Modeling Human-Like Learning in AI: Evaluating Recurrent Neural Networks' Capacity to Replicate Human Leaps of Insight in Sequential Decision-Making

Nina Mizrahi, Computational Neuroscience, Scripps College

W. M. Keck Science Department
Claremont McKenna • Scripps • Pitzer

Introduction

- This thesis investigates the differences between human and artificial intelligence (AI) learning through gameplay in Hexxed.io, a six-sided puzzle game designed to explore problem-solving strategies and learning dynamics .
- Neural networks excel at complex tasks like chess and Go, but lack the spontaneous creativity and adaptability of human problem-solving, relying instead on extensive and costly training processes[1].
- Human learning often involves "leaps of insight"—sudden cognitive shifts to optimal strategies or theories that allow for faster understanding—characterized by a top-down approach that draws on conceptual frameworks and prior knowledge. In contrast, AI relies on bottom-up processes, incrementally refining performance through repeated exposure to data and patterns[2]. (See Figure 1)
- This research examines the extent to which Recurrent Neural Networks (RNNs) can emulate the top-down dynamics of human learning using bottom-up mechanisms such as gradient descent, gating, and backpropagation, while also analyzing the contribution of non-linearity and input features to model learning.

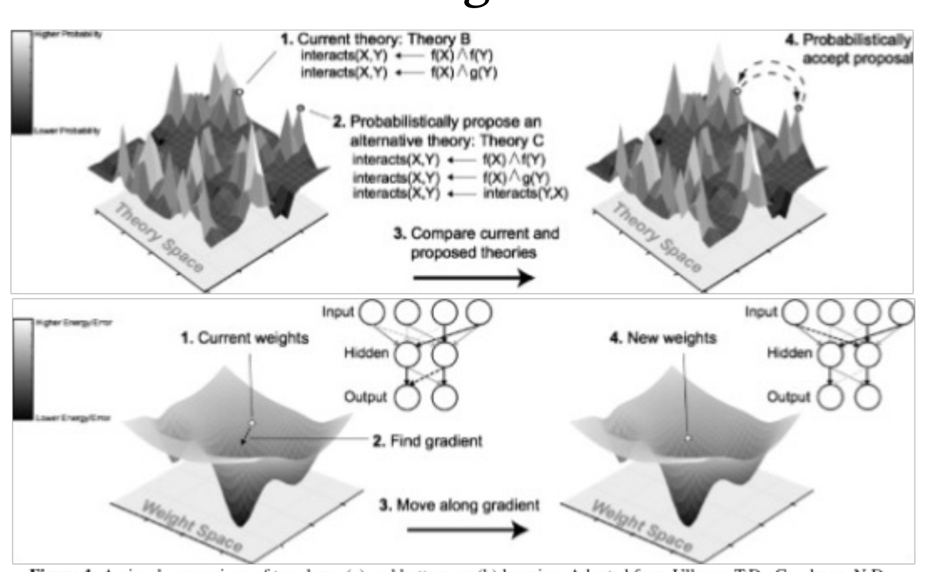
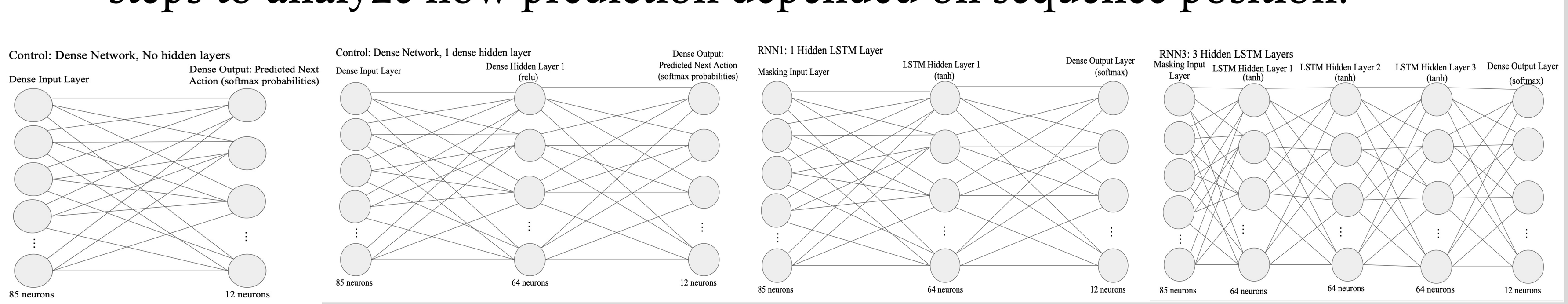


Figure 1: A visual comparison of top-down (a) and bottom up (b) learning. Adapted from Ullman, T.D., Goodman, N.D., & Tenenbaum, J.B. (2012). *Theory learning as stochastic search in the language of thought.* Cognitive Development, 27(4), 455-480.

