MODERN HOPFIELD NETWORKS WITH CONTINUOUS-TIME MEMORIES

Saul Santos [1,2], António Farinhas [1,2], Daniel C. McNamee [3], André F. T. Martins [1,2,4,5]
[1]Instituto de Telecomunicações, [2]Instituto Superior Técnico, Universidade de Lisboa,
[3]Champalimaud Research, [4]Unbabel , [5]ELLIS Unit Lisbon
{saul.r.santos,antonio.farinhas,andre.t.martins}@tecnico.ulisboa.pt,
daniel.mcnamee@research.fchampalimaud.org

ABSTRACT

Recent research has established a connection between modern Hopfield networks (HNs) and transformer attention heads, with guarantees of exponential storage capacity. However, these models still face challenges scaling storage efficiently. Inspired by psychological theories of continuous neural resource allocation in working memory, we propose an approach that compresses large discrete Hopfield memories into smaller, *continuous-time* memories. Leveraging continuous attention, our new energy function modifies the update rule of HNs, replacing the traditional softmax-based probability mass function with a probability *density* over the continuous memory. This formulation aligns with modern perspectives on human executive function, offering a principled link between attractor dynamics in working memory and resource-efficient memory allocation. Our framework maintains competitive performance with HNs while leveraging a compressed memory, reducing computational costs across synthetic and video datasets.

1 INTRODUCTION

Hopfield networks (Hopfield, 1982, HNs) are biologically inspired recurrent models that retrieve complete memories from partial cues through attractor dynamics, simulating episodic memory recall in humans and animals (Tyulmankov et al., 2021; Whittington et al., 2021). Classical HNs (Hopfield, 1982) store memories as fixed-point attractors, with storage capacity scaling linearly with the number of features. Recent research on **modern Hopfield networks** has introduced stronger nonlinearities in the update rule, enabling super-linear and even exponential storage capacity (Krotov & Hopfield, 2016; Ramsauer et al., 2020; Hu et al., 2023; Santos et al., 2024b;a). Despite these improvements, efficiency remains a challenge. While recent work has reduced retrieval complexity by reframing memory retrieval as a regression problem (Hu et al., 2024), and Hoover et al. (2024) recovered the compressed memory versions of classical HNs for dense associative models, it is still an open problem to increase the efficiency of Hopfield networks through **memory compression**—*how can we store information in a more compact form without sacrificing retrieval performance?*

Recent psychology research has characterized the human working memory processing based on **continuous neural resource allocation** (Ma et al., 2014). This theory posits that humans dynamically allocate neural activity across a set of stimuli or events to optimize their collective storage in a compressed format (Bays & Husain, 2008; Tomić & Bays, 2024). We suggest that this mechanism represents a continuous form of attention and contrasts sharply with traditional theories based on discrete memory "slots" (Miller, 1956) akin to discrete attention. Despite its empirical success in explaining human cognitive performance, this theory lacks a detailed recurrent neural network (RNN) implementation explaining memory-related population activity dynamics in prefrontal cortex (PFC) (Fuster & Alexander, 1971). While there exist task-optimized RNN models employing attractor dynamics for dynamic coding in memory tasks (Stroud et al., 2024), they lack the resource allocation mechanism central to neural resource theory. Inspired by such considerations, we sought to integrate compressive neural resource allocation with neural attractor dynamics within modern Hopfield networks, creating a novel framework for memory processing that bridges psychological and neuroscientific perspectives.

