Researching OthelloGPT

Yeu-Tong Lau

Exploring the OthelloGPT model using mechanistic interpretability tools and methods based on some project ideas from this Alignment Forum post by Neel Nanda.

The code is here, which is based on the colab notebook from Neel Nanda.

Key Observations

1. The model knows the cell of the current move is not blank by token embedding.
2. Layer 0 attention layer already knows (90% accuracy) the blank/not-blank board state. Layer 1-4 attention layer increases the blank/not-blank classification accuracy (to 99%)
3. MLP layer seems not related to blank/not-blank classification.
4. The main types of attention heads are “only attend to my previous moves” and “only attend to opponent’s previous moves”.
5. The first MLP layer already starts working on “flipping”(i.e. flip the “mine/theirs” state based on the board state and the current move).

Section 1: Current cell is not blank

Question: How does the model know that the cell for the current move is not blank?

Key findings:

1. The model knows the cell of the current move is not blank by token embedding (the model don't even need attention heads to process this information)
2. Evidence: Get 99% accuracy only using layer 0 resid_pre, and have 100% using layer 0 resid_post

Methods:

Use pre-trained linear probes to calculate the probabilities of "blank", "theirs", "mine" for the cell of current move on 50 example games based on different model representations. Then calculate whether the probability of "blank" is not the largest among three options (which means the cell is not blank). And then calculate the accuracy for different model representations (here we only use representation from Layer 0 since it already reaches 100% accuracy) by averaging the score for each game and each move (to reduce noise, I ignore the first three moves).

$$Accuracy \ = \frac{1}{\#games \ \ \#moves} \left(\sum_{game=1}^{50} \sum_{move=3}^{59} (blank \ \ \neq argmax\{P(blank),\ P(theirs),\ P(mine)\}) \right)$$

Representation	Accuracy (knowing the current cell is not blank)
Layer 0 Resid Pre	99.00%
Layer 0 Attn Out	85.50%
Layer 0 Resid Mid	99.57%
Layer 0 MLP Out	100%
Layer 0 Resid Post	100%

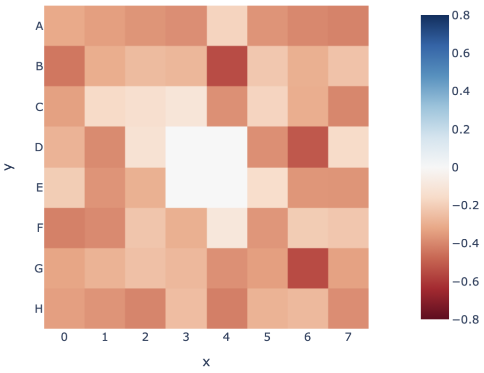
Discussion:

The model can tell the cell of the current move is not blank by token embedding since "blank direction" is negatively aligned with the embedding matrix (W_E). This makes sense since when the model sees some token, it directly implies that the cell should not be blank. Also I found that "my direction probe" is also negatively aligned with the embedding matrix. One possible hypothesis is that some part of the model (maybe layer 0 attention layer) will track the opponent's moves and mark with "theirs", and then some other parts of the model will flip the state (between "theirs" and "mine") according to some algorithms.

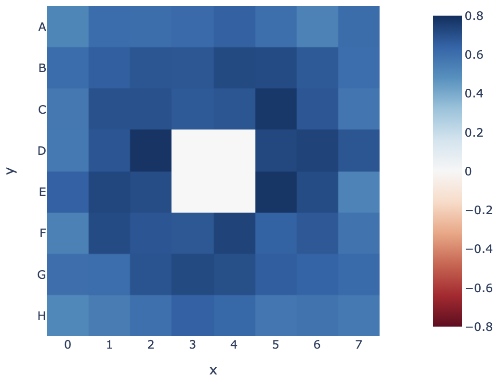
Follow-up evidence:
Calculate the cosine similarity by cell between “normalized blank direction” & “normalized my color direction” and embedding matrix(W_E) & unembedding matrix(W_U).

