# Transformer needs NMDA receptor nonlinearity for long-term memory

**Anonymous Author(s)**
Affiliation
Address
`email`

## Abstract

The NMDA receptor (NMDAR) in the hippocampus is essential for learning and memory. We find an interesting resemblance between deep models' nonlinear activation function and the NMDAR's nonlinear dynamics. In light of a recent study that compared the transformer architecture to the formation of hippocampal memory, this paper presents new findings that NMDAR-like nonlinearity may be essential for consolidating short-term working memory into long-term reference memory. We design a navigation task assessing these two memory functions and show that manipulating the activation function (i.e., mimicking the $Mg^{2+}$-gating of NMDAR) disrupts long-term memory formation. Our experimental data suggest that the concept of place cells and reference memory may reside in the feed-forward network and that nonlinearity plays a key role in these processes. Our findings propose that the transformer architecture and hippocampal spatial representation resemble by sharing the overlapping concept of NMDAR nonlinearity.

## 1 Introduction

In the hippocampus, NMDAR is regarded as an essential component that mediates synaptic plasticity, memory formation, and spatial representation of place cells [9, 18, 6]. It has unique nonlinear dynamics which is modulated by $Mg^{2+}$-gating [13, 10], serving as a switch for synaptic plasticity and long-term memory formation [1, 17, 12] (Fig. 1a). This work is inspired by 1) the fascinating resemblance of NMDAR with the nonlinear GELU activation function that is widely used in the feed-forward networks of modern transformer architectures (Fig. 1c) [5, 4, 2] and 2) recent models relating transformer's self-attention mechanism to hippocampal formation [21, 20]. These findings motivated us to ask a question; **is the NMDAR-like nonlinearity in the feed-forward network of transformers required for long-term memory formation and spatial place cell representation?**

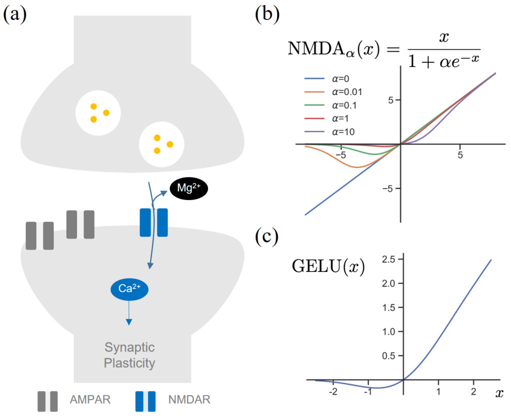

Figure 1: (a) Schematic diagram of $Mg^{2+}$-gated NMDAR modulating synaptic plasticity. (b) $Mg^{2+}$-gated NMDAR-like activation function. (c) Gaussian Error Linear Unit (GELU) activation function in transformer's feed-forward layers.

To address this question, we design a spatial navigation task in a 2D grid environment that assesses two different memory types in neuroscience [15, 16]: working memory and reference memory. Working memory controls the events from a within-trial, while reference memory controls across-trials from the unchanging environment. Our experimental data suggest that NMDAR-like nonlinearity in feed-forward networks of the transformer is essential for reference memory formation and place cell representation.

## 2 Methods

**Relating activation function in transformers with NMDAR nonlinearities** NMDAR's nonlinear dynamics arises from the voltage-gated $\text{Mg}^{2+}$ repulsion at the NMDAR channel's pore [13, 10] (Fig. 1a). Previously, $\text{Mg}^{2+}$-gated NMDAR open probability $\mathbf{p}$ has been shown to follow ion blockade model of A where $x$ represent an input voltage, $\alpha = [\text{Mg}^{2+}]/K_{\text{Mg}^{2+}}$ is a parameter determined by $[\text{Mg}^{2+}]$, $K_{\text{Mg}^{2+}}$ is a dissociation constant, and $\beta$ is a temperature constant. As experimentally shown, increasing the $\text{Mg}^{2+}$ level in the brain can enhance long-term memory formation [17]. We observed the NMDAR's nonlinear dynamics of the $IV$ curve (current-voltage relationship) in the synapse to closely resemble the form of the GELU activation function. GELU is a widely used activation function in transformers (Fig. 1c; $\text{GELU}(x) \approx x\sigma(1.702x)$ where $\sigma$ is the sigmoid function) [5, 4, 2]. Inspired by this resemblance, we define a new nonlinear activation function (Fig. 1b) with $\alpha$ parameter which modulates dynamics as follows:

$$\text{NMDA}_{\alpha}(x) = x\mathbf{p}_{\alpha}(x) = \frac{x}{1 + \alpha e^{-x}}. \tag{1}$$

To investigate this NMDAR-like nonlinearity in transformer memory formation, we replaced the $\text{GELU}(x)$ activation function with $\text{NMDA}_{\alpha}(x)$ in a standard transformer model.

**Transformers learn spatial navigation tasks** We train the transformer model to predict the subsequent sensory observation of an agent that randomly walks a 2D grid environment [20] (Fig. 2). A sequence of previous [Action ($a$), Observation ($x$)] pairs are an input to the model, and the subsequent observation is masked for prediction. Instead of using positional encoding [19] that is commonly used in transformers, we employ the recurrent neural network (RNN) for encoding the sequence of actions [20][1].

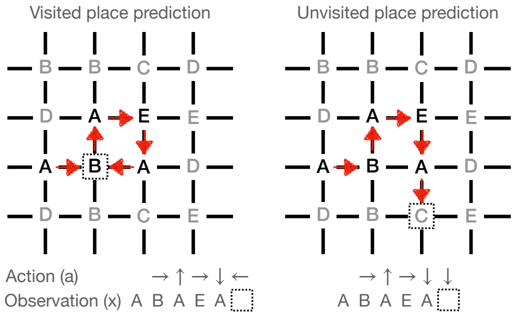

Figure 2: Sensory observation prediction task in a 2D grid, where dotted squares indicate the target position to predict given a sequence of past actions and observations. Gray (black) letters represent the unvisited (visited) places.

We generate the embedding vectors of sensory observation ($x$) sequence with a word embedding layer, but the embedding vectors of the action sequence is generated by RNN; $e_{t+1} = \tanh(e_t W_a)$, where $e_t$ is the positional embedding at step $t$, and $W_a$ is the action-dependent trainable weight matrix. The input is given by $\{[x_1, e_1], [x_2, e_2], \ldots, [x_t, e_t]\}$; the initial positional embedding $e_1$ is sampled from a normal distribution and we mask the last observation $x_t$. We generate $N$ maps of $11 \times 11$ 2D grids. A random sensory observation among ten letters is placed at each position on each map. Agents can move 'up', 'right', 'down', 'left', or 'stay'. An agent starts at a random position and initiates a random walk on the map for 2,048 steps for each trial.

The model is trained with the softmax cross-entropy loss and predicts the subsequent sensory observation (i.e., dotted squares). We evaluate two types of memory: **working memory (WM)** and **reference memory (RM)**[2]. When the prediction on nodes that were previously visited during the random walking is incorrect, it will count as a WM error (see Fig. 2 left). On the other hand, when the prediction on unvisited nodes is incorrect, it will count as a RM error (see Fig. 2 right). Minimizing the RM error by memorizing input sequences is infeasible; the possible number of

[1] Encoding actions with RNN is closely related to the state-of-the-art neuroscience model of hippocampus.

[2] Whittington et al. [20] only evaluated the WM error based on our definitions of WM and RM.

sequence configurations is exponential since the input sequence is randomly generated at each trial. To solve this task, the model should be able to 1) understand the abstract structure of 2D space, 2) infer which map it is on from input sequence data, and 3) memorize what sensory observation is placed at each position in that map. See Appendix A.1 for training, evaluation, and transformer model details.