In this paper, we introduce a **continuous-time memory** mechanism within modern Hopfield networks, inspired by the continuous attention framework of Martins et al. (2020). We modify the energy function of Ramsauer et al. (2020), replacing discrete memory representations with a compressed, continuous alternative. We derive an update rule based on a probability density function (PDF) that links to Martins et al. (2022b)'s ∞-memory transformer. Experiments on synthetic and video datasets show that our approach achieves retrieval performance on par with modern HNs while using a smaller memory, paving the way for more memory-efficient associative models. [1]

2 Hopfield Networks

Hopfield networks perform associative recall over a set of memory patterns $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L \in \mathbb{R}^D$, stored in a memory matrix $\boldsymbol{X} \in \mathbb{R}^{L \times D}$. The network iteratively updates its state $\boldsymbol{q}^{(i)}$ to minimize an energy function, progressively converging toward one of the stored patterns, which serve as stable attractors. Classical Hopfield networks (Hopfield, 1982) minimize an energy function given by $E(\boldsymbol{q}) = -\frac{1}{2}\boldsymbol{q}^\top \boldsymbol{W}\boldsymbol{q}$, where $\boldsymbol{q}$ is the query vector and $\boldsymbol{W} = \boldsymbol{X}^\top \boldsymbol{X}$ is the weight matrix. The state update follows the rule: $\boldsymbol{q}^{(i+1)} = \text{sign}(\boldsymbol{W}\boldsymbol{q}^{(i)})$. However, the network's storage capacity scales only linearly with the input dimensionality, limiting effectiveness for large-scale memory tasks.

To address these limitations, modern Hopfield networks incorporate alternative energy functions that improve storage capacity beyond the classical limit (Krotov & Hopfield, 2016; Demircigil et al., 2017). A notable advancement in this domain was proposed by Ramsauer et al. (2020), who formulated a new energy function for continuous-valued states $\boldsymbol{q} \in \mathbb{R}^D$:

$$E(\boldsymbol{q}) = -\frac{1}{\beta} \log \sum_{i=1}^{L} \exp(\beta \boldsymbol{x}_i^\top \boldsymbol{q}) + \frac{1}{2}\|\boldsymbol{q}\|^2 + \text{const.} \tag{1}$$

A key insight from this formulation is the relationship between modern Hopfield networks and the attention mechanism used in transformers (Vaswani et al., 2017). Specifically, optimizing (1) via the concave-convex procedure (Yuille & Rangarajan, 2003, CCCP) leads to the update rule:

$$\boldsymbol{q}^{(i+1)} = \boldsymbol{X}^\top \text{softmax}(\beta \boldsymbol{X}\boldsymbol{q}^{(i)}), \tag{2}$$

By setting the scaling parameter to $\beta = \frac{1}{\sqrt{D}}$, this update rule recovers the single-head attention mechanisms in transformers, where identity matrices are used as projection layers. In the next section, we show that modern Hopfield networks can be combined with continuous attention to create more memory-efficient and computationally scalable architectures.

3 Continuous Attention

Traditional attention mechanisms operate on discrete representations, such as words in text or pixels in images. However, many data modalities, such as speech or video, are inherently continuous signals—their discretization with a fixed sampling rate might either be data-inefficient or may fail to exploit their smoothness. Continuous attention (Martins et al., 2020; 2022a) addresses this by defining attention over a continuous-time signal $\bar{\boldsymbol{x}}(t)$, treating the input as a representation function evolving smoothly over time. Instead of the probability mass functions having discrete attention weights, this approach leverages a probability density function $p(t)$, making it well-suited for long or unbounded temporal sequences like time series and audio. The attended output, or context $\boldsymbol{c}$, is computed as:

$$\boldsymbol{c} = \mathbb{E}_p[\boldsymbol{v}(t)] = \int p(t)\boldsymbol{v}(t)dt, \tag{3}$$

where $\boldsymbol{v}(t)$ denotes the continuous value function (a linear projection of $\bar{\boldsymbol{x}}(t)$). For modeling $p(t)$, Martins et al. (2022b) use a Gaussian $\mathcal{N}(t; \mu, \sigma^2)$, where μ and σ^2 are input-dependent while Santos et al. (2025) use uniformly spaced rectangular basis function.

[1] Our code is publicly available at `https://github.com/deep-spin/CHM-Net`.

