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

Adrian Brasoveanu · UC Santa Cruz

Joint work with Jakub Dotlačil

UC Davis L^{3}Lab · May 18, 2026

Outline

The Central Question

Can neural networks learn to *interpret* language
in a way that **generalizes compositionally**?

Systematicity

Anyone who understands “the cat chased the mouse”
can also understand “the mouse chased the cat.”

We systematically extend learned interpretation abilities
to **out-of-training-sample** combinations of familiar elements.

(Fodor and Pylyshyn, 1988)

The Classical Argument

Fodor & Pylyshyn's challenge:

- Systematicity requires compositional symbolic representations
- Neural networks can exhibit it only by *covertly implementing* a symbol system

$\Rightarrow$ Neural nets = implementation vehicles, not explanations

The Classical Argument

Fodor & Pylyshyn's challenge:

- Systematicity requires compositional symbolic representations
- Neural networks can exhibit it only by *covertly implementing* a symbol system

$\Rightarrow$ Neural nets = implementation vehicles, not explanations

If that is right, connectionist models of language understanding are explanatorily vacuous.

The Classical Argument

Fodor & Pylyshyn's challenge:

- Systematicity requires compositional symbolic representations
- Neural networks can exhibit it only by *covertly implementing* a symbol system

$\Rightarrow$ Neural nets = implementation vehicles, not explanations

If that is right, connectionist models of language understanding are explanatorily vacuous.

Our question: how much systematicity can a neural interpretation function learn without implementing a symbol system (from structured world knowledge)?

(Fodor and Pylyshyn, 1988)

A Neural Alternative: Frank et al. (2009)

Frank et al. (2009) showed that:

- Simple Recurrent Networks (SRNs) can learn to map sentences to distributed truth-conditional representations
- These representations support *some* systematic generalization
- **Without** implementing symbolic structures

A Neural Alternative: Frank et al. (2009)

Frank et al. (2009) showed that:

- Simple Recurrent Networks (SRNs) can learn to map sentences to distributed truth-conditional representations
- These representations support *some* systematic generalization
- **Without** implementing symbolic structures

Key insight: systematicity can arise from *structure in the world* encoded in analogical situation vectors, not only from symbolic compositionality in mental representations.

A Neural Alternative: Frank et al. (2009)

Frank et al. (2009) showed that:

- Simple Recurrent Networks (SRNs) can learn to map sentences to distributed truth-conditional representations
- These representations support *some* systematic generalization
- **Without** implementing symbolic structures

Key insight: systematicity can arise from *structure in the world* encoded in analogical situation vectors, not only from symbolic compositionality in mental representations.

But: only SRNs, a limited test set ($\sim$80 sentences), and an older MATLAB/Prolog implementation.

(Frank et al., 2009)

Interpretation as Encoding

Core mapping

discrete sentence $\rightarrow$ neural encoder $\rightarrow$ meaning vector in $[0, 1]^{150}$

- Not language modeling, classification, or sequence generation
- **Encoding task**: map linguistic input to continuous truth-conditional representations
- Vector similarity reflects co-occurrence in possible worlds, not just text co-occurrence
- Targets come from **model-theoretic world structure**, not corpus-based distributional statistics

Gaps in Prior Work

1. **Implementation:** MATLAB, C, Prolog $\rightarrow$ limited accessibility & architectural exploration
2. **Sampling:** sequential proposition sampling underestimates soft-constraint correlations
3. **Architectures:** only SRNs evaluated—what about LSTMs, GRUs, Transformers?
4. **Representations:** only truth values (type t); but entities (type e) are a co-equal primitive in formal semantics
5. **Evaluation:** limited test set ($\sim$80–120 sentences)

(Frank et al., 2009; Venhuizen et al., 2019, 2022; Brasoveanu, 2026; Heim, 1982; Kamp, 1981)

Our Contributions

1. **Entity vectors**: extend representations to include entity-level information (type e)
   Training targets: 150-dim (truth-conditional only) or 300-dim ($+$ entities)
2. **Modern architectures**: LSTM, GRU, Transformer (AbsPE / RoPE) alongside SRN
   Capacity-matched at $\sim$66k parameters
3. **Principled competing events**: Z3 validation replaces hardcoded rules
4. **Extended tests**: $\sim$80 $\rightarrow$ $\sim$350 test sentences across 4 groups
5. **Two-dimensional difficulty**: results disaggregated by modifier complexity

140 trained models (7 architectures $\times$ $\pm$entity $\times$ 2 splits $\times$ 5 seeds)
Python / PyTorch reimplementation of evaluation infra publicly available: `https://github.com/abrsvn/neural_model_theoretic_interpretation_ACL_2026`

Preview of Key Findings

Finding 1: Systematicity difficulty has two dimensions

Test type (Word $>$ Sentence $>$ Complex $>$ Basic)
$\times$ modifier complexity (canonical $>$ +location $>$ +manner $>$ +both)

Preview of Key Findings

Finding 1: Systematicity difficulty has two dimensions

Test type (Word $>$ Sentence $>$ Complex $>$ Basic)
$\times$ modifier complexity (canonical $>$ +location $>$ +manner $>$ +both)

Finding 2: Gating in recurrent networks is critical

GRU/LSTM $\gg$ Transformers on the hardest test (Basic Event)
Ungated SRN does *not* show the same Basic-Event advantage