## 3 Results

**WM error & RM error** The feed-forward network (FFN; see Fig. 4a) in the transformer model consists of two linear layers with the NMDAR-inspired activation function NMDA$_\alpha$ (Eq. (1)). To measure the impact of non-linearity $\alpha$ in FFNs, we train the transformer models with different values of $\alpha$ in $[0, 0.01, 0.05, 0.1, 0.5, 1, 5, 10]$ and evaluate WM and RM errors on the train maps (i.e., familiar maps) and test maps (i.e., novel maps).

Figure 3a shows that the RM error on the train maps is rapidly decreased over train trials when $\alpha$ is larger than zero, with a larger improvement for increasing $\alpha$. The RM error on the novel maps, however, is nearly constant at 0.9 ($= 1 - 1/$(number of letters)) for all $\alpha$. Unlike the RM, Fig. 3a inset shows that WM is performing well on novel maps, which had not been shown during the training. This finding suggests that RM is not used for predicting the visited nodes. Training the models on different numbers of maps $N$, Fig. 3b shows that increasing $\alpha$ helps improve RM and the trend of improvement is consistently shown for $N$ = 32, 48, and 64 cases. As $N$ grows, the RM error increases as more 'what'-'where' (letter-place) pairs have to be memorized.

**Place cells in FFNs** Place cell is a neuron in the hippocampus which fires at a particular place of the environment [14]. Selective impairment of NMDAR in hippocampal CA1 disrupts place cell emergence and long-term memory formation [18, 6, 11]. We investigate the role of neurons in FFNs and self-attention layers by measuring the neuron's place specificity. We measure the place cell score by defining a $K \times K$ 2D grid environment as graph $G = (V, E)$ and building a sub-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of all connected components from the source node $i_{\text{max}}$ where the neuron fires maximally; directed edges of sub-graph $\mathcal{G}$ are generated by connecting high to low firing nodes. We run depth-first-search from $i_{\text{max}}$. Given $G$ and $\mathcal{G}$, the place cell score is

$$\text{Place cell score} = \gamma \frac{\sum_{i \in \mathcal{V}} \rho_i}{\sum_{i \in V} \rho_i}, \tag{2}$$

where $\gamma = 1 - |\mathcal{V}^*|/|V|$ is a discount factor and $\mathcal{V}^*$ is a set of nodes from sub-graph without $i_{\text{max}}$ and leaf nodes during depth-first search. $\rho_i$ denotes a firing rate at node $i$. We record the firing rate $\rho_i$ of neurons over a random walking trajectory with $10^5$ steps in one of the training maps. Then we measure the place cell scores of neurons in FFNs and self-attention layers. The place cell score is 1 when the neuron is firing only at a certain node; the score is 0 when the neuron is firing homogeneously across all nodes.

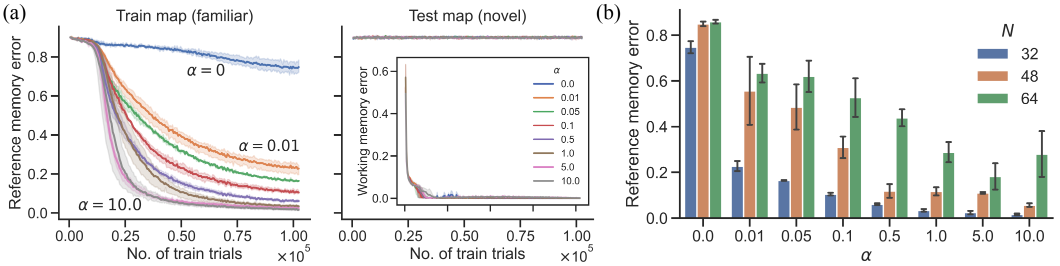

Figure 3: (a) Reference memory errors over training trials for training (familiar) maps and testing (novel) maps for $N = 32$ where $N$ is the number of training maps. Inset: working memory errors on the novel maps over training trials. (b) Reference memory errors over different values of $\alpha$ and $N$. Error bars and shaded areas represent the standard deviation of errors from three independently trained models.