4 Continuous-Time Memory Hopfield Networks

We assume memories form a continuum, and that sequences of observations $\boldsymbol{X} = [\boldsymbol{x}_1^\top, \dots, \boldsymbol{x}_L^\top] \in \mathbb{R}^{L \times D}$ are in fact samples from a smooth function $\boldsymbol{x}(t)$ that exists over a continuous domain. To reconstruct this function, we consider linear combinations of basis functions $\boldsymbol{\psi}(t) \in \mathbb{R}^N$ as

$$\bar{\boldsymbol{x}}(t) = \boldsymbol{B}^\top \boldsymbol{\psi}(t), \tag{4}$$

where $\boldsymbol{B} \in \mathbb{R}^{N \times D}$ denotes the learned coefficients. Usually, the number of basis functions is much smaller than the number of input samples, i.e., $N \ll L$, ensuring a compressed representation of the memory. The coefficient matrix $\boldsymbol{B}$ is derived through multivariate ridge regression (Brown & Zidek, 1980). Each observation in the sequence is assigned a corresponding time point $t_1, t_2, \dots, t_L$ normalized within the interval $[0, 1]$, where the sequence respects the ordering $t_1 \leq t_2 \leq \dots \leq t_L$ with $t_\ell \in [0, 1]$. The basis function evaluations at these time points define the design matrix $\boldsymbol{F} = [\boldsymbol{\psi}(t_1), \dots, \boldsymbol{\psi}(t_L)] \in \mathbb{R}^{N \times L}$. The coefficients $\boldsymbol{B}$ are computed such that $\boldsymbol{x}(t_\ell) \approx \boldsymbol{x}_\ell$ for each index $\ell \in \{1, \dots, L\}$, with a regularization parameter $\lambda > 0$. This leads to the closed-form solution:

$$\boldsymbol{B}^\top = \boldsymbol{X}^\top \boldsymbol{F}^\top (\boldsymbol{F}\boldsymbol{F}^\top + \lambda \boldsymbol{I})^{-1}. \tag{5}$$

We then define the **continuous Hopfield energy** using the reconstructed signal $\bar{\boldsymbol{x}}(t)$:

$$E(\boldsymbol{q}) = -\frac{1}{\beta} \log \int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q}) dt + \frac{1}{2} \|\boldsymbol{q}\|^2 + \text{const.} \tag{6}$$

This energy function encourages state patterns $\boldsymbol{q}$ to remain close to the linear combination of basis functions $\boldsymbol{\psi}(t)$ represented by the coefficient matrix $\boldsymbol{B}$, according to (4). Specifically, $\boldsymbol{\psi}(t)$ form a basis of "discrete memories," and $\boldsymbol{B}$ encodes the weights determining each function's contribution to $\boldsymbol{q}$. The second term is a quadratic regularization that penalizes large deviations, promoting stability.

The next result states the update rule corresponding to the energy (6).

Proposition 1. *Minimizing (6) using the CCCP algorithm (Yuille & Rangarajan, 2003) leads to the Gibbs expectation update, which is given by:*

$$\boldsymbol{q}^{(i+1)} = \mathbb{E}_{p(t)}[\bar{\boldsymbol{x}}(t)] = \boldsymbol{B}^\top \int p(t) \boldsymbol{\psi}(t)\, dt, \tag{7}$$

where $p(t)$ is the Gibbs density with temperature β^{-1}

$$p(t) = \frac{\exp(\beta s(t))}{\int \exp(\beta s(t'))\, dt'}, \tag{8}$$

with the continuous query-key similarity $s(t) = (\boldsymbol{q}^{(i)})^\top \bar{\boldsymbol{x}}(t) = (\boldsymbol{q}^{(i)})^\top \boldsymbol{B}^\top \boldsymbol{\psi}(t)$.

The proof is provided in Appendix A. In our experiments, the integrals in Proposition 1 are approximated with the trapezoidal rule.