Preview of Key Findings

Finding 1: Systematicity difficulty has two dimensions

Test type (Word $>$ Sentence $>$ Complex $>$ Basic)
$\times$ modifier complexity (canonical $>$ +location $>$ +manner $>$ +both)

Finding 2: Gating in recurrent networks is critical

GRU/LSTM $\gg$ Transformers on the hardest test (Basic Event)
Ungated SRN does *not* show the same Basic-Event advantage

Finding 3: Entity information helps most where it matters

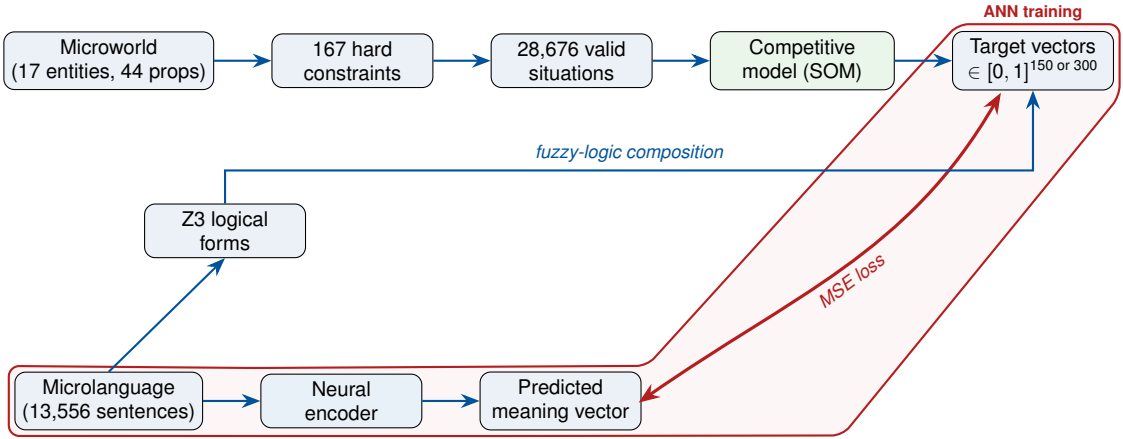
Largest benefit on Basic Event (novel entity combinations);
gated architectures benefit most

(Frank et al., 2009)

Outline

Framework Overview

Target vector = probabilistic representations of truth conditions the network is trained to output.



Microworld: 17 Entities in 6 Categories

Participants

Persons: charlie, heidi, sophia
Games: chess, hide&seek, soccer
Toys: puzzle, ball, doll

Modifiers

Locations: bathroom, bedroom, playground, street
Play-manners: well, badly
Win-manners: with-ease, with-difficulty

Specialty System

charlie $\leftrightarrow$ chess
heidi $\leftrightarrow$ hide&seek
sophia $\leftrightarrow$ soccer

Playing your specialty $\rightarrow$ higher win probability

(Frank et al., 2009)

44 Atomic Propositions

A **situation** $\omega \in \{0,1\}^{44}$ assigns a truth value to each proposition:

Predicate	**Example**	**Count**
play(person, game)	play(charlie, chess)	9
play(person, toy)	play(heidi, ball)	9
win(person)	win(sophia)	3
lose(person)	lose(charlie)	3
location(person, place)	location(heidi, bedroom)	12
play(person, manner)	play(sophia, well)	6
win(manner)	win(with-ease)	2
	Total	**44**

167 Hard Constraints → 28,676 Valid Situations

- **Game** (19): at most 1 game per person; player-count requirements
- **Toy** (33): at most 1 toy; soccer needs ball; no toys during chess/hide&seek
- **Location** (72): exactly 1 location per person; co-location for shared games
- **Winning** (31): can't both win & lose; at most 1 winner
- **Manner** (12): at most 1 play manner; requires playing a game

All formalized as Z3 formulas: $\neg(c_1 \wedge c_2 \wedge \ldots)$

Drastic reduction

$2^{44} \approx 17.6$ **trillion** possible binary vectors

$\downarrow$ 167 hard constraints

28,676 valid situations

Reduction factor $\sim$613 million

$+$ **Soft constraints** shape the probability distribution (playing well → winning more likely)

(Frank et al., 2009; De Moura and Bjørner, 2008; Brasoveanu, 2026)

Truth-Conditional Representations

Pool MCMC sampling (Brasoveanu, 2026) produces representative situations.
A **situation vector** is a continuous, distributed & probabilistic representation of the situations where a proposition is true.
A **competitive neural model** (Self-Organizing Map variant, $n = 150$ neurons) learns a weight matrix $\mathbf{W} \in [0,1]^{150 \times 44}$:

$$[\![\varphi_k]\!] = \mathbf{W}_{\cdot k} \in [0,1]^{150} \qquad \text{(column } k \text{ = meaning of atomic prop } \varphi_k\text{)}$$

Truth-Conditional Representations

Pool MCMC sampling (Brasoveanu, 2026) produces representative situations.
A **situation vector** is a continuous, distributed & probabilistic representation of the situations where a proposition is true.
A **competitive neural model** (Self-Organizing Map variant, $n = 150$ neurons) learns a weight matrix $\mathbf{W} \in [0,1]^{150 \times 44}$:

$$[\![\varphi_k]\!] = \mathbf{W}_{\cdot k} \in [0,1]^{150} \qquad \text{(column } k \text{ = meaning of atomic prop } \varphi_k\text{)}$$

