The Learnability of Model-Theoretic Interpretation Functions in Artificial Neural Networks

Adrian Brasoveanu (UC Santa Cruz)
Jakub Dotlačil (Utrecht University)

CPL 2025

The Task: Learning Sentence → Truth Conditions

Question: Can neural networks learn model-theoretic interpretation functions?

Setup (following Frank et al. 2009):

- **Microworld**: 3 people, 3 games, 3 toys, 4 locations → 44 atomic propositions
- **Situation vectors**: Encode event co-occurrence structure (details at poster)
- **Training**: Map sentences to target vectors; measure comprehension

The systematicity question: Does the learned interpretation function **generalize** to novel sentences not seen during training?

Complementary Train/Test Splits

What's held out in Split 1 is trained in Split 2 (and vice versa)—ensuring results don't depend on which particular sentences are excluded (C=charlie, H=heidi, S=sophia).

Test Group	**Split 1 Held Out**	**Split 2 Held Out**	**Truth Cond in Train**
Word (easiest)	C+soccer	boy+football	yes
Sentence	C {beats/loses to} H ...	C {beats/loses to} S ...	yes
Complex Event	chess+outside ...	chess+inside ...	no
Basic (hardest)	C+doll ...	C+ball ...	no

Four test groups of increasing difficulty:

- **Word**: Novel word combinations (synonym substitution), but target truth cond seen in training
- **Sentence**: Novel person pairs in "beats"/"loses to"—can model learn argument alternations?
- **Complex**: Novel game+location conjunctions, target truth cond **not** seen in training (only truth conditions for individual conjuncts seen)
- **Basic**: Novel person+toy combinations (hardest), target truth cond **not** seen in training

Training: ∼6,500 consistent sentences per split

Evaluation: Described vs. Competing Events

Problem: High score for correct interpretation isn't enough, model might learn event *type* rather than specific event.

Example: "charlie beats heidi" $\Rightarrow$ win(charlie) $\wedge$ lose(heidi), but model might just learn "someone wins, someone loses"

Add **competing events** that match event type but contradict described event (Frank et al. 2009 hardcodes; we generalize notion of "competing" so that applicable to any sentence):

- **Described**: win(charlie) $\wedge$ lose(heidi) — should score **positive**
- **Competing**: win(heidi), win(sophia), lose(charlie), lose(sophia) — should score **negative**

Comprehension score (Frank et al. 2009): Normalized belief change

$$\text{Comprehension}(a|z) = \begin{cases} \dfrac{P(a|z) - P(a)}{1 - P(a)} & \text{if } P(a|z) > P(a) \\ \dfrac{P(a|z) - P(a)}{P(a)} & \text{otherwise} \end{cases}$$

Systematicity / OOT Generalization = Advantage = score(described) $-$ score(competing)
Positive advantage $\Rightarrow$ model correctly distinguishes described from competing

What We Vary: Architectures & Entity Vectors

Four architectures (capacity-matched at ≈66k parameters for no-entity condition):

Architecture	Type	Hidden	Layers	Params (no entity)
SRN	Recurrent	178	1	66,010
LSTM	Recurrent (gated)	80	1	66,950
Attention AbsPE	Transformer	48	2	65,670
Attention RoPE	Transformer	48	2	65,670

Entity vectors — our extension to Frank et al truth-conditional target vectors:

- Original: 150-dim targets (truth-conditional only)
- Our extension: 300-dim targets (150 truth + 150 entity information)
  Comprehension scores evaluated only on truth-conditional part

Scale: 4 architectures × 2 entity conditions × 2 splits × 5 seeds = **80 models**

Results at our poster!

Which architectures generalize best? How much do entity vectors help?

Zoom link: Adrian available during poster session; QR on poster also