On Sparse Modern Hopfield Model

Jerry Yao-Chieh Hu[†] Donglin Yang[†] Dennis Wu[†]
Chenwei Xu[†] Bo-Yu Chen[‡] Han Liu[†♮]

[†]Department of Computer Science, Northwestern University, Evanston, IL 60208 USA
[‡]Department of Physics, National Taiwan University, Taipei 10617, Taiwan
[♮]Department of Statistics and Data Science, Northwestern University, Evanston, IL 60208 USA

{jhu, dlyang, hibb, cwxu}@u.northwestern.edu
b12202023@ntu.edu.tw, hanliu@northwestern.edu

Abstract

We introduce the sparse modern Hopfield model as a sparse extension of the modern Hopfield model. Like its dense counterpart, the sparse modern Hopfield model equips a memory-retrieval dynamics whose one-step approximation corresponds to the sparse attention mechanism. Theoretically, our key contribution is a principled derivation of a closed-form sparse Hopfield energy using the convex conjugate of the sparse entropic regularizer. Building upon this, we derive the sparse memory retrieval dynamics from the sparse energy function and show its one-step approximation is equivalent to the sparse-structured attention. Importantly, we provide a sparsity-dependent memory retrieval error bound which is provably tighter than its dense analog. The conditions for the benefits of sparsity to arise are therefore identified and discussed. In addition, we show that the sparse modern Hopfield model maintains the robust theoretical properties of its dense counterpart, including rapid fixed point convergence and exponential memory capacity. Empirically, we use both synthetic and real-world datasets to demonstrate that the sparse Hopfield model outperforms its dense counterpart in many situations. [September 25, 2023]

1 Introduction

We address the computational challenges of modern Hopfield models by introducing a sparse Hopfield model. Our sparse continuous Hopfield model equips a memory-retrieval dynamics that aligns with the sparse-structured attention mechanism. By establishing a connection to sparse attention, the proposed model not only offers a theoretically-grounded energy-based model for associative memory but also enables robust representation learning and seamless integration with deep learning architectures. This approach serves as an initial attempt of pushing the correspondence[1] between Hopfield models and attention mechanism [Ramsauer et al., 2021] toward sparse region, both theoretically and empirically, resulting in data-dependent sparsity for meaningful and robust pattern representations, and a focus on the most relevant information for each specific instance.

Hopfield models are classic associative memory models for both biological and artificial neural networks [Hopfield, 1982, 1984]. These models are designed to store and retrieve memory patterns[2]. They achieve these by embedding the memories in the energy landscape of a physical system (e.g., the Ising model in [Hopfield, 1982, Peretto and Niez, 1986]; see Figure 3 for a visualization), where

[1] While this equivalence only holds when the retrieval dynamics is applied exactly once, as originally shown in [Ramsauer et al., 2021] and later emphasized in [Krotov and Hopfield, 2021], it allows us to view modern Hopfield models as generalized attentions with additional functionalities and hence opens new avenues for Hopfield-based architecture designs. See Appendix C for more discussions.

[2] For instance, if the stored memories are images of all the dogs you've seen in the past, and the query is the image of a dog you see today, the Hopfield model retrieves the memory of the dog that most closely resembles the one you saw today.

Preprint. Under review.

each memory corresponds to a local minimum. When a query is presented, the model initiates energy-minimizing retrieval dynamics at the query, which then navigate the energy landscape to find the nearest local minimum, effectively retrieving the memory most similar to the query.

In the same vein, Ramsauer et al. [2021] propose the modern Hopfield model and integrate it into deep learning architectures via a strong connection with transformer attention, offering enhanced performance, theoretically guaranteed exponential memory capacity, and the ability to handle continuous patterns. In addition, the modern Hopfield models have found success in various applications, such as immunology [Widrich et al., 2020] and large language model [Fürst et al., 2022]. Apart from the elegant connection to attention, theoretical advantages and empirically successes, the modern Hopfield models have been shown to be computationally heavy and vulnerable against noisy queries [Millidge et al., 2022]. In particular, the dense output alignments of the retrieval dynamics in modern Hopfield models [Ramsauer et al., 2021] can be computationally inefficient, making models less interpretable and noise-sensitive by assigning probability mass to many implausible outputs (patterns/keys).