Complex meanings via differentiable vectorized fuzzy logic:

$$\begin{aligned} [\![\neg\varphi]\!] &= \mathbf{1} - [\![\varphi]\!] \\ [\![\varphi \wedge \psi]\!] &= [\![\varphi]\!] \odot [\![\psi]\!] \quad \text{(element-wise product)} \\ [\![\varphi \vee \psi]\!] &= [\![\varphi]\!] + [\![\psi]\!] - [\![\varphi]\!] \odot [\![\psi]\!] \end{aligned}$$

Probability estimates: $P(\varphi) = \frac{1}{n} \sum_{i=1}^{n} [\![\varphi]\!]_i$
(correlation $r \geq 0.998$ with MCMC sample frequencies)

(Frank et al., 2009; Kohonen, 1995, 2013; Brasoveanu, 2026)

Entity Vectors: Our Extension

Formal semantics treats **entities** (type e) and truth values (type t) as **co-equal primitives**.

Extension: append 17 binary entity-presence indicators to the competitive-model input:

- Entity e is 1 in situation ω if e appears as an argument in any true atomic prop
- Input dimension: $44 \rightarrow 61$
- Competitive model produces $\mathbf{W}^{+} \in [0,1]^{150 \times 61}$

Entity Vectors: Our Extension

Formal semantics treats **entities** (type e) and truth values (type t) as **co-equal primitives**.

Extension: append 17 binary entity-presence indicators to the competitive-model input:

- Entity e is 1 in situation ω if e appears as an argument in any true atomic prop
- Input dimension: $44 \rightarrow 61$
- Competitive model produces $\mathbf{W}^{+} \in [0,1]^{150\times 61}$

Two training-target conditions:

Condition	Target dim	Content
$-$ent	150	truth-conditional only
$+$ent	300	truth-conditional $\|$ entity-presence

Question: Does making entity information explicit improve compositional generalization?

(Kamp, 1981; Heim, 1982; Brasoveanu, 2026)

Microlanguage

13,556 sentences from a feature-based context-free grammar:

- 40 lexical items
- Active voice: *charlie plays chess*
- Passive voice: *chess is played by charlie*
- Beat / lose: *sophia beats boy*, *boy loses to sophia*
- Optional modifiers: manner (*plays well*), location (*in bedroom*)

Microlanguage

13,556 sentences from a feature-based context-free grammar:

- 40 lexical items
- Active voice: *charlie plays chess*
- Passive voice: *chess is played by charlie*
- Beat / lose: *sophia beats boy*, *boy loses to sophia*
- Optional modifiers: manner (*plays well*), location (*in bedroom*)

Four synonym pairs for systematicity testing:

charlie $\leftrightarrow$ boy	soccer $\leftrightarrow$ football
puzzle $\leftrightarrow$ jigsaw	bathroom $\leftrightarrow$ shower

Existential quantifiers $\rightarrow$ disjunctions:

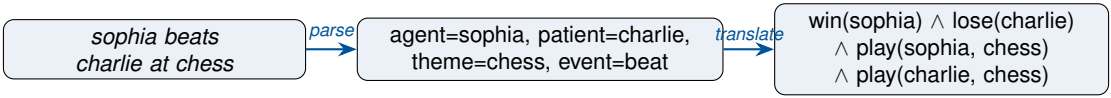
someone $\rightarrow$ charlie $\vee$ heidi $\vee$ sophia

girl $\rightarrow$ heidi $\vee$ sophia

(Frank et al., 2009)

Sentence → Z3 Logical Form → Target Vector

Translation via intermediate event structure (thematic roles):



Target vector: the desired output meaning vector for this sentence.

$$[\![\text{sentence}]\!] = [\![\text{win(sophia)}]\!] \odot [\![\text{lose(charlie)}]\!] \odot [\![\text{play(sophia, chess)}]\!] \odot [\![\text{play(charlie, chess)}]\!]$$

After filtering inconsistent sentences:
6,279–6,789 train / **2,330–2,840 test** per split

Outline

Seven Neural Architectures

All share the same encoding structure:

word embedding $\rightarrow$ sequence encoder $\rightarrow$ linear projection $\rightarrow$ sigmoid $\rightarrow [0,1]^n$

Recurrent (Experiment 1)

SRN: Simple Recurrent Network
(Elman, 1990; Frank et al., 2009)

LSTM: Long Short-Term Memory
(Hochreiter and Schmidhuber, 1997; Gers et al., 2000)

GRU: Gated Recurrent Unit
(Cho et al., 2014)

Attention (Exp. 1 & 2)

AbsPE: absolute/sinusoidal positional encoding

RoPE: rotary positional encoding
(Vaswani et al., 2017; Su et al., 2021)

H48/**H80**: hidden dimension 48/80
Exp. 1 uses H48; Exp. 2 tests higher-capacity H80

140 total models: 7 arch $\times$ $\pm$entity $\times$ 2 splits $\times$ 5 seeds

Capacity Matching ($\sim$66k Parameters)

To isolate **architectural effects** from parameter count:

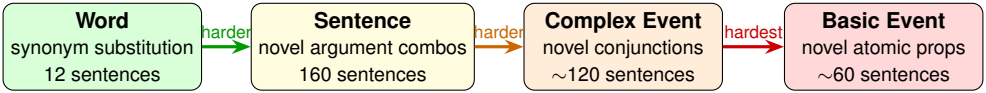
Architecture	Hidden	Layers	Heads	Params ($-$ent)
SRN	178	1	—	66,010
GRU	90	1	—	66,210
LSTM	80	1	—	66,950
Attn AbsPE	48	2	4	65,670
Attn RoPE	48	2	4	65,670

