No Free Lunch in Linguistics or Machine Learning: Response to Pater

Rawski and Heinz

Introduction / Background

- Pater (2019) claims that ‘generative linguistics is unlikely to fulfill its promise of accounting for language learning if it continues to maintain its distance from neural and statistical approaches to learning’.
- Pater’s answer to ‘how systems that adequately represent linguistic knowledge can be learned’ is neural networks
  - Disregards existing research that answers this question
- Pater conflates ignorance of bias with absence of bias
- Machine learning has morphed more into engineering/alchemy than science

No Free Lunch

Mitchell, 2017:

‘When we consider it carefully, it is clear that no system—computer program or human—has any basis to reliably classify new examples that go beyond those it has already seen during training, unless that system has some additional prior knowledge or assumptions that go beyond the training examples. In short, there is no free lunch—no way to generalize beyond the specific training examples, unless the learner commits to some additional assumptions.’

Bias is a two-way street

- Poverty of the stimulus (aka Data sparsity problem) and Zipf's law–most things are rare and therefore not in the training data
  - More data exacerbates the problem, rather than easing it
- The intrinsic biases of neural network models are almost completely unknown, and are not necessarily weak or impoverished
  - Ignorance of bias does not equal absence of bias
- Additionally, the utility of a weak-bias learning method for NLP tasks does not guarantee explanatory adequacy.

Pater v. Science

- Pater sees neural networks as a counterpoint to UG because RNNs lack parameters, constraints, and structure in general
- ‘No free lunch’ principle: learning requires a structured hypothesis space, and the structural properties matter.
    - the core of any generative theory of linguistic competence
- Computational Learning Theory seeks to answer:
    - what successful learning means in different scenarios, and
    - what resources successful learning requires in terms of computation and number of training examples
- Pater doesn’t mention the link between neural networks and formal languages
- Complete adoption of neural networks would be premature because it leaves us without a ‘precise characterization’ of language acquisition and learnability.

Computational Theories of Language

- Chomsky Hierarchy–divides all logically possible patterns into nested regions of complexity
- All grammars are associated with a function

FUNCTION	DESCRIPTION	LINGUISTIC CORRELATE
$f: \Sigma^* \rightarrow \{0, 1\}$	Binary classification	well-formedness
$f: \Sigma^* \rightarrow [0, 1]$	Maps strings to real values	gradient well-formedness
$f: \Sigma^* \rightarrow \Delta^*$	Maps strings to strings	single-valued transformation
$f: \Sigma^* \rightarrow \wp(\Delta^*)$	Maps strings to sets of strings	multi-valued transformation

- Pater argues we should move toward statistical generalization in grammars and learning
  - However, the addition of probabilities to a grammar formalism does not increase its expressivity

Learners as Functions to Grammars

- Learners = functions from experience to grammars
- How do we define successful learning? ‘’the circumstances under which these hypotheses stabilize to an accurate representation of the environment from which the evidence is drawn.’’ (Osherson et al, 1986)
- Grammatical inference–biases are analytically transparent

Typological Consequences of Learning Theories

3 types of learning strategies:

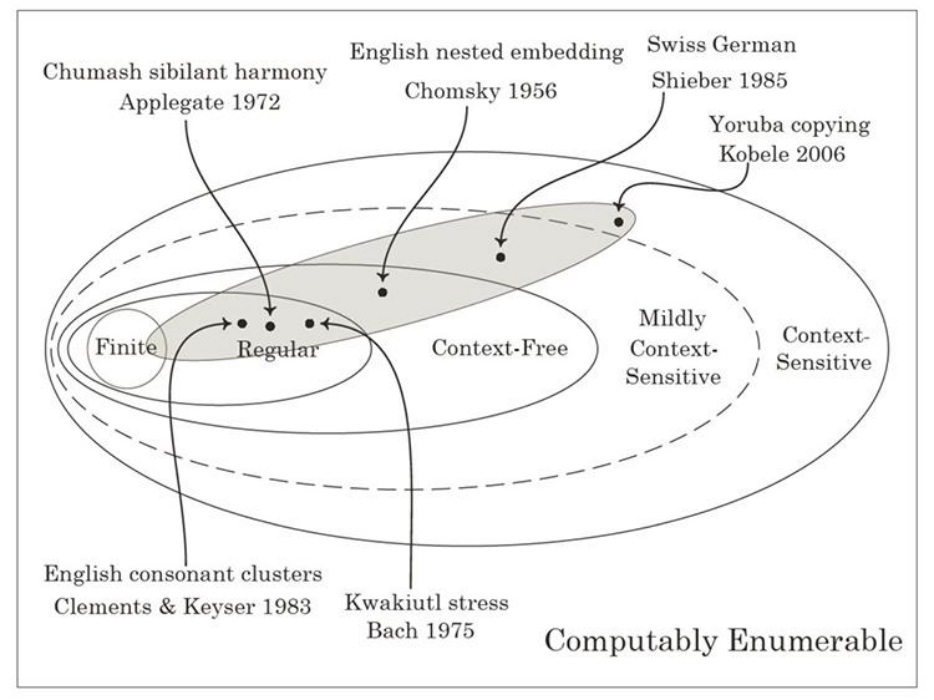
1. Learner assumes no structural variations.
2. Learner assumes one unified learning mechanism.
3. Learner has a modular approach that ties the learning strategy to the grammars being learned.

