterns
  - 12 stress systems never learned
  - 13 stress systems learned, but not at all 10 runs

Never learned by EDPL (or NPL)

- 10 of the systems where location of stress **depends on (silent) feet** built throughout the word

(σ σ) (σ σ) (ˈσ σ) *penult stress in even-syllable words*
(σ σ) (σ σ) (ˈσ σ) σ *antepenult stress in odd-syllable words*

- 2 systems where only **long vowels attract stress** to the word edge

ma ta ˈ**ka** tan	ma taa ˈ**ka** ta	ma ta ˈ**kaa** ta
ma ta ka ˈ**taa**	ma taa ka ˈ**taa**	ma ta kaa ˈ**taa**

- <u>None of these systems are learned by the NPL varieties, either</u>

Dependence on silent feet

- 10 systems where the location of stress depends on (silent) feet built throughout the word

(σ σ) (σ σ) (ˈσ σ)	*penult stress in even-syllable words*
(σ σ) (σ σ) (ˈσ σ) σ	*antepenult stress in odd-syllable words*

  - 1 of these resembles Cairene Arabic (McCarthy 1979)
    - But: silent foot analysis disputed by Buell (1996), (Becker 2017)
    - Other 9 not-learned stress systems unattested

  - Negev Bedouin Arabic and Cyrenaican Arabic (StressTyp2) also have a silent foot system, but one that IS learned by the EDPL

Stress VV at word edge

- 2 systems where only long vowels attract stress to the word edge

ma ta ˈka tan	ma taa ˈka ta	ma ta ˈkaa ta
ma ta ka ˈtaa	ma taa ka ˈtaa	ma ta kaa ˈtaa

- Both systems are attested (StressTyp2)
- Corresponding patterns where both VV and VC are heavy: learned
    - Suggests that the problem is that VV syllables are in the **minority** in our artificial languages

Conclusion

Domain-general vs. domain-specific

- Are domain-specific mechanisms necessary for stress parameter setting
  - Yang’s (2002) NPL (domain-general learner) deemed insufficient (Pearl 2011)
  - However, domain-specific mechanisms increase the amount and kind of information stored in UG

- Nazarov and Jarosz (2017) propose alternative domain-general learner (EDPL)
  - Has stronger statistical inference component
  - Still computable in linear time

Typological test

- Sufficiency of domain-general learning:
  - 3 attested stress systems not learned by EDPL may have alternative explanation
  - For the rest, domain-specific learning mechanisms only serve to learn unattested languages

- Future work/in progress:
  - Other parameter systems (syntactic parameters: in progress)
  - Vary implementational details of NPL and EDPL (sample size, …)
  - What predicts learnability under NPL and EDPL?

Thank you!

References

Becker, Michael. 2017. Affix-specificity makes stress learnable. 91st meeting of the Linguistic Society of America, Austin, TX.

Berwick, Robert C., and Partha Niyogi. 1996. Learning from Triggers. *Linguistic Inquiry* 27(4): 605-622.

Bush, Robert, and Frederick Mosteller. 1951. A mathematical model for simple learning. *Psychological Review* 58, 313–323.

Buell, Leston. 1996. A Footless, Constraint-Based Analysis of Stress in Cairene Arabic. Ms., UCLA.

Chomsky, Noam. 1981. *Lectures on government and binding*. Dordrecht: Foris.

Dresher, B. Elan. 1999. “Charting the Learning Path: Cues to Parameter Setting.” *Linguistic Inquiry* 30, 27-67.

Dresher, B. Elan, and Jonathan D. Kaye. 1990. A computational learning model for metrical phonology. *Cognition* 34: 137-195.

Fodor, Janet D. 1998. Parsing to Learn. *Journal of Psycholinguistic Research* 27(3), 339-374.

References

Gibson, Edward, and Kenneth Wexler. 1994. Triggers. *Linguistic Inquiry* 25(3): 407-454.

Goedemans, R.W., Jeffrey Heinz, and Harry van der Hulst. 2015. *StressTyp2, version 1*. Web download archive. http://st2.ullet.net. (=**StressTyp2**)

Gould, Isaac. *Syntactic Learning from Ambiguous Evidence: Errors and End-States*. PhD dissertation, Massachusetts Institute of Technology.