Training: AdamW, batch 64, gradient clip 3.0, adaptive LR schedule, 750 epochs
Loss: MSE on full target vector (150 or 300 dims)

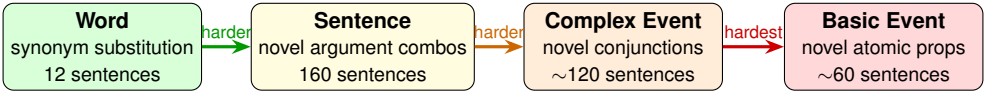
Experiment 2: attention with hidden dim **80** (H80, $\sim$170k params)
$\rightarrow$ tests whether per-position capacity alone explains recurrent advantage

(Loshchilov and Hutter, 2019)

Four Systematicity Tests: Increasing Difficulty



Four Systematicity Tests: Increasing Difficulty



Key distinction: what has the model seen during training?

- **Word**: same truth conditions, different lexical form
- **Sentence**: same truth conditions, different syntactic structure
- **Complex**: individual conjuncts seen, but not their conjunction
- **Basic**: the atomic proposition itself was never a training target

Systematicity Test Examples I

Word (easiest)

Train: *charlie plays football*, *boy plays soccer*
Test: *charlie plays soccer*, *boy plays football*
Same truth conditions seen under different lexical items

Sentence

Train: *X beats Y* + *X loses to Y* (various pairs)
Test: held-out *Y beats X* + *Y loses to X*
Same truth conditions seen under different syntactic structures

(Frank et al., 2009)

Systematicity Test Examples II

Complex Event

Train: *charlie plays chess in bedroom* (not playground)
Test: *charlie plays chess in playground*
Individual conjuncts seen, but not this conjunction

Basic Event (hardest)

Test: *heidi plays with ball*—target for *play(heidi, ball)*
never seen in any training sentence
The atomic proposition itself is entirely novel

(Frank et al., 2009)

Evaluation Metrics I: Comprehension Score

Comprehension score (CompS) (Frank et al., 2009): how much the model output **z** changes belief in target event **a** relative to prior belief.

$$\text{CompS}(\mathbf{a} \mid \mathbf{z}) = \begin{cases} \frac{P(\mathbf{a}|\mathbf{z}) - P(\mathbf{a})}{1 - P(\mathbf{a})} & \text{if } P(\mathbf{a} \mid \mathbf{z}) \geq P(\mathbf{a}) \\ \frac{P(\mathbf{a}|\mathbf{z}) - P(\mathbf{a})}{P(\mathbf{a})} & \text{otherwise} \end{cases}$$

- $P(\mathbf{a} \mid \mathbf{z}) = \frac{\frac{1}{150}\sum_{i=1}^{150}(\mathbf{a}_i \cdot \mathbf{z}_i)}{\frac{1}{150}\sum_{i=1}^{150}\mathbf{z}_i}$: probability of target event **a** given output vector **z**.
- $P(\mathbf{a}) = \frac{1}{150}\sum_{i=1}^{150}\mathbf{a}_i$: prior probability of **a**.
- CompS is the attained proportion of the maximum possible belief change:
  - if belief increases, divide by the maximum possible increase $1 - P(\mathbf{a})$;
  - if belief decreases, divide by the maximum possible decrease $P(\mathbf{a})$.
- Positive CompS indicates correct support for the described situation; negative CompS indicates misunderstanding.
- CompS **always computed on truth-conditional 150-dim part**.

Evaluation Metrics II: Foils and Advantage

Z3 validation: Z3 is a constraint solver; SAT means "possible under the microworld constraints," UNSAT means "impossible under those constraints."

Foil events (competing interpretations) via two-step Z3 validation:

1. **Validity:** SAT($f \wedge$ constraints)—foil event is possible.
2. **Competition:** UNSAT($f \wedge d \wedge$ constraints)—foil f incompatible with described event d.

Candidate foils vary exactly one event-structure slot at a time: agent, theme, location, or manner.

Advantage score (our primary metric):

$$\text{advantage} = \text{CompS}_{\text{described}} - \text{mean}(\text{CompS}_{\text{foils}})$$

- High advantage: model supports the described event more than its foils.
- Low or negative advantage: model fails to prefer the intended interpretation.

Outline

Main Results: Test-Only Advantage Scores

Arch	Ent	Word	Sent	Cmplx	Basic
SRN	−ent	1.65	**1.16**	**1.36**	0.67
	+ent	**1.73**	1.12	**1.37**	0.74
GRU	−ent	1.72	1.13	1.30	0.65
	+ent	1.72	1.08	1.20	**0.78**
LSTM	−ent	1.68	1.14	1.28	0.67
	+ent	1.71	1.12	1.24	**0.77**
AbsPE	−ent	1.71	0.95	1.05	0.59
	+ent	1.72	1.02	1.10	0.68
RoPE	−ent	1.67	1.14	1.07	0.63
	+ent	1.70	1.12	1.05	0.69

Each cell: mean across 10 models (2 splits $\times$ 5 seeds), aggregated over modifier variants.