To combat above, incorporating sparsity is an essential and common strategy. While there is a vast body of work on sparsifying attention mechanisms [Tay et al., 2022, Beltagy et al., 2020, Qiu et al., 2019, Child et al., 2019, Peters et al., 2019, Martins and Astudillo, 2016], similar developments for the Hopfield models remain less explored. To bridge this gap, we present a sparse Hopfield model that corresponds to the sparsemax attention mechanism [Martins and Astudillo, 2016]. In this paper, we study the sparsification of the modern Hopfield model. The challenges are three-fold:

(C1) **Non-Trivial Sparsification — Sparse Hopfield $\leftrightarrow$ Sparse Attention:** To enable the use of sparse Hopfield models as computational devices (DNN learning models) akin to [Ramsauer et al., 2021], it is essential to achieve *non-trivial* sparsifications that exhibit equivalence to specific sparse attention models. In other words, any meaningful sparsification should extend the established equivalence [Ramsauer et al., 2021] between modern Hopfield models and attention to encompass the sparse domain. While generalizing such equivalence is potentially impactful as it may lay the groundwork for future Hopfield-based methodologies, architecture designs and bio-computing systems (as in [Kozachkov et al., 2023]), the *heuristic* design of the modern Hopfield model poses great difficulty to developing desired sparse models.

(C2) **Introducing Sparsity into Hopfield Models:** Unlike attention mechanisms where sparsification is typically achieved either on the attention matrix (e.g., structured-sparsity [Tay et al., 2020, Child et al., 2019]) or on the element-wise normalization map (e.g., sparsity-inducing maps [Correia et al., 2019, Peters et al., 2019, Martins and Astudillo, 2016]), the sparsification of Hopfield models is applied to *both* the energy function and the memory-retrieval dynamics, where the latter monotonically decreases the Hopfield energy over time. Since attention mechanisms (transformers) are typically not equipped with such a dynamical description, introducing sparsity into Hopfield models while retaining the connection to attention is a less straightforward process.

(C3) **Properties of the Sparse Hopfield Model:** Further, it is unclear how the introduced sparsity may affect different aspects of the model, such as memory capacity, fixed point convergence, retrieval accuracy, and so on. Ideally, we are looking for sparsities that offer provable computational benefits, such as enhanced robustness and increased memory capacity, among others.

Challenges (C1) and (C2) are inherent in Hopfield model, and certain requirements on the design of energy function and retrieval dynamics are inevitable to obtain non-trivial sparse models. Hence, we suppose the sparsified models should satisfy some conditions and verify them accordingly. Concretely, a formulation for deriving desired sparse Hopfield energy via convex conjugation of entropic regularizers is proposed. Furthermore, by applying Danskin's theorem and convex-concave procedure [Yuille and Rangarajan, 2003, 2001] on the sparse Hopfield energy function, we obtain sparse retrieval dynamics linked to sparse attention. For (C3), the convergence of energy stationary points and retrieval dynamics fixed points are connected via Zangwill's method [Zangwill, 1969]. The sparse retrieval error bound is derived and used to determined the well-separation condition for successful memory storage and retrieval. Lastly, the fundamental limit of memory capacity is derived using the expected separation of random points on spheres [Cai and Jiang, 2012, Brauchart et al., 2018, Ramsauer et al., 2021].

In summary, this work handles sparsification of modern Hopfield models while linking them to sparse attention by addressing the following question:

Is it possible to develop a theoretically-grounded (non-trivial) sparse Hopfield model capable of storing information or learned prototypes throughout various layers of DNN models?

Contributions. We propose the Sparse Modern Hopfield Model. Our contributions are as follows:

- We propose a novel sparse Hopfield model whose retrieval dynamics corresponds to sparsemax attention mechanism. It leads to sparse patterns by design, inheriting both noise robustness and potential computational efficiency[3] from [Martins and Astudillo, 2016], compared to its dense counterparts. This work extends the theoretical understanding of the correspondence between artificial and biological neural networks to sparse region. In addition, the sparse Hopfield layer, a new deep learning component, is introduced with data-dependent sparsity.
- Theoretically, we establish provably advantages from sparsity and identify the conditions under which these benefits arise. We begin by deriving the closed-form sparse Hopfield energy from the convex conjugation of sparse entropic regularizer. Next, we demonstrate the correspondence between sparse Hopfield retrieval dynamics and sparsemax attention. In addition, we prove the fast convergence of the fixed points (also known as memory patterns, attractor states in literature) for the retrieval dynamics and establish the exponential (in pattern size) memory capacity lower bound with *tighter* retrieval error bound, *compared* with modern Hopfield models.
- Empirically, we conduct synthetic and realistic experiments to verify our theoretical results and proposed methodology. Specifically, the sparse Hopfield model outperforms the dense Hopfield model and machine learning baselines in *sparse* Multiple Instance Learning (MIL), time series prediction and neural machine translation problems. This is observed with both *sparse* synthetic and real-world datasets, where the baselines tend to fall short. Moreover, even in cases without data sparsity, our proposed model delivers performance on par with its dense counterpart.