Result:

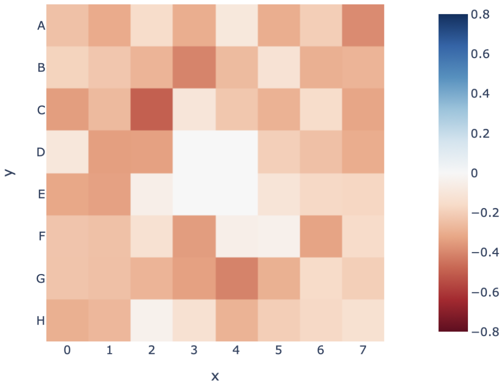
Cosine Similarity of Blank Probe and W_E



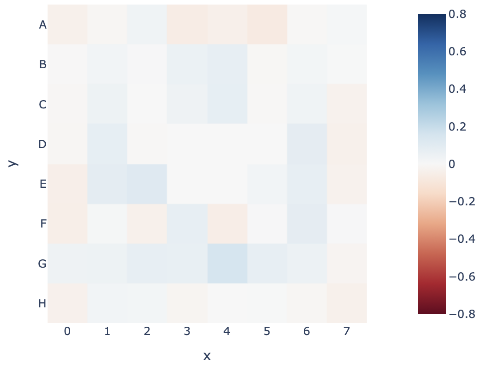
Cosine Similarity of Blank Probe and W_U



Cosine Similarity of My Probe and W_E



Cosine Similarity of My Probe and W_U



Section 2: Board state: Blank or Not Blank

Question: How and when does the model know the blank/not-blank board state (for all 64 cells)?

Key findings:

1. Layer 0 attention layer already knows (90% accuracy) the blank/not-blank board state.
2. Layer 1-4 attention layer increases the blank/not-blank classification accuracy (to 99%)
3. MLP layer seems not related to blank/not-blank classification.

Methods:

Use pre-trained linear probes to calculate the probabilities of “blank”, “theirs”, “mine” for all 64 cells on 50 example games based on different model representations. If the probability of “blank” is the largest among three options, we conclude that the model thinks that the cell is blank. Then map the model’s belief with the true blank/not-blank board state, and calculate the error rate for each cell.

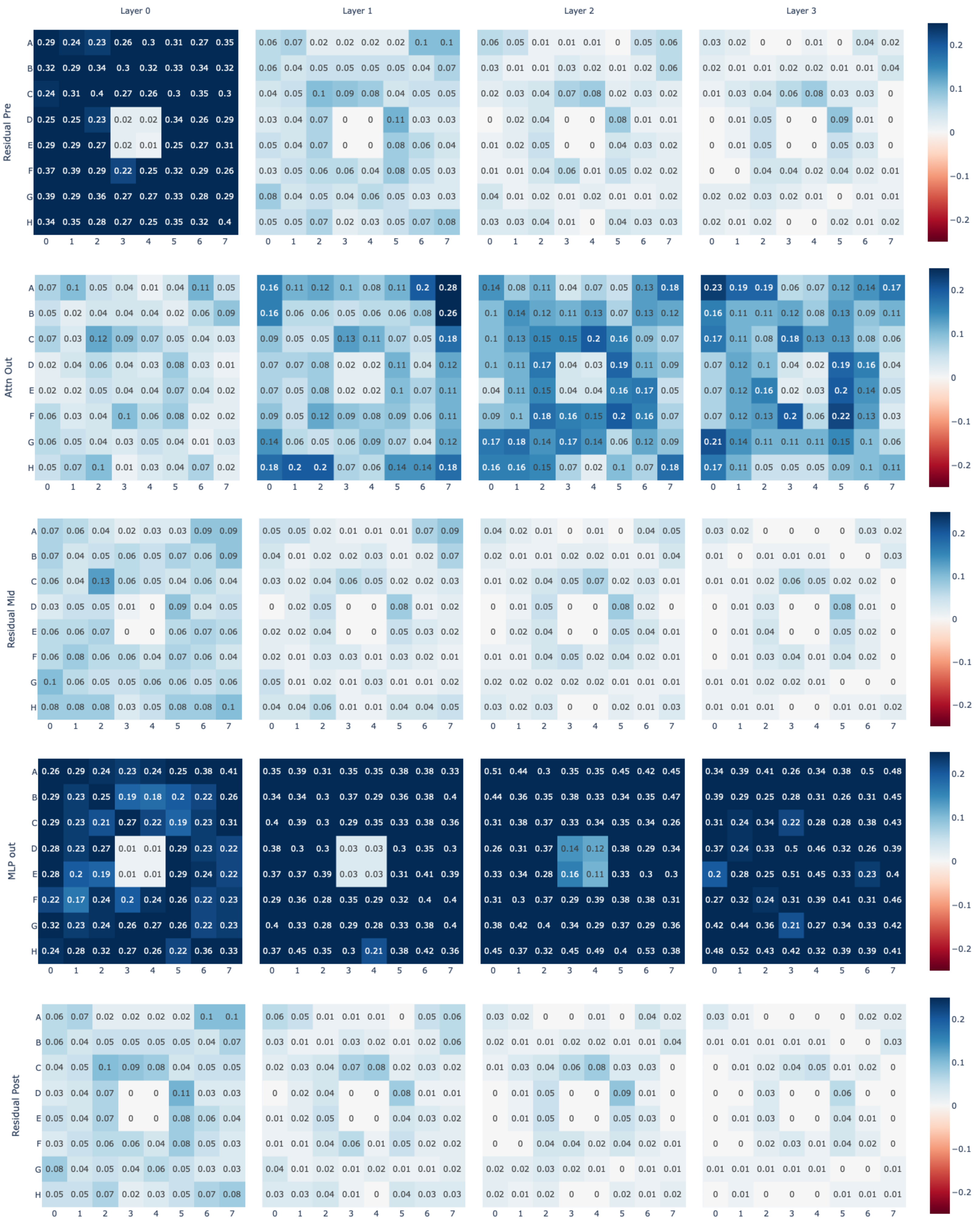
$$ErrorRate \; = \; 1 \; - \; Accuracy$$