Ent labels: −ent = truth-conditional target only; +ent = target also includes entity-presence vectors.

Test labels: Word = synonym substitution; Sent = novel beat/lose pattern; Cmplx = novel conjunction; Basic = novel atomic proposition.

Finding 1: Difficulty Has Two Dimensions

Systematicity difficulty $\neq$ single gradient

Dimension 1: Test type

Word $>$ Sentence $>$ Complex Event $>$ Basic Event

Dimension 2: Modifier complexity

Canonical $>$ +Location $>$ +Manner $>$ +Both

Each modifier adds conjuncts to the Z3 formula $\rightarrow$
the model must compose *more* semantic components simultaneously

Modifier Complexity Breakdown

Grand means across all 5 architectures and $\pm$entity. Canonical = no modifier; +Loc = location modifier; +Man = manner modifier; +L+M = both.

Test	Canonical	+Loc	+Man	+L+M	Agg
Word	**1.94**	—	1.58	—	1.70
Sentence	**1.62**	1.26	1.12	0.86	1.10
Complex	**1.37**	—	1.10	—	1.19
Basic	**0.83**	0.65	—	—	0.68

Canonical column preserves the expected hierarchy: $1.94 > 1.62 > 1.37 > 0.83$

Modifier Complexity Breakdown

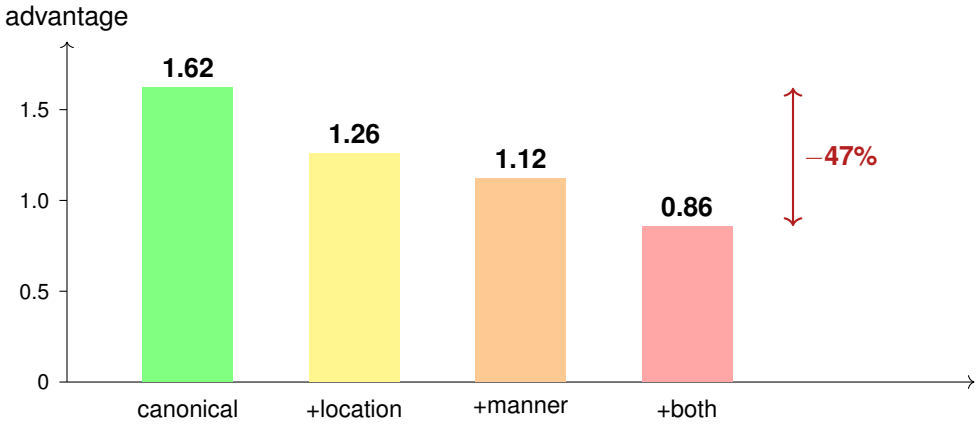
Grand means across all 5 architectures and $\pm$entity. Canonical = no modifier; +Loc = location modifier; +Man = manner modifier; +L+M = both.

Test	Canonical	+Loc	+Man	+L+M	Agg
Word	**1.94**	—	1.58	—	1.70
Sentence	**1.62**	1.26	1.12	0.86	1.10
Complex	**1.37**	—	1.10	—	1.19
Basic	**0.83**	0.65	—	—	0.68

Canonical column preserves the expected hierarchy: $1.94 > 1.62 > 1.37 > 0.83$

Why does Aggregate reverse Sentence and Complex?
Sentence is 94% non-canonical (with hardest +L+M, i.e. both modifiers, at 0.86)
Complex is only 67% non-canonical (manner-only at 1.10)
$\rightarrow$ **different mixtures** of modifier complexity, not a true reversal

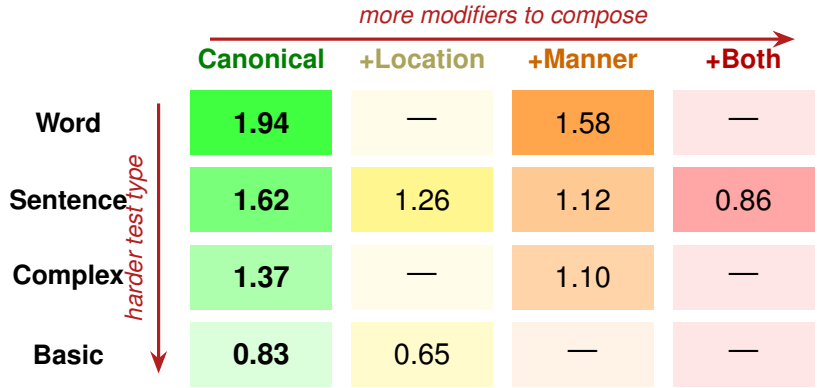
Modifier Complexity in Action: Sentence Group



sophia beats charlie → *sophia beats charlie in the bedroom*

→ *sophia beats charlie with ease* → *sophia beats charlie with ease in the bedroom*

Systematicity Difficulty: A Two-Dimensional Landscape



Hue = modifier type, as on previous slide; darker shade = higher advantage within column; — = not applicable.