To the best of our knowledge, we are the first to propose a sparse Hopfield model whose retrieval dynamics is equivalent to sparse attention mechanism with provably computational advantages. Methodologically, the proposed model complements existing Hopfield-based DNN architectures [Hoover et al., 2023, Paischer et al., 2022, Seidl et al., 2022, Fürst et al., 2022, Ramsauer et al., 2021] by introducing a sparse Hopfield layer into deep learning models.

Organization. In Section 2, the sparse Hopfield model is introduced. In Section 3, the memory capacity is discussed. In Section 4, experimental studies are conducted. In Section 5, concluding discussions are provided. Additionally, related works and limitations are discussed in Appendix C.

Notations. We write $\langle \mathbf{a}, \mathbf{b} \rangle \coloneqq \mathbf{a}^\mathsf{T}\mathbf{b}$ as the inner product for vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$. The index set $\{1, \cdots, I\}$ is denoted by $[I]$, where $I \in \mathbb{N}_+$. The spectral norm is denoted by $\|\cdot\|_2$, which is equivalent to the l_2-norm when applied to a vector. Throughout this paper, we denote the memory patterns (keys) by $\boldsymbol{\xi} \in \mathbb{R}^d$ and the state/configuration/query pattern by $\mathbf{x} \in \mathbb{R}^d$, and $\boldsymbol{\Xi} \coloneqq (\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_M) \in \mathbb{R}^{d\times M}$ as shorthand for stored memory (key) patterns $\{\boldsymbol{\xi}_\mu\}_{\mu\in[M]}$. Moreover, we set norm $n \coloneqq \|\mathbf{x}\|$ be the norm of the query pattern, and $m \coloneqq \mathrm{Max}_{\mu\in[M]} \|\boldsymbol{\xi}_\mu\|$ be the largest norm of memory patterns. We also provide a nomenclature table (Table 3) in the appendix.

2 Sparse Hopfield Model

In this section, we introduce the sparse Hopfield energy from convex conjugate of entropic regularizer, and then the sparse retrieval dynamics. In this paper we only consider the Gini entropic regularizer corresponding to the sparsemax distribution [Martins and Astudillo, 2016].

Let $\mathbf{x} \in \mathbb{R}^d$ represent the query pattern, and let $\boldsymbol{\Xi} \coloneqq (\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_M) \in \mathbb{R}^{d\times M}$ denote the memory patterns. The objective of the Hopfield models is to store the memory patterns $\boldsymbol{\Xi}$ and then retrieve a specific memory pattern $\boldsymbol{\xi}_\mu$ based on a given query $\mathbf{x}$. Consequently, any Hopfield model consist of two main components: an *energy function* $\mathcal{H}(\mathbf{x})$, encoding memories into its local minima, and a *retrieval dynamics* $\mathcal{T}(\mathbf{x})$, which retrieves a memory by iteratively minimizing $\mathcal{H}(\mathbf{x})$ when initialized with a query. We provide a visualization of this procedure in Figure 3. The construction of the energy function $\mathcal{H}(\mathbf{x})$ is straightforward. As emphasized in [Krotov and Hopfield, 2016], the

[3] Note that, the proposed model's sparsity falls under the category of *sparsity-inducing normalization maps*. Consequently, the forward pass still requires $\mathcal{O}(n^2)$ space complexity. Here, "*potential* computational efficiency" refers that the computational efficiency can be enhanced if one employs efficient implementations that leverage sparsity, such as sort operations or median-finding algorithms, to circumvent unnecessary computations, see Appendix C and [Martins and Astudillo, 2016, Section 2] for more discussions.