$$Accuracy \;\; = \frac{1}{\#games \;\; \#moves} (\sum_{game=1}^{50} \sum_{move=3}^{59} (blank \;\; \neq \, argmax\{P(blank), \; P(theirs), \; P(mine)\}))$$

Observations and Hypothesis:

1. Look at the Residual Pre Layer 0 (left top corner), this is only using the information of residual pre (the model only knows what is the current move from token embedding, and the number of moves from the positional embedding). This shows that the model can form a good guess of the entire blank/not-blank board state with about 65-75% accuracy without any information of previous moves.
2. Layer 0 attn out is doing pretty good (90%-95% Accuracy). Layer 1- 3 attn out is helping to gradually improve the accuracy. Though looking at attn out itself seems it is not correlated with distinguishing between blank/not-blank. However, if we look at residual mid, the accuracy is increasing from layer 0 to layer 3. For example, the cell C2 has an error rate of 12% at Layer 0 Resid Mid, and decreases to 4% at Layer 1 Resid Mid. If we go to check Layer 1 Attn Out, we can see the error rate of C2 is low (5%). Note: it is larger than 4% may due to some information that attn layer is trying to calculate, resulting in some noise in blank/not-blank classification (may due to bias in blank probe that it includes some information of valid moves or not).
3. MLP layers seem not related to blank/not-blank classification. When using MLP Out to classify the blank/not-blank state, the error rate is similar to the “guessing” result (the left top corner, only use layer 0 residual pre to estimate blank/not-blank). So I think the MLP layers are doing something else (see Section 4).

Average Error Rate of Blank Probe (Blank vs Not Blank)



Section 3: Attention Heads

Question: What are attention heads doing? Are they distinguishing blank/not-blank state or

Key findings:

1. The main types of attention heads are “only attend to my previous moves” and “only attend to opponent’s previous moves”.
2. Many heads choose to strongly attend to the first move.

Method: Look at Attention Pattern
Plot average attention pattern(i.e. attention score after softmax) for 50 example games. For the sake of readability, we only show the results of the first ten moves.

Discussion:
Based on attention pattern, the behavior of heads can be classified into five categories:

1. Attend to my previous moves, for example: L0H3, L0H6, L1H5, L1H6

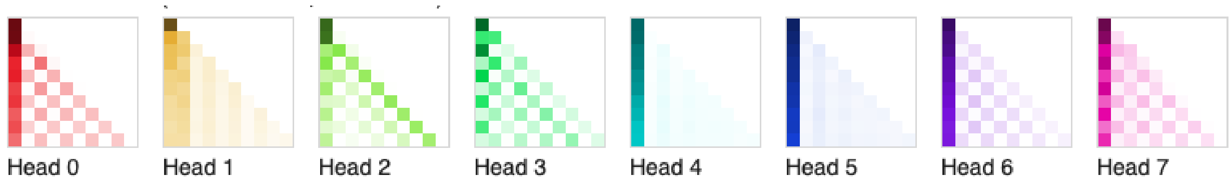
2. Attend to opponent’s previous moves, for example: L0H0, L0H2, L1H2, L1H3