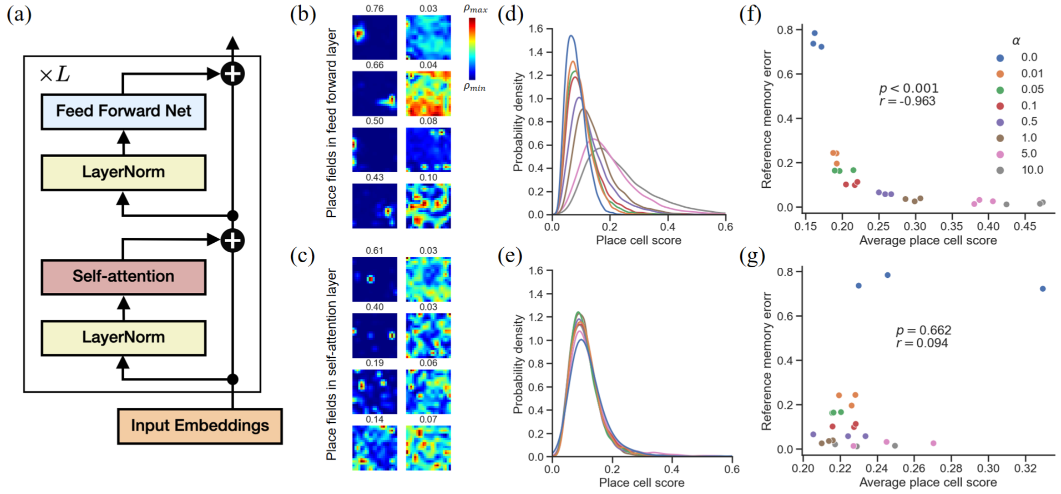

Figure 4: Reference memory-related place cells selectively emerge in the feed-forward layer but not in the self-attention layer along with $\alpha$ increase. (a) The transformer architecture used in the current study. (b, c) Example rate maps with place scores in feed-forward layers and self-attention layers at $\alpha = 10$; from top left (high) to bottom right (low) (d) Place cell score distribution in feed-forward layers change along with $\alpha$ modulation. (e) Place cell score distribution in self-attention layers does not change along with $\alpha$ modulation. (f-g) Scatter plot of average place cell scores and reference memory errors. $r$ and $p$ denote Spearman's rank correlation coefficient and significance score, respectively.

Fig. 4b and 4c show the rate maps of neurons with place cell scores in the FFNs and self-attention layers, respectively (Fig. 4a). As can be seen, our metric well represents place specificity. Fig. 4d and 4e show the distribution of place cell scores in FFNs and self-attention layers with different values of $\alpha$. As we increase $\alpha$, the place cell score distribution found in FNNs gets positively shifted (see Fig. 5 for rate maps for $\alpha = 0$, 1.0, and 10.0 in Appendix A.2), whereas place cell score distribution in the self-attention layers remains. In addition, Fig. 4f and 4g show a relationship between the average place cell score and RM error for each $\alpha$. While average place cell scores in the self-attention layer show no correlation with RM errors whatsoever, neurons in the FFN layer exhibit substantial correlation. These results imply that NMDAR-like nonlinearity in FFNs induces RM formation and the emergence of place cells.

## 4 Discussion and Conclusion

Whittington et al. [20] showed that softmax neurons in the self-attention layer behave like place cells and demonstrated that changing the softmax function to linear slows the WM learning process. However, the role of neurons in FFNs has not been studied. We demonstrate for the first time that place cells could emerge in transformers' FFNs, which we show by testing the emergence of place cells in FFNs with an NMDA-inspired activation function. Even though there are trainable parameters in the self-attention layer, the quantitative analysis of the place cell score indicates that most of the RM is stored in FFNs. Our results agree qualitatively with previous NMDAR impairment experiments from neuroscience: 1) hippocampal CA1 NMDAR perturbation does not impair WM [8], 2) changing NMDAR $Mg^{2+}$-gating (changing $\alpha$ in this work) enhances or disrupts long-term memory formation [17, 12], 3) NMDAR is required for long-term stabilization of newly forming place fields [11, 6]. Our contribution is at showing these patterns experimentally for the first time.