memories can be easily encoded into $\mathcal{H}(\mathbf{x})$ through the *overlap-construction*: $\mathcal{H}(\mathbf{x}) = F(\mathbf{\Xi}^\mathsf{T}\mathbf{x})$, where $F : \mathbb{R}^M \to \mathbb{R}$ is a smooth function. This ensures that the memories $\{\boldsymbol{\xi}_\mu\}_{\mu\in[M]}$ are located at the stationary points of $\mathcal{H}(\mathbf{x})$, since $\nabla_\mathbf{x} F(\mathbf{\Xi}^\mathsf{T}\mathbf{x})|_{\boldsymbol{\xi}_\mu} = 0$ for all $\mu \in [M]$. Different choices of F lead to different Hopfield models, as demonstrated in [Krotov and Hopfield, 2016, Demircigil et al., 2017, Ramsauer et al., 2021, Krotov and Hopfield, 2021]. However, finding a corresponding retrieval dynamics, $\mathcal{T}$, for a given energy $\mathcal{H}(\mathbf{x})$, is generally more challenging. This is because $\mathcal{T}$ needs to satisfy two conditions to ensure successful memory retrieval: (i) To ensure consistent retrieval, an appropriate $\mathcal{T}$ should monotonically minimize $\mathcal{H}(\mathbf{x})$ when iteratively applied. (ii) To ensure accurate retrieval, an appropriate $\mathcal{T}$ should align its fixed points (the points where iterative application terminates) with the stationary points of $\mathcal{H}(\mathbf{x})$.

To this end, we introduce the sparse Hopfield model, providing a principled construction for $\mathcal{H}$ and $\mathcal{T}$. This model not only fulfills the aforementioned desirable properties, but also enables more robust and faster memory retrieval compared to the modern Hopfield model [Ramsauer et al., 2021].

2.1 Sparse Hopfield Energy

Let $\mathbf{x} \in \mathbb{R}^d$ be the query pattern, and $\mathbf{\Xi} := (\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_M) \in \mathbb{R}^{d\times M}$ be the memory patterns. We introduce the sparse Hopfield energy as

$$\mathcal{H}(\mathbf{x}) = -\Psi^\star\left(\beta\mathbf{\Xi}^\mathsf{T}\mathbf{x}\right) + \frac{1}{2}\left\langle \mathbf{x}, \mathbf{x}\right\rangle, \tag{2.1}$$

with $\Psi^\star(\mathbf{z}) := \frac{1}{2}\|\mathbf{z}\|^2 - \frac{1}{2}\|\mathrm{Sparsemax}(\mathbf{z}) - \mathbf{z}\|^2 + \frac{1}{2}$, where $\mathrm{Sparsemax}(\cdot)$ is defined as follows. Let $\mathbf{z}, \mathbf{p} \in \mathbb{R}^M$, and $\Delta^M := \{\mathbf{p} \in \mathbb{R}^M_+ \mid \sum_\mu^M p_\mu = 1\}$ be the $(M-1)$-dimensional unit simplex.

Definition 2.1 (Sparsemax in Variational Form [Martins and Astudillo, 2016], also see Remark F.1)**.**

$$\mathrm{Sparsemax}(\mathbf{z}) := \mathop{\mathrm{ArgMin}}_{\mathbf{p}\in\Delta^M} \|\mathbf{p} - \mathbf{z}\|^2 = \mathop{\mathrm{ArgMax}}_{\mathbf{p}\in\Delta^M} \left[\mathbf{p}^\mathsf{T}\mathbf{z} - \Psi(\mathbf{p})\right], \tag{2.2}$$

where $\Psi(\mathbf{p}) := -\frac{1}{2}\sum_\nu^M p_\nu(1-p_\nu)$ is the negative Gini entropy or Gini entropic regularizer.

Remark 2.1. Recall that, the variational form (2.2) is in fact general, that applies to various entropic regularizers, as discussed in [Peters et al., 2019, Wainwright et al., 2008]. The choice of Ψ determines the resulting sparse probability distribution. For instance, if we choose the Gibbs' entropic regularizer $\Psi_{\text{Gibbs}} = -\sum_\nu^M p_\nu \ln p_\nu$, (2.2) reduces to the standard softmax distribution.

Overview of Theoretical Results. At first glance, the energy function (2.1) may seem peculiar. However, it indeed represents a non-trivial sparse Hopfield model with appealing properties, including:

(i) In response to challenge (C1) & (C2), as we shall see in Section 2.2, the energy (2.1) leads to a sparse retrieval dynamics that not only retrieves memory by monotonically decreasing (Lemma 2.1) to its stationary points (Lemma 2.2), but also associates with sparsemax attention through its single-step approximation (Remark 2.2);

(ii) In response to challenge (C3), as we shall see in Section 3, it indulges fast convergence of retrieval (Corollary 3.1.2), exponential-in-d memory capacity akin to modern Hopfield models (Lemma 3.1). Notably, it accomplishes these with a tighter retrieval error bound (Theorem 2.1).

