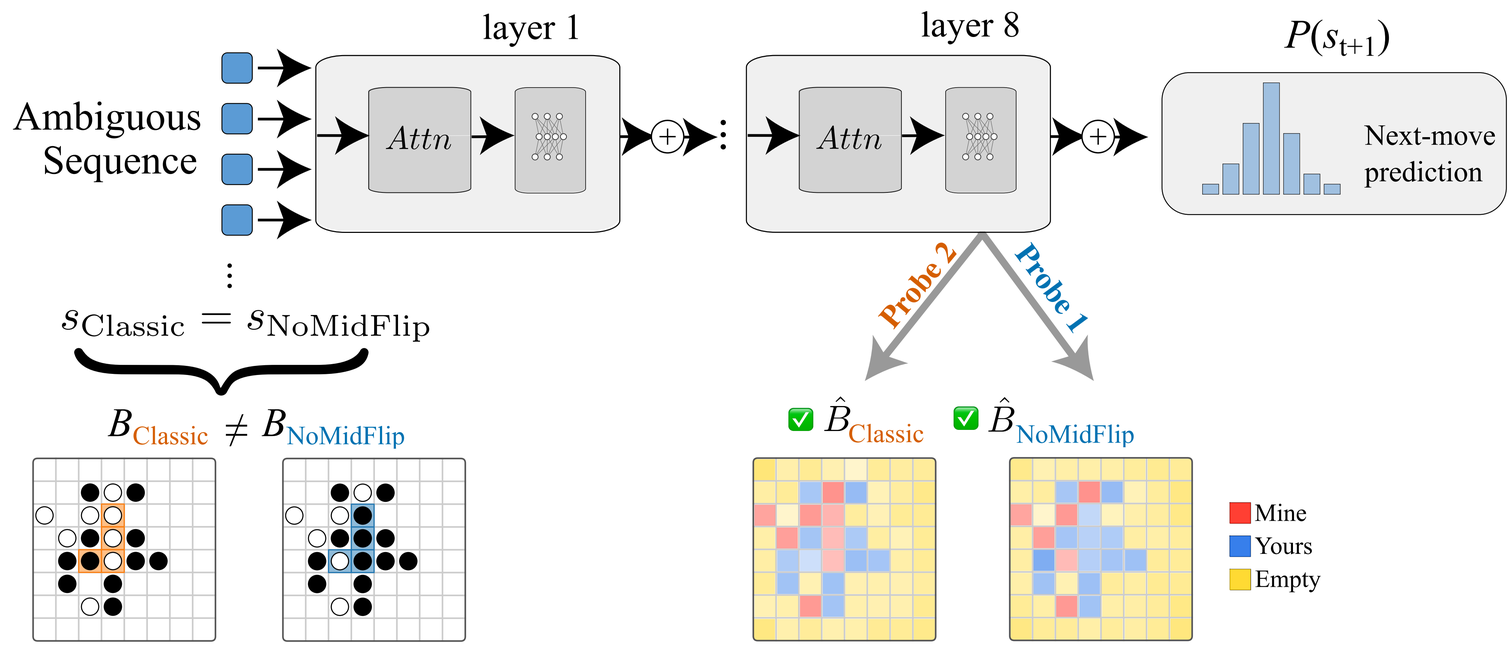
MetaOthello: A Controlled Study of Multiple World Models in Transformers

Aviral Chawla* — University of Vermont, Burlington, VT

Galen Hall* — University of Michigan, Ann Arbor, MI

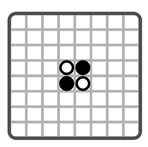
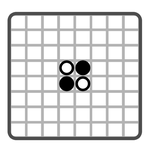
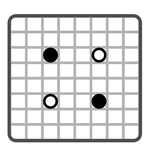
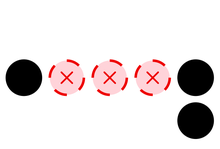
Juniper Lovato — University of Vermont, Burlington, VT



TL;DR: ***How do transformers represent the world when different latent processes can generate the same data?*** To answer this question, we construct a set of toy "worlds" based on the game Othello, each using different rules but operating on the same board and syntax. We train small GPT models on sequences generated from these mixtures. Finally, we show that the models learn shared world-state representations and localize a steerable mechanism by which they resolve ambiguity.