Our research has exciting future directions. The current study only examined what-where memory using a sensory observation task in a static environment. However, our real-world environment is changing dynamically. Unfortunately, modern deep learning systems are generally incapable of adapting to a dynamic environment or reordering sensory inputs. In future work, we intend to explore what-where-when memory, called *episodic memory*, in transformer and other deep models.

## References

[1] Tim VP Bliss and Graham L Collingridge. A synaptic model of memory: long-term potentiation in the hippocampus. *Nature*, 361(6407):31–39, 1993.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020. URL `https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf`.

[3] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Association for Computational Linguistics*, pages 2978–2988, 2019.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[6] Clifford Kentros, Eric Hargreaves, Robert D Hawkins, Eric R Kandel, Matthew Shapiro, and Robert V Muller. Abolition of long-term stability of new hippocampal place cell maps by nmda receptor blockade. *Science*, 280(5372):2121–2126, 1998.

[7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL `http://arxiv.org/abs/1412.6980`.

[8] Inah Lee and Raymond P Kesner. Differential contribution of nmda receptors in hippocampal subregions to spatial working memory. *Nature neuroscience*, 5(2):162–168, 2002.

[9] Fei Li and Joe Z Tsien. Memory and the nmda receptors. *The New England journal of medicine*, 361(3):302, 2009.

[10] Mark L Mayer, Gary L Westbrook, and Peter B Guthrie. Voltage-dependent block by mg2+ of nmda responses in spinal cord neurones. *Nature*, 309(5965):261–263, 1984.

[11] Thomas J McHugh, Kenneth I Blum, Joe Z Tsien, Susumu Tonegawa, and Matthew A Wilson. Impaired hippocampal representation of space in ca1-specific nmdar1 knockout mice. *Cell*, 87 (7):1339–1349, 1996.

[12] Tomoyuki Miyashita, Yoshiaki Oda, Junjiro Horiuchi, Jerry CP Yin, Takako Morimoto, and Minoru Saitoe. Mg2+ block of drosophila nmda receptors is required for long-term memory formation and creb-dependent gene expression. *Neuron*, 74(5):887–898, 2012.

[13] LPPAA Nowak, P Bregestovski, P Ascher, A Herbet, and Aa Prochiantz. Magnesium gates glutamate-activated channels in mouse central neurones. *Nature*, 307(5950):462–465, 1984.

[14] John O'Keefe and Jonathan Dostrovsky. The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain research*, 1971.

[15] David S Olton, Christine Collison, and Mary Ann Werz. Spatial memory and radial arm maze performance of rats. *Learning and motivation*, 8(3):289–314, 1977.

[16] David S Olton, James T Becker, and Gail E Handelmann. Hippocampus, space, and memory. *Behavioral and Brain sciences*, 2(3):313–322, 1979.

[17] Inna Slutsky, Nashat Abumaria, Long-Jun Wu, Chao Huang, Ling Zhang, Bo Li, Xiang Zhao, Arvind Govindarajan, Ming-Gao Zhao, Min Zhuo, et al. Enhancement of learning and memory by elevating brain magnesium. *Neuron*, 65(2):165–177, 2010.

[18] Joe Z Tsien, Patricio T Huerta, and Susumu Tonegawa. The essential role of hippocampal ca1 nmda receptor–dependent synaptic plasticity in spatial memory. *Cell*, 87(7):1327–1338, 1996.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[20] James C. R. Whittington, Joseph Warren, and Tim E.J. Behrens. Relating transformers to models and neural representations of the hippocampal formation. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=B8DVo9B1YE0`.