5 Experiments

5.1 Hopfield Dynamics and Energy Contours

In this experiment, we illustrate the optimization trajectories and energy contours for various queries and 20 artificially pattern configurations, sampled from continuous functions, for HNs with both discrete and continuous memories, with $\beta = 1$. For the continuous memory version, we use 10 rectangular basis functions. Figure 1 shows comparable retrieval and energy behavior when memory points are sampled from the unit circumference (first and second columns). The final converged point does not correspond to a stored memory due to the dense nature of softmax and Gibbs PDF and the small value of β. This observation does not hold when the memory function is a line or sinusoid, as the third and fourth columns indicate that for continuous memories, the HN converges to a pattern closer to the initial query than its discrete memory counterpart. The clusters of energy minima, shown in darker blue, are also centered around a group with more points, indicating a more favorable energy landscape for HNs with continuous-time memories. Even when operating with reduced-dimensionality memory representations, continuous memories offer a stable and efficient convergence behavior, making them appealing to replace discrete memories in modern HNs.

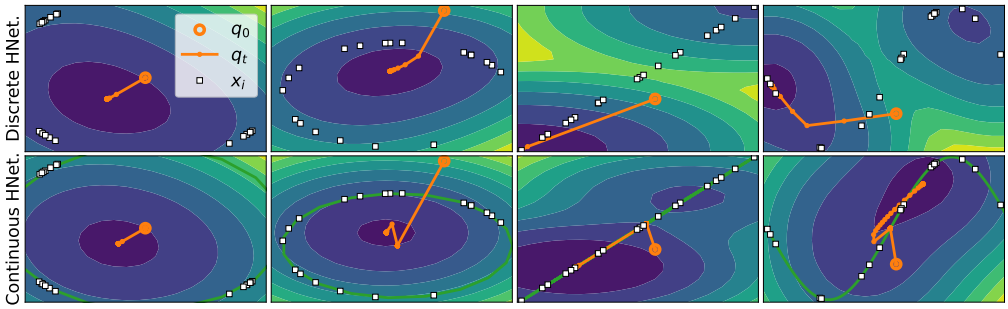

Figure 1: Optimization trajectories and energy contours for Hopfield networks with discrete **(top)** and continuous memories **(bottom)**. Green illustrates the continuous function shaped by discrete memory points, while darker shades of blue indicate lower energy regions.

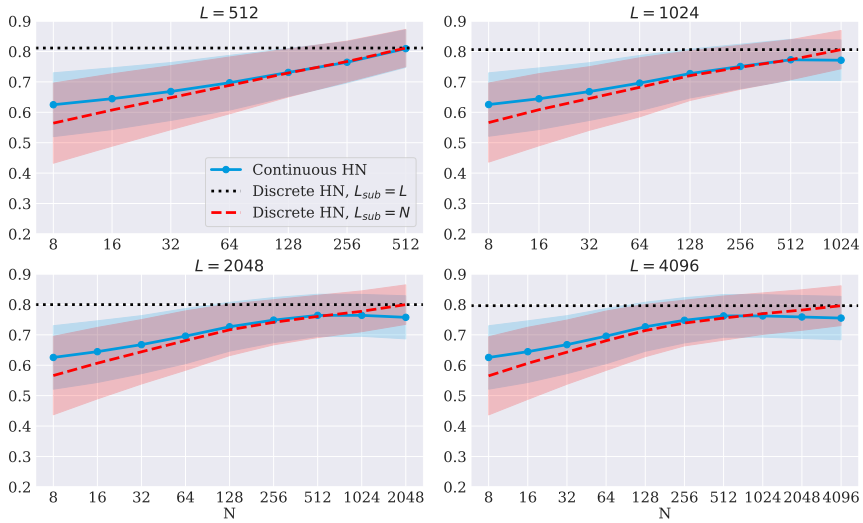

Figure 2: Video retrieval performance across different numbers of basis functions. Plotted are the cosine similarity means and standard deviations across videos.