3. Attend to the move before the current move: L4H5
   Which is my last move.

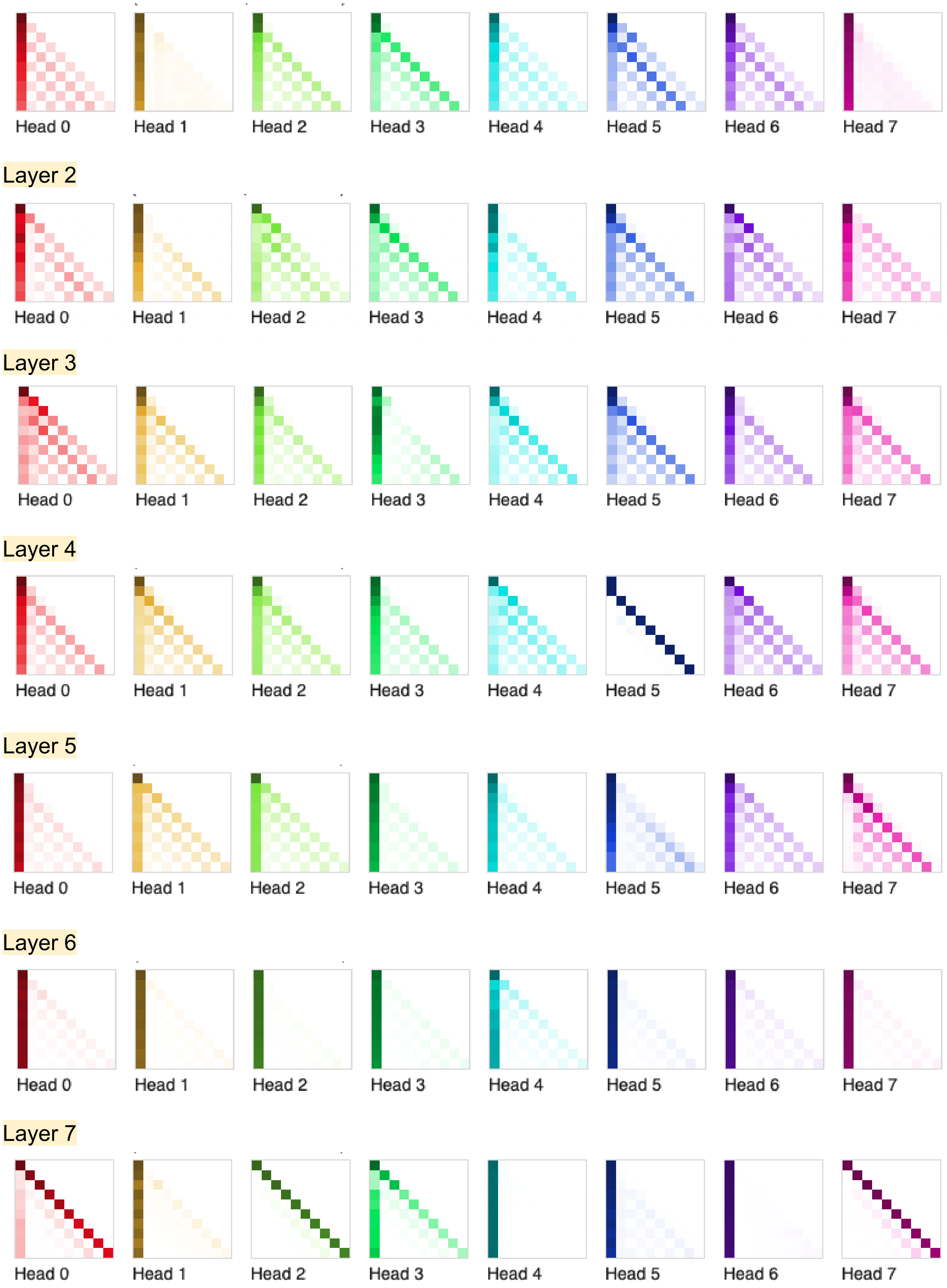
4. Attend to the current move: L7H0, L7H2, L7H3, L7H7
   This behavior is reasonable but I am confused because this only appears in the last layer. What can the model do in the last layer? It seems the model is working on the “flipping”(which needs information of the current move) in early MLP layers. Then why does the model need to know “what is the current move”? I tried doing zero ablation to the entire Layer 7, but the model just works fine (I only run it on a few games so might be bias)