Finding 2: Architecture Effects Across Difficulty Levels

Easy tests (Word): all architectures cluster around $\sim$1.7

Hard tests: architecture becomes the dominant factor

- Sentence: $F(6, 139) = 20.72$, $p < .001$
- Complex: $F(6, 145) = 48.76$, $p < .001$
- Basic: $F(6, 137) = 7.18$, $p < .001$

Finding 2: Architecture Effects Across Difficulty Levels

Easy tests (Word): all architectures cluster around $\sim$1.7

Hard tests: architecture becomes the dominant factor

- Sentence: $F(6, 139) = 20.72$, $p < .001$
- Complex: $F(6, 145) = 48.76$, $p < .001$
- Basic: $F(6, 137) = 7.18$, $p < .001$

These gaps exist **even at the canonical level**:

- Canonical Sentence: AbsPE 1.27 vs. SRN 1.63 and LSTM 1.69
- RoPE does well on canonical Sentence (1.72), but not on canonical Complex
- Canonical Complex: AbsPE 1.16, RoPE 1.19 vs. SRN 1.60

$\Rightarrow$ The attention deficit is an **intrinsic architectural effect**,
not an artifact of modifier complexity

Finding 3: Gated Recurrent Networks $\gg$ Transformers on Basic Event

Basic Event means (Experiment 1):

	Basic (+ent)	Basic (−ent)
GRU	**0.78**	0.65
LSTM	**0.77**	0.67
SRN	0.74	0.67
RoPE	0.69	0.63
AbsPE	0.68	0.59

Pairwise significance statements below use the full 7-architecture analysis, including H80 attention models.

Finding 3: Gated Recurrent Networks $\gg$ Transformers on Basic Event

Basic Event means (Experiment 1):

	Basic (+ent)	Basic (−ent)
GRU	**0.78**	0.65
LSTM	**0.77**	0.67
SRN	0.74	0.67
RoPE	0.69	0.63
AbsPE	0.68	0.59

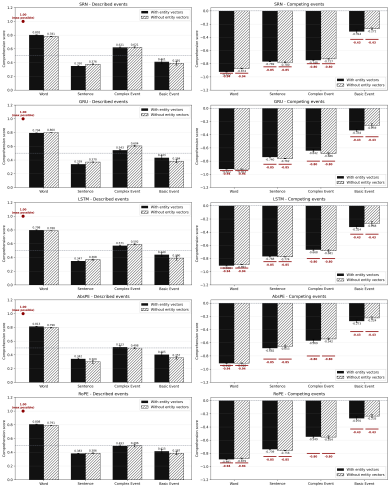
Pairwise significance statements below use the full 7-architecture analysis, including H80 attention models.

Pairwise contrasts on Basic Event (+ent):

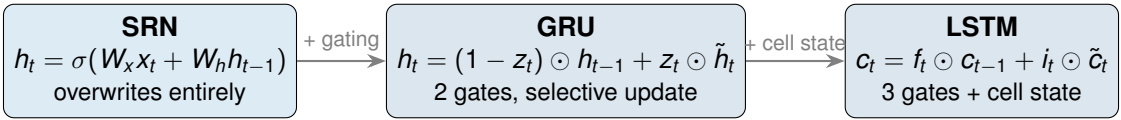
- GRU vs. all 4 attention variants: $p = .001$–$.034$
- LSTM vs. 3 of 4 attention variants: $p = .002$–$.034$; vs. RoPE: $p = .053$
- GRU vs. LSTM: $p = 1.00$ (statistically indistinguishable)
- **SRN does not significantly outperform attention** in the +ent condition

$\Rightarrow$ **Gating**, not recurrence alone, is the critical factor

Described vs. Competing Scores Across Architectures



Why Gating, Not Just Recurrence?



Basic: 0.74 **Basic: 0.78** **Basic: 0.77**

$$\text{GRU: } r_t = \sigma(W_{xr} x_t + W_{hr} h_{t-1} + b_r) \text{ (reset)}, \quad z_t = \sigma(W_{xz} x_t + W_{hz} h_{t-1} + b_z) \text{ (update)}$$
$$\tilde{h}_t = \tanh(W_{xh} x_t + W_{hh}(r_t \odot h_{t-1}) + b_h) \text{ (candidate)}$$
$$\text{LSTM: } f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + b_f) \text{ (forget)}, \quad i_t = \sigma(W_{xi} x_t + W_{hi} h_{t-1} + b_i) \text{ (input)}$$
$$o_t = \sigma(W_{xo} x_t + W_{ho} h_{t-1} + b_o) \text{ (output)}, \quad \tilde{c}_t = \tanh(W_{xc} x_t + W_{hc} h_{t-1} + b_c) \text{ (candidate)}$$

- GRU matches LSTM despite lacking separate cell state
- SRN shares the sequential bottleneck but lacks selective retention

$\Rightarrow$ **Gating** = the ability to selectively retain/update information

This helps **sub-sentential composition**: combining words inside an unseen event description.

But Wait: SRN Excels at Complex Event

	Complex (−ent)	Complex (+ent)
SRN	**1.36**	**1.37**
GRU	1.30	1.20
LSTM	1.28	1.24
AbsPE	1.05	1.10
RoPE	1.07	1.05

SRN **significantly outperforms both gated architectures** on Complex Event, and this holds at both Complex modifier levels (canonical and +manner). **Interpretation:** Complex Event tests *sentence-level* composition (recombining familiar conjuncts: activity $\wedge$ location).

- SRN's simpler overwriting dynamics may suit incremental propositional conjunction.
- Gating may not help recombine familiar sentence-level components.