5.2 Reconstruction of Video Frames

We now assess the performance of our models in video retrieval using the MovieChat-1K test set (Song et al., 2023), a benchmark of 100 long videos, each averaging 8 minutes. Our focus is on evaluating how HNs with continuous memories perform in retrieving frames from memory sequences of varying lengths, compared to traditional HNs with discrete memories. To do this, we subsample L frames from each video at a resolution of 224×224 pixels and normalize the pixel values to a range of $[-1, 1]$. Each video is treated as a unique memory. Subsequently, we query both variants of HNs using these memories, with the lower half of each frame masked to 0.

Figure 2 presents the mean and standard deviation of the cosine similarities between the memories and the retrieved patterns for the HN with continuous memories, across varying values of rectangular basis functions. For the HN with discrete memories, we subsample different numbers of frames, L_{sub}, from the total L, ensuring a fair comparison between both models. Our results indicate that for smaller memories, such as $L = 512$, the continuous HN consistently outperforms the discrete HN, achieving comparable performance when $N = L$. As N increases, and for larger memory sizes, the continuous HN maintains this trend, although we observe a degradation in performance as N approaches L. We hypothesize that this degradation is due to the discrete representation of the queries (*i.e.*, pixel-level representations), which favors the discrete HN, as both the query and memory are discrete. In the following section, we demonstrate that when the reconstruction is performed over continuous domains, such as embeddings, our model shows improvements.

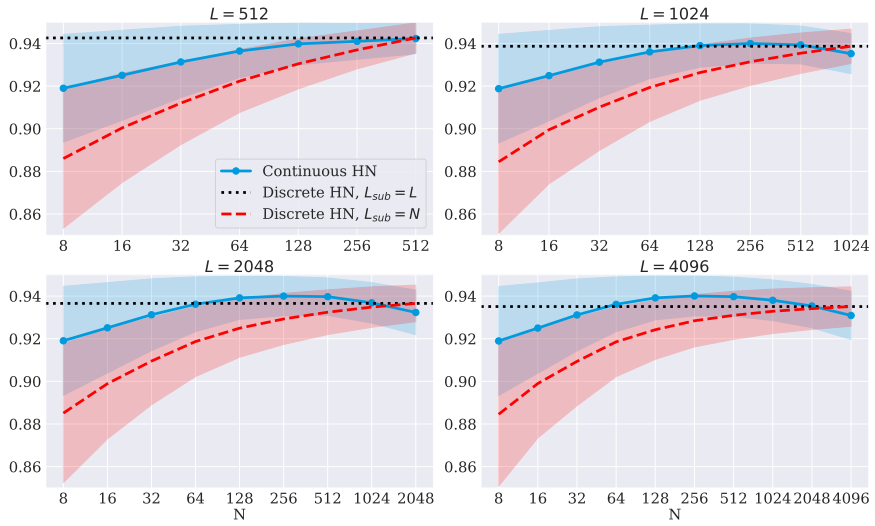

Figure 3: Video embedding retrieval performance across different numbers of basis functions. Plotted are the cosine similarity means and standard deviations across videos.

5.3 Reconstruction of Video Embeddings

Next, we present the performance of our method in domains with inherently continuous representations, such as video embeddings. We apply the same pre-processing mentioned earlier; however, we now pass the frames to a visual encoder, specifically EVA-CLIP's ViT-G/14 (Fang et al., 2022), followed by a Q-former (Li et al., 2023) that extracts 32 tokens per frame while considering the spatial structure in the frames. To ensure smoothness and continuity, we perform average pooling over the 32 tokens, obtaining L representations. We then add Gaussian noise with $\sigma = 5$ to the memories and use them as queries. Figure 3 illustrates the results of embedding reconstruction, emphasizing the consistently superior performance of our HN for larger memories, even surpassing the discrete HN for the full memory $L_{\text{sub}} = L$ but using $N \ll L$, except for $L = 512$. This superiority tends to increase as the number of memories L grows, suggesting that our approach could be a promising direction for improving efficiency in modern HNs. We present ablation studies in Appendix B.