Jarosz, Gaja. 2015. *Expectation Driven Learning of Phonology*.

Lightfoot, D. 1999. *The Development of Language: Acquisition, Change, and Evolution*. Oxford: Blackwell.

Pater, Joe. 2000. Non-Uniformity in English Secondary Stress: The Role of Ranked and Lexically Specific Constraints. *Phonology* 17(2): 237–74.

Pearl, Lisa. 2007. *Necessary Bias in Natural Langauge Learning*. Doctoral dissertation, University of Maryland.

References

Pearl, Lisa. 2008. Putting the emphasis on unambiguous: The feasibility of data filtering for learning English metrical phonology. In Harvey Chan, Heather Jacob & Enkeleida Kapia (eds.), *Proceedings of the 32nd annual Boston University Conference on Child Language Development (BUCLD 32)*, 390–401. Somerville, MA: Cascadilla Press.

Pearl, Lisa. 2009. *Acquiring complex linguistic systems from natural language data: What selective learning biases can do*. Ms. University of California, Irvine.

Pearl, Lisa. 2011. When Unbiased Probabilistic Learning is Not Enough: Acquiring a Parametric System of Metrical Phonology. *Language Acquisition* 18(2): 87-120.

Sakas, William G., and Fodor, Janet D. 2001. The structural triggers learner. In Stefano Bertolo (ed.), *Language Acquisition and Learnability*, 172-233. Cambridge University Press, Cambridge, UK.

Yang, Charles. 2002. *Knowledge and Learning in Natural Language*. Oxford: Oxford University Press.

Appendix

Scope of application

- **Domain-specific** learning mechanisms are picky with respect to the nature of the systems they can optimize (specific to a domain)
  - Require information on the content of certain elements in the system
  - Example: Markedness-over-Faithfulness bias

- **Domain-general** learning mechanisms treat the system they optimize as a black box, and can be applied to data of any nature
  - Require only access to settings/probabilities in the system and feedback about their success
  - Example: Gradual Learning Algorithm

Yang (2002)

- Naïve Parameter Learner (NPL) with **batch** > 0 (Pearl 2011, p.c.) assigns a Reward value of 1 or 0 for each parameter setting

- For each data point, generates stress pattern
  - If **match** (observed = predicted), **counter =+ 1** for all parameter settings utilized

  - If **mismatch** (observed = predicted), **counter =- 1** for all parameter settings utilized

- Once a parameter's counter variable reaches **batch**, update with R = 1 for the default value
- Once a parameter's counter variable reaches **–batch**, update with R = 0 for default value

Not always learned: antepenult

- 13 stress systems not always learned by EDPL
  - 10 systems where stress depends on feet built throughout the word (like never learned systems): all unattested
  - 3 systems of weight-insensitive stress (see appendix)

  - Only attested one: antepenultimate stress with no secondary stress (15 lects in StressTyp2)

σ ˈσ σ σ σ σ ˈσ σ σ σ σ σ ˈσ σ σ σ σ σ σ ˈσ σ σ

  - Learned at 9 out of 10 iterations (not learned at all by NPL)

Ambiguity in stress systems

- Some data points are completely uninformative for certain parameters:

a.	(CV.CV̀)(CV.CV́)	QS = Off, HeavyVC = On/Off?	(CVV.CV̀)(CV.CV́C)
		QS = On, HeavyVC = Off?	(CV̀V)(CV.CV́)CVC
		QS = On, HeavyVC = On?	(CV̀V)(CV.CV̀)(CV́C)
b.	(σ σ̀) (σ σ́)	L-to-R = On?	(σ σ̀) (σ σ́) σ
		L-to-R = Off?	σ (σ σ̀) (σ σ́)

Ambiguity in stress systems

- Opposite settings of parameters can yield the same outcome:

a. $\sigma\ \sigma\ \sigma\ \sigma\ \acute{\sigma}\ \sigma$

b. **Trochee = On**, Bounded = On, MainStrLeft = Off, SilentSecStr = On, **XMetrical = Off**