We reveal each of these properties in the following sections.

2.2 Sparse Retrieval Dynamics and Connection to Sparse Attention

The optimization problem $\mathrm{ArgMax}_{\mathbf{p}\in\Delta^M}\left[\mathbf{p}^\mathsf{T}\mathbf{z} - \Psi(\mathbf{p})\right]$ does not necessarily have a closed-form solution for arbitrary Ψ. However, a family of Ψ has been investigated in literature [Correia et al., 2019, Martins and Astudillo, 2016] with closed-form solutions derived, including the $\mathrm{Sparsemax}(\cdot)$.

Sparsemax in Closed-Form (Proposition 1 of [Martins and Astudillo, 2016])**.** Let $\mathbf{z} \in \mathbb{R}^M$. Denote $[a]_+ := \mathrm{Max}\{0, a\}$, $z_{(\nu)}$ the ν'th element in a sorted descending z-sequence $\mathbf{z}_{\text{sorted}} := z_{(1)} \geq z_{(2)} \geq \ldots \geq z_{(M)}$, and $\kappa(\mathbf{z}) := \mathrm{Max}\left\{k \in [M] \;\middle|\; 1 + kz_{(k)} > \sum_{\nu\leq k} z_{(\nu)}\right\}$. The optimization problem(s) (2.2) has closed-form solution

$$\mathrm{Sparsemax}(\mathbf{z}) = \left[\mathbf{z} - \tau(\mathbf{z})\mathbf{1}_M\right]_+, \tag{2.3}$$

where $\tau : \mathbb{R}^M \to \mathbb{R}$ is the threshold function $\tau(\mathbf{z}) = \left[\left(\sum_{\nu\leq\kappa(\mathbf{z})} z_{(\nu)}\right) - 1\right]/\kappa(\mathbf{z})$, satisfying $\sum_{\mu=1}^M \left[z_\mu - \tau(\mathbf{z})\right]_+ = 1$ for all $\mathbf{z}$. Notably, $\kappa(\mathbf{z}) = |S(\mathbf{z})|$ where $S(\mathbf{z}) = \{\mu \in [M] \mid \mathrm{Sparsemax}_\mu(\mathbf{z}) > 0\}$ is the support set of $\mathrm{Sparsemax}(\mathbf{z})$.

In this case, we present the following theorem to derive the convex conjugate of Ψ in closed-form:

Theorem 2.1 (Convex Conjugate of Negative Gini Entropy)**.** Let $F(\mathbf{p}) := \langle \mathbf{p}, \mathbf{z}\rangle - \Psi(\mathbf{p})$ with Ψ being the negative Gini entropy, $\Psi(\mathbf{p}) = \frac{1}{2}\|\mathbf{p}\|^2 - \frac{1}{2}$. The convex conjugate of $\Psi(\mathbf{p})$ is

$$\Psi^\star(\mathbf{z}) := \underset{\mathbf{p}\in\Delta^M}{\mathrm{Max}}\, F(\mathbf{p}, \mathbf{z}) = \frac{1}{2}\|\mathbf{z}\|^2 - \frac{1}{2}\|\mathbf{p}^\star - \mathbf{z}\|^2 + \frac{1}{2}, \tag{2.4}$$

where $\mathbf{p}^\star = \mathrm{Sparsemax}(\mathbf{z})$ is given by (2.3).

Corollary 2.1.1. By Danskin's Theorem, $\boldsymbol{\nabla}\Psi^\star(\mathbf{z}) = \mathrm{ArgMax}_{\mathbf{p}\in\Delta^M} F(\mathbf{p}, \mathbf{z}) = \mathrm{Sparsemax}(\mathbf{z})$.

Proof. A detailed proof is shown in Appendix E.1. □

Theorem 2.1 and Corollary 2.1.1 not only provide the intuition behind the sparse Hopfield energy (2.1) — the memory patterns are stored in local minima aligned with the overlap-function constructions (i.e. $\left\|\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right\|^2 = \sum_{\mu=1}^{M} \langle \boldsymbol{\xi}_\mu, \mathbf{x}\rangle^2$) in [Ramsauer et al., 2021, Demircigil et al., 2017, Krotov and Hopfield, 2016] — but also prepare us for the following corresponding sparse retrieval dynamics.