6 Conclusions and Discussion

We introduced an alternative formulation of Hopfield networks that employs a continuous-time memory, offering a more efficient representation without sacrificing retrieval performance. This approach extends the connection between Hopfield networks and transformer-based architectures such as the ∞-former (Martins et al., 2022b) by replacing discrete memories with continuous representations. Our new energy function introduces an update rule grounded in the continuous attention framework, where the query is iteratively updated according to a Gibbs probability density function over the continuous reconstructed signal. Memory recall experiments on synthetic and video datasets indicate that our continuous memory formulation achieves retrieval performance on par with discrete Hopfield networks, showcasing its potential for scalable memory-augmented models.

Despite the promising results when $N \ll L$, we observed performance degradation when the number of basis functions approaches the length of the discrete memory. We hypothesize that this stems from the rigid allocation of uniformly spaced rectangular functions, which may not optimally capture the memory's underlying structure. Future research should explore adaptive mechanisms, such as neural approaches for dynamically learning the widths and centers of basis functions, enabling the model to focus on important regions of the discrete signal. Additionally, replacing the multivariate ridge regression step with a neural network could improve expressiveness and adaptivity in memory modeling. The impact of continuous memories on storage capacity also requires further investigation. Future work will assess the applicability of our approach in real-world problems by exploring the integration of learnable Hopfield layers with continuous memories in practical applications.

ACKNOWLEDGMENTS

We thank Sweta Agrawal, Patrick Fernandes, and the SARDINE lab team for helpful discussions. This work was supported by EU's Horizon Europe Research and Innovation Actions (UTTER, contract 101070631), by the project DECOLLAGE (ERC-2022-CoG 101088763), by the Portuguese Recovery and Resilience Plan through project C645008882-00000055 (Center for Responsible AI), and by FCT/MECI through national funds and when applicable co-funded EU funds under UID/50008: Instituto de Telecomunicações.

REFERENCES

Paul M Bays and Masud Husain. Dynamic shifts of limited working memory resources in human vision. *Science*, 321(5890):851–854, 2008.

Philip J Brown and James V Zidek. Adaptive multivariate ridge regression. *The Annuals of Statistics*, 1980.

Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168(2):288–299, May 2017. ISSN 1572-9613. doi: 10.1007/s10955-017-1806-y. URL http://dx.doi.org/10.1007/s10955-017-1806-y.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. 2022.

Joaquin M. Fuster and Garrett E. Alexander. Neuron activity related to short-term memory. *Science*, 173(3997):652–654, 1971. doi: 10.1126/science.173.3997.652.

Benjamin Hoover, Duen Horng Chau, Hendrik Strobelt, Parikshit Ram, and Dmitry Krotov. Dense associative memory through the lens of random features. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=164QnJsYjF.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2309.12673.

Jerry Yao-Chieh Hu, Bo-Yu Chen, Dennis Wu, Feng Ruan, and Han Liu. Nonparametric modern hopfield models, 2024. URL https://arxiv.org/abs/2404.03900.

Dmitry Krotov and John J. Hopfield. Dense associative memory for pattern recognition. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/eaae339c4d89fc102edd9dbdb6a28915-Paper.pdf.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Wei Ji Ma, Masud Husain, and Paul M Bays. Changing concepts of working memory. *Nature Neuroscience*, 17(3):347–356, 2014. doi: 10.1038/nn.3655. URL https://doi.org/10.1038/nn.3655.

André Martins, António Farinhas, Marcos Treviso, Vlad Niculae, Pedro Aguiar, and Mario Figueiredo. Sparse and continuous attention mechanisms. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 20989–21001. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f0b76267fbe12b936bd65e203dc675c1-Paper.pdf.