$(\underline{\sigma}\ \sigma)\ (\underline{\sigma}\ \sigma)\ (\underline{\acute{\sigma}}\ \sigma)$ (multiple trochaic feet are built over the word, no extrametricality, but only the rightmost one, the head foot, receives stress)

c. **Trochee = Off**, MainStrLeft = Off, L-to-R = Off, **XMetrical = On,** XMetricalLeft = Off

$\sigma\ (\sigma\ \underline{\sigma})\ (\sigma\ \underline{\acute{\sigma}})$ <σ> (iambic feet are built R-to-L over the word minus the rightmost syllable, but only the rightmost (head) foot receives stress)

$(\sigma\ \sigma\ \sigma\ \sigma\ \underline{\acute{\sigma}})$ <σ>

Ambiguity in stress systems

- There may be logical dependencies between parameter settings:

a. $\sigma\ \grave{\sigma}\ \sigma\ \acute{\sigma}\ \sigma$

b. **Trochee** = **Off**, **L-to-R** = **On**, XMetrical = Off, Defooting = On
$(\sigma\ \underline{\grave{\sigma}})\ (\sigma\ \underline{\acute{\sigma}})\ \sigma$ (trochaic feet built from left to right, no degenerate feet)

c. **Trochee** = **On**, **L-to-R** = **Off**, XMetrical = Off, Defooting = On
$\sigma\ (\underline{\grave{\sigma}}\ \sigma)\ (\underline{\acute{\sigma}}\ \sigma)$ (iambic feet built from right to left, no degenerate feet)

d. **Trochee** = **On**, (L-to-R = On), **XMetrical** = **On**, XMetricalLeft = On
$\langle\sigma\rangle\ (\underline{\grave{\sigma}}\ \sigma)\ (\underline{\acute{\sigma}}\ \sigma)$ (trochaic feet built in either direction over the word minus its leftmost syllable)

Dresher and Kaye (1990)

- 11 parameters:

  (parameter 10 formulated according to Dresher 1999)

1. The word-tree is strong on [Left/Right] *(henceforth MainStressLeft = On/Off)*
2. Feet are [Binary/Unbounded] *(henceforth Bounded = On/Off)*
3. Feet are built from the [Left/Right] *(henceforth L-to-R = On/Off)*
4. Feet are strong on the [Left/Right] *(henceforth Trochee = On/Off)*
5. Feet are quantity sensitive (QS) [Yes/No] *(henceforth QS = On/Off)*
6. Feet are QS to the [Rime/Nucleus] *(henceforth HeavyVC = On/Off)*
7. A strong branch of a foot must itself branch [No/Yes]

   (henceforth NoLightFootHead = On/Off)
8. There is an extrametrical syllable [No/Yes] *(henceforth XMetrical = On/Off)*
9. It is extrametrical on the [Left/Right] *(henceforth XMetricalLeft = On/Off)*
10. Feet consisting of a single light syllable are removed [No/Yes][1]

    (henceforth Defooting = On/Off)
11. Feet are noniterative [No/Yes] *(henceforth SilentSecondaryStress = On/Off)*

Examples of cues

- Quantity-sensitivity:
  - Default: off; set to on if there is a pair of words of equal length with different stress patterns (This "cheats" on the one-data-point-at-a-time criterion)
- Foot Boundedness:
  - Default: off; set to on if the corpus contains a non-peripheral stressed Light syllable (with periphery modified by extrametricality)
- Footing Direction (L-to-R or R-to-L) and Foot Headedness (Left or Right)
  - Not [L-to-R and Left-headed] if corpus has stressed L after $H(LL)_0$ or $\#(XL)_0X$
  - Not [L-to-R and Right-headed] if corpus has stressed L after $H(LL)_0$ or $\#(LX)_0$
  - Not [R-to-L and Left-headed] if corpus has stressed L before $(LL)_0H$ or $(XL)_0\#$
  - Not [R-to-L and Right-headed] if corpus has stressed L before $(LL)_0H$ or $X(LX)_0\#$