$\Rightarrow$ Gating specifically helps **sub-sentential** composition
(combining words inside an unseen event description; Basic Event)

Finding 4: Entity Vectors Help Most on Basic Event

Basic Event entity benefit (Experiment 1):

Arch	Δ (+ent $-$ $-$ent)	p
GRU	+0.125	$< .001$
LSTM	+0.106	$< .001$
SRN	+0.063	—
AbsPE	+0.096	.002
RoPE	+0.063	—

Overall entity effect on Basic:
$F(1, 189) = 59.17$, $p < .001$

Entity effects by test group:

- **Basic:** strong pooled effect ($p < .001$)
- **Word:** pooled positive effect ($p < .001$)
- Sentence: null ($p = .95$)
- Complex: null ($p = .44$)

Entity benefit on Basic is **much larger on test** than on training sentences

(test: +0.094 vs. train: +0.008)

$\Rightarrow$ Gated architectures benefit most, but the broader pattern is the same: entity information helps most when composing **novel entity combinations**

Finding 5: Substantial Generalization Gap

Generalization gap = performance on training sentences minus performance on held-out test sentences.

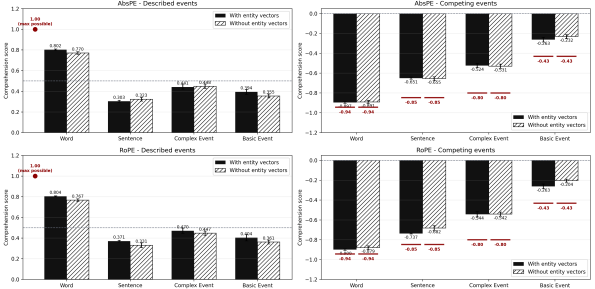
	Training	**Test**
Basic Event (all archs)	$\sim$1.01–1.10	$\sim$0.57–0.78

- Train/test contrast on Basic: $F = 292.45$, $p < .001$
- Entity $\times$ train/test interaction: $F = 67.07$, $p < .001$
  $\rightarrow$ entity vectors help in both subsets, but **much more on test**
- Complex Event also shows a strong train/test gap with architecture $\times$ train/test interaction ($F = 44.98$, $p < .001$)

All architectures are ultimately doing fairly poorly on the hardest test.
True compositional generalization **remains challenging**.

Experiment 2: More Capacity $\neq$ Better Generalization

Attention H80 models (hidden dimension 80, matching LSTM):
$\sim$170k parameters—**2.5**$\times$ the recurrent models



Higher capacity does **not** materially improve over H48 attention (hidden dimension 48); in some cases, H80 is slightly *worse*.

$\Rightarrow$ The recurrent advantage is about **inductive bias**, not parameter count.

Outline

Why Do Gated Recurrent Models Generalize Better?

Three hypotheses tested:

1. **Capacity per position?**
   Recurrent models have larger hidden dims (80–178 vs. 48)
   $\Rightarrow$ *Ruled out*: H80 attention ($\sim$170k params) doesn't improve

2. **Sequential inductive bias?**
   Left-to-right processing, fixed-size hidden state
   $\Rightarrow$ *Necessary but insufficient*: SRN shares this but doesn't match GRU/LSTM

3. **Gating mechanism** ✓
   Selective retention and updating of information
   $\Rightarrow$ Maintains compositionally relevant features across timesteps

Self-Attention and Information Entanglement

Self-attention computes pairwise interactions between *all* positions:

- Every head mixes "what" (lexical features) with "where/how" (structural roles)
- This may **entangle** object-level and relational information
  (Webb et al., 2024; Altabaa and Lafferty, 2025)

For compositional semantics, this may be problematic:

- **Lexical semantics** (meanings of words) and
  compositional semantics (how meanings combine)
  may benefit from *separate processing streams*

Self-Attention and Information Entanglement

Self-attention computes pairwise interactions between *all* positions:

- Every head mixes "what" (lexical features) with "where/how" (structural roles)
- This may **entangle** object-level and relational information
  (Webb et al., 2024; Altabaa and Lafferty, 2025)

For compositional semantics, this may be problematic:

- **Lexical semantics** (meanings of words) and
  compositional semantics (how meanings combine)
  may benefit from *separate processing streams*

One possible interpretation of our results:

- Sequential bottlenecks force compression
- Gates enable selective retention of compositionally relevant features
- Standard self-attention mixes all positions pairwise at every layer

Promising direction: Relational Abstractors, Dual Attention Transformers
(Altabaa et al., 2024; Altabaa and Lafferty, 2025)

Implications for Compositionality Research

- **Architectural sophistication $\neq$ compositional ability**
  Transformers underperform simpler gated recurrent models on the hardest compositional generalization tests

- Aligns with previous findings: ANNs struggle with compositional generalization
  (but our setting targets truth conditions, not symbolic transduction)

- **All** architectures are ultimately doing fairly poorly:
  best Basic Event advantage $= 0.78$ (vs. Word ≈ 1.7–1.9)

- The right inductive bias matters more than model size for compositional generalization in this paradigm

(Lake and Baroni, 2018; Kim and Linzen, 2020; Keysers et al., 2020)

Implications for Formal Semantics

Entities as Theoretical Primitives

- Entity-presence vectors provide the **largest benefit** precisely where formal theory predicts: novel entity combinations (Basic Event)
- Benefit is much larger on *test* than on *training* sentences
- Echoes dynamic semantics (Kamp, 1981; Heim, 1982): entity discourse referents are essential for quantification & anaphora