André F. T. Martins, Marcos Treviso, António Farinhas, Pedro M. Q. Aguiar, Mário A. T. Figueiredo, Mathieu Blondel, and Vlad Niculae. Sparse continuous distributions and fenchel-young losses. *Journal of Machine Learning Research*, 23(257):1–74, 2022a. URL http://jmlr.org/papers/v23/21-0879.html.

Pedro Henrique Martins, Zita Marinho, and André FT Martins. ∞-former: Infinite memory transformer. In *Proc. ACL*, 2022b.

George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97, 1956.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Saul Santos, Vlad Niculae, Daniel McNamee, and André F. T. Martins. Hopfield-fenchel-young networks: A unified framework for associative memory retrieval. *arXiv preprint arXiv:2411.08590*, 2024a. URL https://arxiv.org/abs/2411.08590.

Saul Santos, Vlad Niculae, Daniel C McNamee, and Andre F.T. Martins. Sparse and structured hopfield networks. In *International Conference on Machine Learning*, 2024b.

Saul Santos, António Farinhas, Daniel C. McNamee, and André F. T. Martins. ∞-video: A training-free approach to long video understanding via continuous-time memory consolidation. *arXiv preprint arXiv:2501.19098*, 2025. URL https://arxiv.org/abs/2501.19098.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Xun Guo, Tian Ye, Yan Lu, Jenq-Neng Hwang, et al. Moviechat: From dense token to sparse memory for long video understanding. *arXiv preprint arXiv:2307.16449*, 2023.

Jake P. Stroud, John Duncan, and Máté Lengyel. The computational foundations of dynamic coding in working memory. *Trends in Cognitive Sciences*, 28(7):614–627, July 2024.

Ivan Tomić and Paul M Bays. A dynamic neural resource model bridges sensory and working memory. *eLife*, 12:RP91034, may 2024. ISSN 2050-084X. doi: 10.7554/eLife.91034. URL https://doi.org/10.7554/eLife.91034.

Danil Tyulmankov, Ching Fang, Annapurna Vadaparty, and Guangyu Robert Yang. Biological learning in key-value memory networks. In *Advances in Neural Information Processing Systems*, 2021.

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.

James CR Whittington, Joseph Warren, and Tim EJ Behrens. Relating transformers to models and neural representations of the hippocampal formation. In *International Conference on Learning Representations*, 2021.

Alan L Yuille and Anand Rangarajan. The concave-convex procedure. *Neural computation*, 15(4): 915–936, 2003.

A Proof of Proposition 1

The CCCP algorithm works as follows: at the tth iteration, it linearizes the concave function E_{concave} by using a first-order Taylor approximation around $\boldsymbol{q}^{(t)}$,

$$E_{\text{concave}}(\boldsymbol{q}) \approx \tilde{E}_{\text{concave}}(\boldsymbol{q}) := E_{\text{concave}}(\boldsymbol{q}^{(t)}) + \left(\frac{\partial E_{\text{concave}}(\boldsymbol{q}^{(t)})}{\partial \boldsymbol{q}}\right)^{\top} (\boldsymbol{q} - \boldsymbol{q}^{(t)}). \tag{9}$$

Then, it computes a new iterate by solving the convex optimization problem

$$\boldsymbol{q}^{(i+1)} := \arg\min_{\boldsymbol{q}} E_{\text{convex}}(\boldsymbol{q}) + \tilde{E}_{\text{concave}}(\boldsymbol{q}), \tag{10}$$

which leads to the equation

$$\nabla E_{\text{convex}}(\boldsymbol{q}^{(i+1)}) = -\nabla E_{\text{concave}}(\boldsymbol{q}^{(i)}). \tag{11}$$