Nazarov and Jarosz (2017)

- Each parameter setting's reward (R) is a probability
  - $R(\psi_i) = p(\psi_i | \text{data point})$ – How useful is this parameter setting for successfully analyzing this data point?
  - Approximates E-step in Expectation Maximization
- Easily computed through Bayesian reformulation (Jarosz 2015):

$$p(\boldsymbol{data\,point}) = \begin{array}{c} p(data\,point|\psi_i) * \ p(\psi_i)_{old} \\ + \\ p(data\,point|\neg\psi_i) * \ p(\neg\psi_i)_{old} \end{array}$$

Probability of picking opposite value of the same parameter

Nazarov and Jarosz (2017)

- Tested NPL (batch = 0) and EDPL on 23 stress systems possible in Dresher and Kaye (1990)
  - Omitted Defooting parameter
  - Learners run for 1,000,000 iterations, 10 runs per stress system

- Results:
  - NPL (batch 0): converged in 4.3% of runs/stress systems
    - mean number of iterations: 89,400
  - EDPL: converged in 96.1% of runs, 95.7% of stress systems (100% of stress systems had at least one convergent run)
    - mean number of iterations: 200

Nazarov and Jarosz (2017)

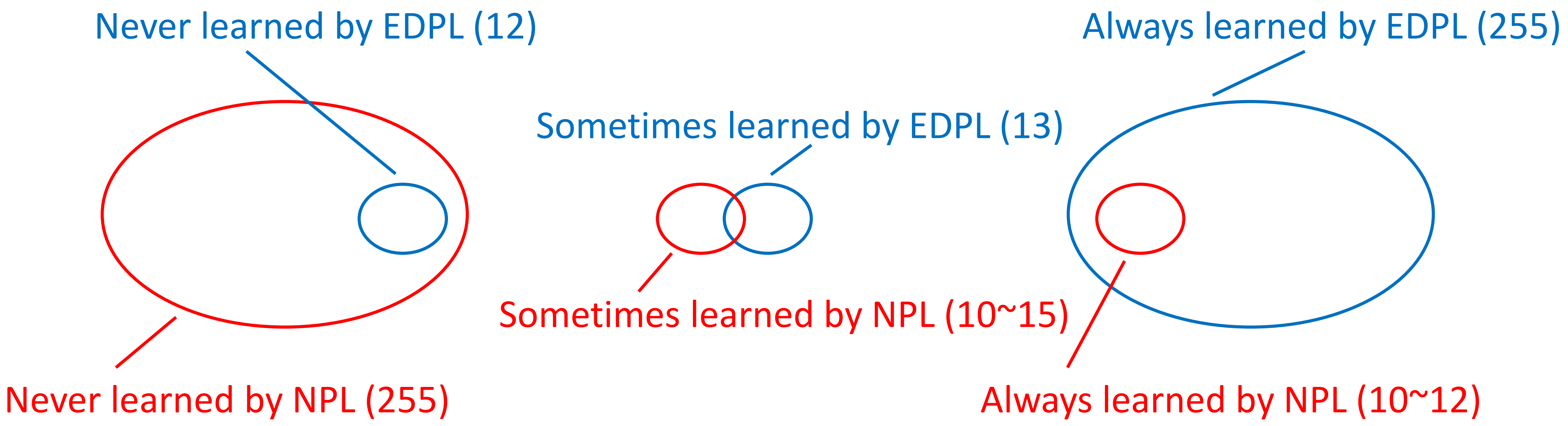
- Results:
  - NPL only learned the language with edgemost stress, which is compatible with the greatest number of settings of all parameters (SAPs)

    ˈσ σ σ ˈσ σ σ σ ˈσ σ σ σ σ ˈσ σ σ σ σ

  - EDPL learned all but a stress pattern with L-to-R trochees and Left extrametricality, which shares all its forms with some other stress pattern