Two-Dimensional Difficulty

- Modifier complexity—the number of semantic components being simultaneously composed—is a **linguistically grounded** predictor of generalization difficulty
- Distinct from (and orthogonal to) the type of systematicity test

Implications for the Systematicity Debate

Positive: All architectures exhibit positive systematicity even on Basic Event (0.57–0.78)

- Neural networks *can* learn interpretation functions supporting compositional generalization without classical symbols

Implications for the Systematicity Debate

Positive: All architectures exhibit positive systematicity even on Basic Event (0.57–0.78)

- Neural networks *can* learn interpretation functions supporting compositional generalization without classical symbols

But: Systematicity is **graded**, not all-or-nothing

- Near-ceiling on synonym substitution in canonical sentences
- Very modest on novel atomic propositions with modifiers

Implications for the Systematicity Debate

Positive: All architectures exhibit positive systematicity even on Basic Event (0.57–0.78)

- Neural networks *can* learn interpretation functions supporting compositional generalization without classical symbols

But: Systematicity is **graded**, not all-or-nothing

- Near-ceiling on synonym substitution in canonical sentences
- Very modest on novel atomic propositions with modifiers

The neural systematicity landscape:

- Complex interaction of architectural inductive biases, representation richness, and compositional complexity

Code & data: `https://github.com/abrsvn/neural_model_theoretic_interpretation_ACL_2026`

References I

Altabaa, A. and Lafferty, J. (2025). Disentangling and integrating relational and sensory information in transformer architectures. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 1271–1297. PMLR.

Altabaa, A., Webb, T., Cohen, J. D., and Lafferty, J. (2024). Abstractors and relational cross-attention: An inductive bias for explicit relational reasoning in transformers. In *ICLR*.

Brasoveanu, A. (2026). Distributed situation models: Probabilistic representations of truth-conditions. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 48.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

De Moura, L. and Bjørner, N. (2008). *Z3: An efficient SMT solver*. Springer.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

Fodor, J. A. and Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71.

Frank, S. L., Haselager, W. F., and Van Rooij, I. (2009). Connectionist semantic systematicity. *Cognition*, 110(3):358–379.

Gers, F. A., Schmidhuber, J., and Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12(10):2451–2471.

Heim, I. (1982). *The Semantics of Definite and Indefinite Noun Phrases*. PhD thesis, University of Massachusetts, Amherst.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Kamp, H. (1981). A theory of truth and semantic representation. In Groenendijk, J., Janssen, T., and Stokhof, M., editors, *Formal Methods in the Study of Language*, pages 277–322, Amsterdam. Mathematical Centre.

Keysers, D., Schärli, N., Scales, N., Buisman, H., Furrer, D., Kashubin, S., Momchev, N., Sinopalnikov, D., Stafiniak, L., Tihon, T., Tsarkov, D., Wang, X., van Zee, M., and Bousquet, O. (2020). Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations*.

Kim, N. and Linzen, T. (2020). COGS: A compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9087–9105. Association for Computational Linguistics.

Kohonen, T. (1995). *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg.

References II

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37:52–65.

Lake, B. and Baroni, M. (2018). Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2873–2882. PMLR.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Su, J., Lu, Y., Pan, S., Wen, B., and Liu, Y. (2021). Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008.

Venhuizen, N. J., Crocker, M. W., and Brouwer, H. (2019). Expectation-based comprehension: Modeling the interaction of world knowledge and linguistic experience. *Discourse Processes*, 56(3):229–255.

Venhuizen, N. J., Hendriks, P., Crocker, M. W., and Brouwer, H. (2022). Distributional formal semantics. *Information and Computation*, 287:1–21.

Webb, T. W., Frankland, S. M., Altabaa, A., et al. (2024). The relational bottleneck as an inductive bias for efficient abstraction. *Trends in Cognitive Sciences*.

Appendix: Per-Modifier Breakdown — Complex & Basic

Arch		Complex			Basic		
		Can	**+Man**	**Agg**	**Can**	**+Loc**	**Agg**
SRN	−ent	**1.60**	**1.22**	**1.35**	0.79	0.64	0.66
	+ent	**1.61**	**1.25**	**1.37**	0.85	0.70	0.73
LSTM	−ent	1.48	1.17	1.27	0.83	0.63	0.66
	+ent	1.44	1.14	1.24	0.90	0.74	0.76
GRU	−ent	1.51	1.18	1.29	0.83	0.61	0.64
	+ent	1.36	1.10	1.18	**0.95**	0.73	**0.77**
AbsPE	−ent	1.16	0.98	1.04	0.78	0.54	0.58
	+ent	1.18	1.03	1.08	0.78	0.66	0.68
RoPE	−ent	1.19	0.98	1.05	0.81	0.59	0.62
	+ent	1.14	0.99	1.04	0.79	0.67	0.68

Appendix: Detailed Architecture Specifications

Parameter	**SRN**	**GRU**	**LSTM**	**Attn AbsPE**	**Attn RoPE**
Hidden dim	178	90	80	48	48
Num layers	1	1	1	2	2
Attn heads	—	—	—	4	4
Head dim	—	—	—	12	12
FFN expansion	—	—	—	$4\times$	$4\times$
FFN activation	—	—	—	GELU	GELU
Pos. encoding	implicit	implicit	implicit	sinusoidal	rotary
Params ($-$ent)	66,010	66,210	66,950	65,670	65,670
Params ($+$ent)	92,860	79,860	79,100	73,020	73,020