Setup

We define each Othello-like world as a game, $g = (B_0, V, U)$, played on an 8 x 8 board where $B \in \mathcal{B} = \{\text{white}, \text{black}, \text{empty}\}^{64}$ is the latent board state. Above are 3 game rules with different rules. We also create **Iago:** which has same rules as classic but presented with shuffled token indices.

Title	Initialization	Validation	Update
Classic		 Flanking	
NoMiddleFlip		 Flanking	
DeleteFlanking		 Neighbor	

We train a small GPT for a ***single*** game or a ***mixed*** pair of games. Next, we train linear probes on model activations to decode if it learns the **latent board state.**

Findings

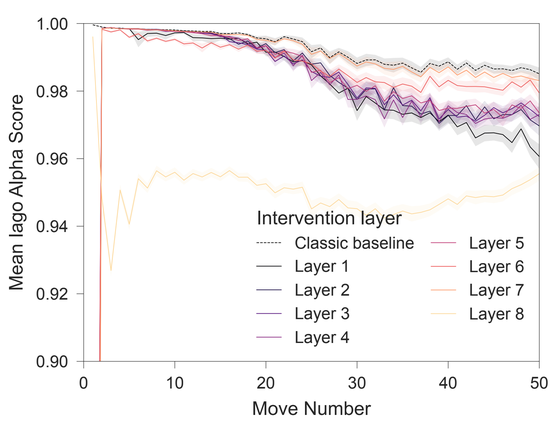
Q: Do models learn to predict valid moves?

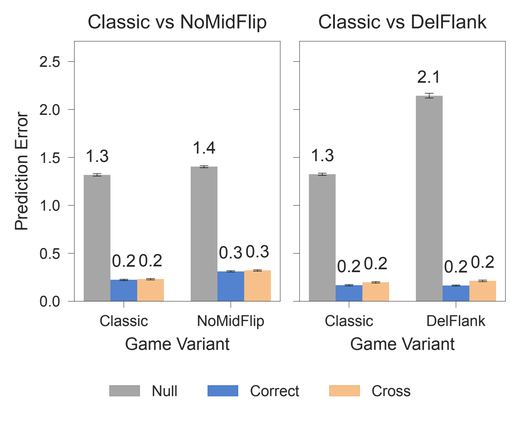
✅ **All models perform very well**. All models score ~ 1 α score. In mixed-game models, performance dips slightly (~0.5 %)

Q: Can linear probes recover board states?

✅ **Linear probes recover all games on all models.** Board states are linearly decodable for all games, even in mixed-game models. Results comparable to Nanda et al., 2025.

Q: How are the board representations related?

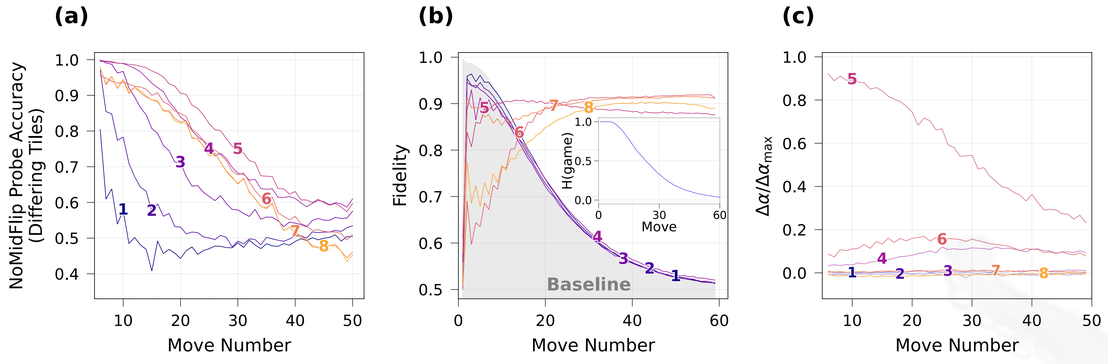




Classic and Iago: Representations for games with same rules, but different token mappings ➡️ world models identical up to a rotation.

Classic and rest: Steering board activation changes the model's predicted board state. Cross-game probes work nearly as well as matched probes!

Q: How do models handle ambiguity?



Ambiguity: Classic–NoMidFlip, often share same sequences: the same moves, but different board. We find early layers perform shared computation (L1-4), middle layer (L5) identifies game identity, which we can steer.

Takeaways

1. **Shared Representations.** Board latents for two disparate games is causally interchanegable.
2. **Shared Computation.** Models also perform general compute in early layers and then later specialize.
3. **Ambiguity.** We show mechanisms of how model resolve ambiguity.