QI L-to-R languages	3 syllables	4 syllables	5 syllables	6 syllables
a. Trochees, Ext. L	<σ> (ˈσ σ)	<σ> (ˈσ σ)(ˌσ)	<σ> (ˈσ σ)(ˌσ σ)	<σ> (ˈσ σ)(ˌσ σ)(ˌσ)
b. Iambs, Ext. R	(σ ˈσ) <σ>	(σ ˈσ)(ˌσ) <σ>	(σ ˈσ)(σ ˌσ) <σ>	(σ ˈσ)(σ ˌσ)(ˌσ) <σ>
c. Iambs, No Ext.	(σ ˈσ)(ˌσ)	(σ ˈσ)(σ ˌσ)	(σ ˈσ)(σ ˌσ)(ˌσ)	(σ ˈσ)(σ ˌσ)(σ ˌσ)

Number of stress patterns (not) learned



QI systems not always learned by EDPL

- Post-peninitial stress with alternating secondary stress (unattested)

σ σ ˈσ σ   σ σ ˈσ σ σ   σ σ ˈσ σ ˌσ σ   σ σ ˈσ σ ˌσ σ σ

- Penultimate stress with alternating secondary stress EXCEPT on the first syllable (unattested)

σ σ ˈσ σ   σ ˌσ σ ˈσ σ   σ σ ˌσ σ ˈσ σ   σ ˌσ σ ˌσ σ ˈσ σ

- Antepenultimate stress with no secondary stress (attested)

σ ˈσ σ σ   σ σ ˈσ σ σ   σ σ σ ˈσ σ σ   σ σ σ σ ˈσ σ σ

Statistics on language attestedness

- StressTyp2 contains 699 languages (and 699+ lects)
  - 137 (about 20%) of these have regular stress but cannot be analyzed with Dresher & Kaye's system
- Of 280 stress systems in Dresher and Kaye's system, 46 are attested in StressTyp2
  - 43/46 sometimes learned by EDPL
  - 42/46 always learned by EDPL
  - NPL/batch=0: 3/46 sometimes learned
  - NPL/batch=5 or 10: 9/46 sometimes learned

Predictors of success

- Number of grammars compatible with stress pattern
  ($\propto$ likelihood of *guessing* a correct grammar)

SAPs per stress system	330	122	32	16	9	8	6	4	3	2	1
Number of stress systems	2	2	8	16	4	4	18	40	8	94	84

- Somers' D rank correlation between individual runs' performance and grammars per stress system:

  D = 0.998 for NPL with no batch;

  D = 0.813 for NPL with batch = 5;

  D = 0.865 for NPL with batch = 10

  D = 0.045 for EDPL!

Typological test

- Both learners tested on a 280 possible stress systems in Dresher and Kaye (1990):
  - EDPL performs accurately on 95% of runs and >90% of stress systems
  - NPL performs accurately on <10% of runs/stress systems
- NPL’s performance can be somewhat improved by using Yang’s (2002), Pearl’s (2011) batch mechanism
  - However, the stress systems learned still have a strong tendency to be compatible with many grammars (higher chance of “guessing” the right grammar)

Domain-specific learner

Dresher and Kaye (1990)

- Categorical learner (no probabilities)
  - Each parameter has a **cue**: a configuration in the data that uniquely signifies a particular setting of the parameter
    - Once you see a cue for parameter setting ψ in a data point, add ψ to the grammar

      e.g., if you see a cue for Extrametricality = Off (stress on the first syllable when stress on the last syllable has been observed in another data point, or vice versa), the grammar now contains Extrametricality = Off

    - This allows the learner to only consider unambiguous evidence

  - There is also a fixed **order** in which cues are considered:
    - Look out for cues for Parameter 1 first, then those for Parameter 2, then those for Parameter 3, etc., until all parameters are set

Dresher and Kaye (1990)

1. Look for cue for parameter 1

~~data point 1~~ ~~data point 2~~ ~~data point 3~~ data point 4

matches cue for parameter 1 = off

2. Parameter 1 set to Off
3. Look for cue for parameter 2

~~data point 5~~ ~~data point 6~~ ~~data point 7~~ …..

(Data point 1, Data point 2, etc. are tokens –
the learner learns from one data point token at a time, simulating real-time acquisition)