5. Attend to the first move (surprisingly, all the heads in Layer 6 have this weird behavior).
   Hypothesis: the first position stores the least information (or can be treated as a constant since there are only 4 possible options for first move), so maybe the head just gives up producing any information.

Result:
Layer 0



Layer 1


Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 2
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 3
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 4
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 5
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 6
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7
Layer 7
Head 0
Head 1
Head 2
Head 3
Head 4
Head 5
Head 6
Head 7

Section 4: MLP Layers

Question: What are MLP layers doing?

Key findings:

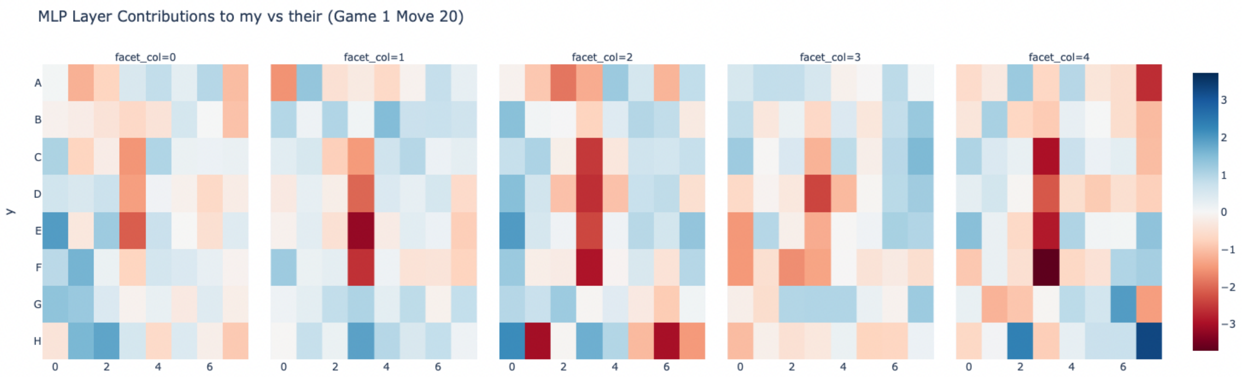
1. The first MLP layer already starts working on “flipping”(i.e. flip the “mine/theirs” state based on the board state and the current move).

Method:

These graphs are directly stolen from Neel Nanda’s Colab notebook. Which demonstrates the MLP layer contribution to “my” versus “their” color. Below is the corresponding board state.

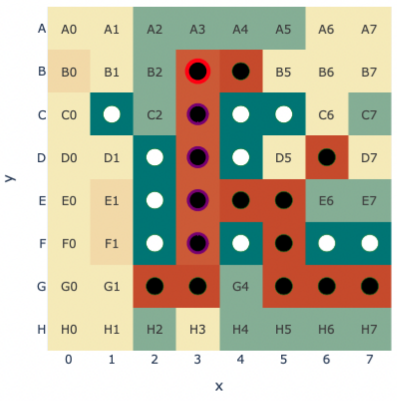
Discussion:

The current move is B3. And the cells being flipped(by the current move) are: C3, D3, E3, and F4. We can see that the MLP layer starts thinking about which cells to flip at the first layer, and is pretty sure at the second and third layer.



MLP Layer Contributions to my vs their (Game 1 Move 20)

White To Play. Board State After Black Plays B3



	0	1	2	3	4	5	6	7
A	A0	A1	A2	A3	A4	A5	A6	A7
B	B0	B1	B2	●	●	B5	B6	B7
C	C0	○	C2	●	○	○	C6	C7
D	D0	D1	○	●	○	D5	●	D7
E	E0	E1	○	●	●	●	E6	E7
F	F0	F1	○	●	○	●	○	○
G	G0	G1	●	●	G4	●	●	●
H	H0	H1	H2	H3	H4	H5	H6	H7