Learning Type 1: Learner assumes no structural variations

- Chater and Vitányi (2007) used minimum description length principles to prove that in a particular learning setting, any computably enumerable distribution can be learned from positive evidence.
- Possible in principle, infeasible in practice.
- Predicts that any pattern is learnable with enough data
    - Generalizations in real languages are not arbitrary and are much more restricted than being computable.
    - This same problematic assumption is often made about neural networks

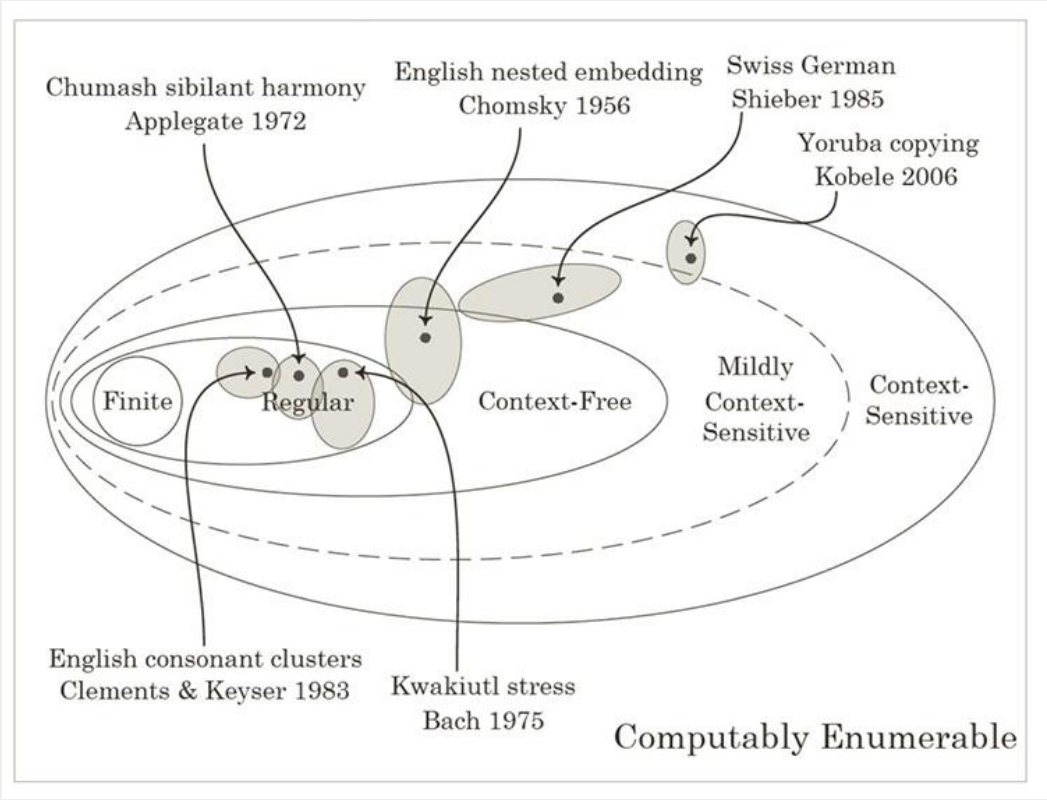
Learning Type 2: Learner Assumes 1 Unified Learning Mechanism

- Clark (2010)
- Overgeneralizes (“predicts that the sorts of dependencies seen in syntax at the sentence level should also be seen at the word level, in phonology and morphology”)



Learning Type 3: modular approach that adjusts the learning strategy to each particular grammar

- Distinct hypothesis spaces for each of the distinct subclasses of grammars that characterize the patterns found in natural language



Grammatical Inference and Neural Networks

- Comparisons of neural networks and various grammatical inference algorithms (Avcu et al, 2017; Weiss, Goldberg, Yahav, 2018) found that neural networks were…
  - Less simple,
  - Slower,
  - Made more mistakes
    - Even RNNs that trained with 100% accuracy!

Key Takeaways

- Learning types have concrete typological consequences on the end result
- Neural networks are seen as a black box/alchemy, so we don’t know their biases–this means that their use comes with risking inaccurate results
- Learner bias cannot be avoided, so we should strive to make all biases transparent
- Pater’s presentation of neural networks is too easy–we must pay attention to the hard problems