[21] James CR Whittington, Timothy H Muller, Shirley Mark, Guifen Chen, Caswell Barry, Neil Burgess, and Timothy EJ Behrens. The tolman-eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183(5):1249–1263, 2020.

## Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Abstract and Introduction.
   (b) Did you describe the limitations of your work? [Yes] See Discussion section.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] Yes, I read it and this paper conforms to them.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] All training, evaluation, and model details have been specified in the text. The code will be released with the camera-ready version.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix A.1
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We ran 3 different random seeds.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix A.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

# A Appendix

## A.1 Training, evaluation, and model configuration details

All runs used the same training method and model configuration except for the nonlinearity $\alpha$ of NMDA$_\alpha$ activation function. We used TransformerXL [3] with an extended memory length of 32 and segment length of 32 so that working memory error is measured within a sequence length of 65($= 64 + 1$; 1 for the masked sensory input); i.e. a node that the agent had never visited within recent 64 steps is treated as an unvisited node. The model consisted of two layers with a word embedding dimension of 256 and a positional embedding size of 256. The input embedding is concatenated vector $[x, e]$ of the word embedding $x$ and positional embedding $e$ so that the input embedding dimension is 512. The number of heads in the self-attention layer is 8 and the number of neurons in the feed forward net (FFN) is 2,048. The dropout rate is set to 0.1 and the maximum clip norm of gradient is set to 0.25. We employed ADAM [7] optimizer and a learning rate schedule with a linear decay from $0.0001$ (start) to $0$ (end). We ran 512 random walk simulations in parallel for collecting training trajectories. The total number of random walking steps is 2,048 for each simulation so the total number of gradient steps for each run was 512 (batch size) × 2,048 (total number of steps in a trial) × 200 (number of trials). All runs were performed on a single NVIDIA TITAN V GPU.

## A.2 Analysis details of place cell distribution in transformer

We plot each place cell score distribution with neurons from 3 independent experiments. For the self-attention layer, the total number of neurons in the softmax layer is 65 (number of sequence length) × 8 (number of head) × 2 (number of layers). For the feed-forward networks, the total number of neurons in the feed-forward layer is 2048 (number of neurons) × 2 (number of layers). Rate maps of neurons with top-64 place scores in FFNs with varying $\alpha$ are shown in Figure 5.

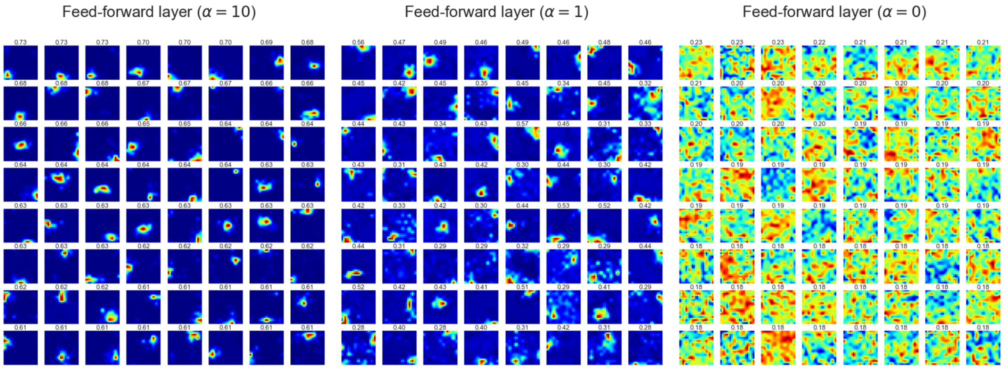

Figure 5: Rate maps of neurons with top-64 place scores in FFNs with varying values of $\alpha$; $\alpha = 10$ (left), $\alpha = 1$ (middle), and $\alpha = 0$ (right).