Methods

- RNNs, specifically dense networks with Long Short-Term Memory (LSTM) layers, were trained on gameplay data from 1,647 human players who demonstrated leaps of insight.
- Human action sequences were processed into vectorized representations of board states and previous actions to train 3 RNNs with varying depths (see figures below). These models were compared to negative control models (NCM), and both a linear and non-linear baseline control network. All models were trained on each independent feature (e.g., board states or previous actions) to isolate their contribution to performance.
- Sequence tracking calculated the average accuracy across all sequence steps to analyze how prediction depended on sequence position.



Results

Figure 3 shows the plotted loss and accuracies for all models:

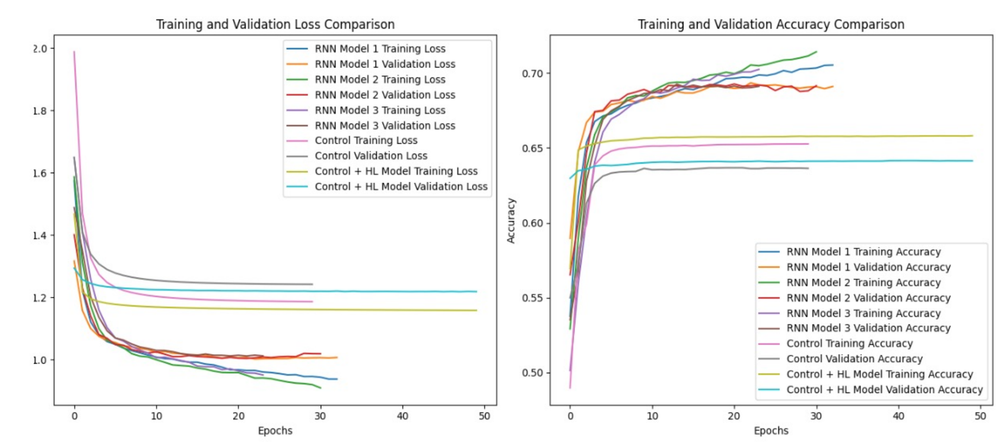


Figure 3: comparison of model performance across models trained on gamestate

- ❖ All models achieved accuracy >0.4746 demonstrate that they are identifying meaningful patterns in the data rather than relying on chance or the frequency of a single action.
- ❖ RNNs significantly outperformed the baseline models in capturing temporal and contextual patterns. The single-layer RNN achieved the highest accuracy of all models (68%).
- ❖ Adding more LSTM layers did not improve performance, suggesting that our specific task complexity was sufficiently captured by simpler architectures.

Figure 4 shows plotted loss and accuracies for models trained on each isolated feature:

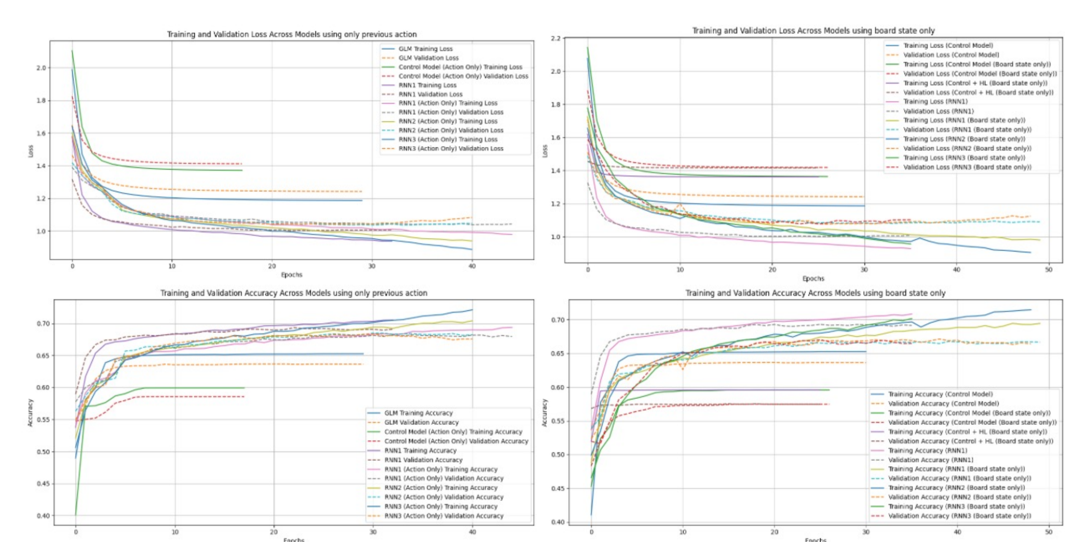


Figure 4: comparison of loss and accuracy across models using previous action (left) and board state (right).

- ❖ Models do best when provided full game state
- ❖ previous action is a more informative feature for predicting the next action compared to the board state alone
- ❖ RNN's performance did not decline as sharply as controls, indicating RNNs can generalize better without being overly reliant on specific features.

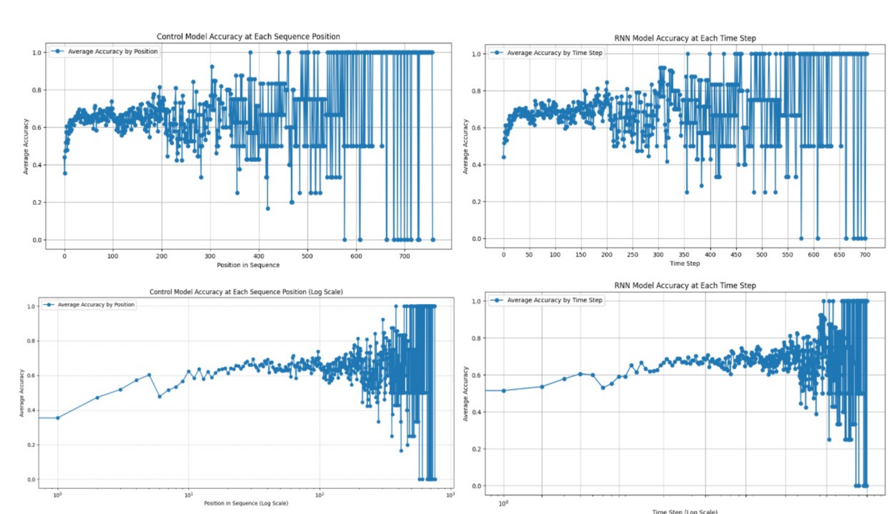


Figure 5: comparison of baseline control and RNN1 accuracy at each sequence position

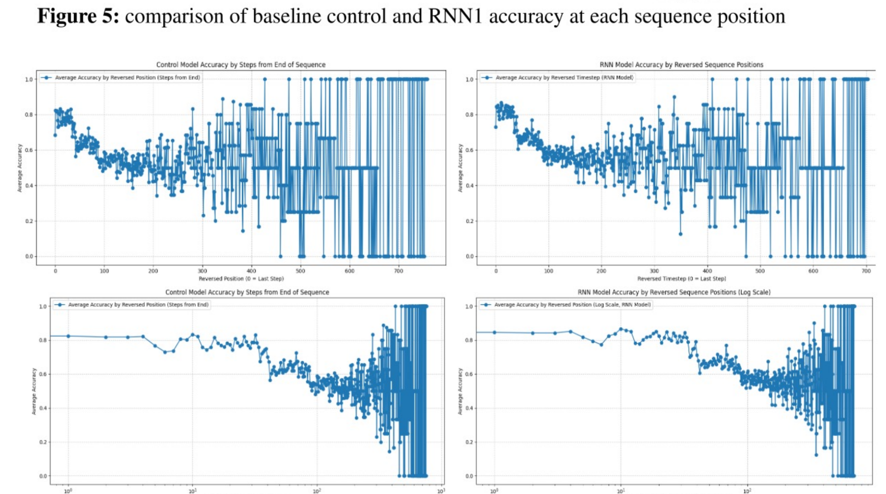


Figure 5.5: Comparison of baseline control and RNN1 accuracy at each reversed sequence position