Using the fact that $E_{\text{concave}}(\boldsymbol{q})$ and $E_{\text{convex}}(\boldsymbol{q})$ are defined as

$$E_{\text{concave}}(\boldsymbol{q}) = -\frac{1}{\beta} \log \int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q}) dt, \tag{12}$$

$$E_{\text{convex}}(\boldsymbol{q}) = \frac{1}{2} \boldsymbol{q}^\top \boldsymbol{q}, \tag{13}$$

we compute their gradients:

$$\nabla E_{\text{concave}}(\boldsymbol{q}) = -\frac{1}{\beta} \nabla \log \int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q}) \tag{14}$$

$$\nabla E_{\text{convex}}(\boldsymbol{q}) = \boldsymbol{q}. \tag{15}$$

Using the chain rule for the E_{concave}, we get:

$$\nabla E_{\text{concave}}(\boldsymbol{q}) = -\frac{1}{\beta \int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt} \nabla \left(\int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt \right).$$

Next, we compute the gradient of the integral. Since the integral is with respect to t, we differentiate under the integral sign:

$$\nabla \left(\int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt \right) = \int_0^1 \beta \bar{\boldsymbol{x}}(t) \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt.$$

Therefore, the gradient of $E_{\text{concave}}(\boldsymbol{q})$ becomes:

$$\nabla E_{\text{concave}}(\boldsymbol{q}) = -\frac{\int_0^1 \bar{\boldsymbol{x}}(t) \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt}{\int_0^1 \exp(\beta \bar{\boldsymbol{x}}(t)^\top \boldsymbol{q})\, dt} = -\int_0^1 p(t) \bar{\boldsymbol{x}}(t) dt,$$

where $p(t)$ is the Gibbs probability density function. Now, recognizing that the above expression represents the expectation with respect to the distribution $p(t)$, we write:

$$\nabla E_{\text{concave}}(\boldsymbol{q}) = -\mathbb{E}_{p(t)}[\beta \bar{\boldsymbol{x}}(t)].$$

Thus, the update equation is given by

$$\boldsymbol{q}^{(i+1)} = \mathbb{E}_{p(t)}[\bar{\boldsymbol{x}}(t)], \tag{16}$$

which corresponds to the Gibbs expectation update.

B Ablation Studies

We present in Figure 4 the average cosine similarities as a function of the number of points used to approximate the integrals from Proposition 1. For the video dataset with 50% masked frames, we use $L = 512$ and $N = 512$. For embedding reconstruction, we set $L = 2048$ and $N = 1024$ with $\sigma = 5$. The results demonstrate that 500 sampling points are sufficient for the approximation, where the Hopfield network with continuous memories performs comparably to the modern Hopfield network.

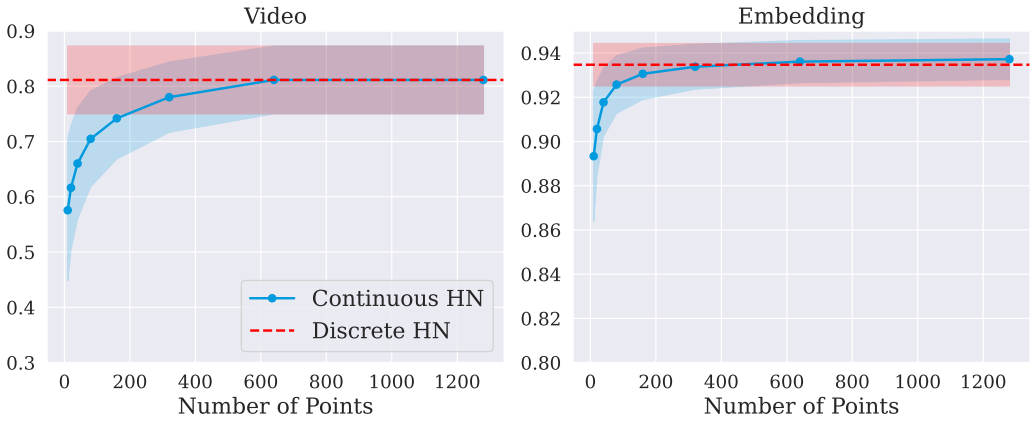

Figure 4: Performance on video and embedding data across different numbers of sampling points used to approximate the integrals of our framework.