Sequence tracking
- Similar distribution of accuracy within the first ~200 steps was the same between the two models
- growing accuracy up to the first 200 positions
- peak in accuracy around position 300

Sequence tracking from reversed position
- same patterns present as were observed in sequence tracking
- Plateaus in accuracy begin to appear at position 300 and are closer in count compared to above which started around 350

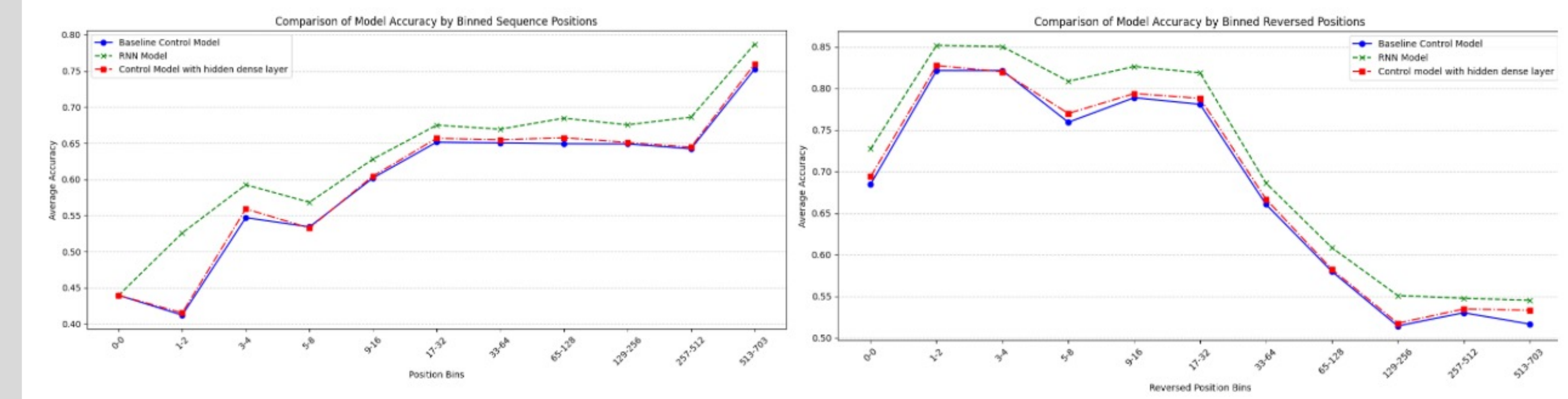


Figure 6: Comparison of model accuracies by binned positions

- ❖ RNN1 model outperforms both of the controls with higher accuracy at every position, both reversed and chronological
- ❖ All models follow roughly the same accuracy trends across bins, in terms of where they increase, decrease, and plateau
- ❖ The control model with an added hidden layer, though having marginally higher accuracy than the baseline control at certain position bins, was nearly identical.

Discussion

- Temporal dependencies, rather than non-linearity, were the primary drivers of the RNNs' increased ability to predict human actions, suggesting that temporal dependencies are critical for accurately modeling human-like decision-making.
- Though they were able to accurately approximate human learning behavior by 6.37%. RNN models relied heavily on patterns more explicitly present in the training data, failing to infer or generalize transferable strategies when handling longer sequences (>200 steps) that were underrepresented in the data. Therefore they failed to replicate the true leaps of insight which characterize top-down learning.
- Future studies should explore hybrid models that integrate reinforcement learning with RNNs, combining bottom-up pattern recognition with top-down goal-driven learning to make AI systems more human-like in how they learn and adapt.

Literature Cited

1. Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., & Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science, 362(6419), 1140–1144. https://doi.org/10.1126/science.aar6404
2. Ullman TD, Goodman ND, Tenenbaum JB. Theory learning as stochastic search in the language of thought. Cognitive Development. 2012;27(4):455-480. doi:10.1016/j.cogdev.2012.07.005

Acknowledgements

I would like to thank Professor Agarwal for their guidance and support throughout this project. I would also like to thank Professor Hope and for his support with the model design and implementation. I would like to recognize Tiago Quandera, Guillermo Franco-Mendoza, and my other Beatlab teammates who helped contribute to Hexxed and this project. Finally, I thank my family for their feedback and support throughout the semester.