Lemma 2.1 (Sparse Retrieval Dynamics)**.** Let t be the iteration number. The energy (2.1) can be monotonically decreased by the following sparse retrieval dynamics over t:

$$\mathcal{T}\left(\mathbf{x}_t\right) := \boldsymbol{\nabla}_{\mathbf{x}}\Psi\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)\Big|_{\mathbf{x}_t} = \boldsymbol{\Xi}\,\mathrm{Sparsemax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}_t\right) = \mathbf{x}_{t+1}. \tag{2.5}$$

Proof Sketch. To show monotonic decreasing property, we first derive the sparse retrieval dynamics by utilizing Theorem 2.1, Corollary 2.1.1, along with the convex-concave procedure [Yuille and Rangarajan, 2003, 2001]. Then, we show the monotonicity of $\mathcal{H}$ by constructing a iterative upper bound of $\mathcal{H}$ which is convex in $\mathbf{x}_{t+1}$ and thus, can be lowered iteratively by the convex-concave procedure. A detailed proof is shown in the Appendix E.2. □

Remark 2.2. Similar to [Ramsauer et al., 2021], (2.5) is equivalent to sparsemax attention [Martins and Astudillo, 2016] when the $\mathcal{T}$ is applied only once, see Appendix D for more details. Importantly, β acts as a scaling factor for the energy function, often referred to as the "inverse temperature". It influences the sharpness of energy landscape Equation (2.1), thereby controlling the dynamics. High β values, corresponding to low temperatures, encourage that the basins of attraction for individual memory patterns remain distinct, leading to easier retrieval.

Notably, since $\left\|\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right\|^2 = \sum_{\mu=1}^{M} \langle \boldsymbol{\xi}_\mu, \mathbf{x}\rangle^2$, (2.5) implies that the local optimum of $\mathcal{H}$ are located near the patterns $\boldsymbol{\xi}_\mu$. Different from previous studies on binary Hopfield models [Demircigil et al., 2017, Krotov and Hopfield, 2016], for continuous patterns, we adopt the relaxed definition from [Ramsauer et al., 2021][4] to rigorously analyze the memory retrieval, and the subsequent lemma.

Definition 2.2 (Stored and Retrieved)**.** Assuming that every pattern $\boldsymbol{\xi}_\mu$ surrounded by a sphere S_μ with finite radius $R := \frac{1}{2}\,\mathrm{Min}_{\mu,\nu\in[M]}\left\|\boldsymbol{\xi}_\mu - \boldsymbol{\xi}_\nu\right\|$, we say $\boldsymbol{\xi}_\mu$ is *stored* if there exists a generalized fixed point of $\mathcal{T}$, $\mathbf{x}^\star_\mu \in S_\mu$, to which all limit points $\mathbf{x} \in S_\mu$ converge to, and $S_\mu \cap S_\nu = \emptyset$ for $\mu \neq \nu$. We say $\boldsymbol{\xi}_\mu$ is ϵ-*retrieved* by $\mathcal{T}$ with $\mathbf{x}$ for an error[a] ϵ, if $\left\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\right\| \leq \epsilon$.

[a] The retrieval error has a naive bound $\epsilon := \mathrm{Max}\left\{\left\|\mathbf{x} - \boldsymbol{\xi}_\mu\right\|, \left\|\boldsymbol{\xi}_\mu - \mathbf{x}^\star_\mu\right\|\right\}$ by interpolating from $\mathbf{x}$ to $\boldsymbol{\xi}_\mu$.

Definition 2.2 sets the threshold for a memory pattern $\boldsymbol{\xi}_\mu$ to be considered *stored* at a fixed point of $\mathcal{T}$, $\mathbf{x}^\star_\mu$. However, this definition does not imply that the fixed points of $\mathcal{T}$ are also stationary points of the energy function $\mathcal{H}$. In fact, monotonicity of (2.5) does not assure the existence of stationary points of energy $\mathcal{H}$ [Sriperumbudur and Lanckriet, 2009]. To establish a well-defined Hopfield model, we need two types of convergence guarantees. The first is the convergence between $\mathbf{x}^\star_\mu$ and $\boldsymbol{\xi}_\mu$, which ensures that the retrieved memory is close to the stored memory. The second is the convergence of $\mathcal{H}$ to its stationary points through the dynamics of $\mathcal{T}$, which ensures that the system reaches a state of minimal energy. The following lemma provides the convergence results for both.

Lemma 2.2 (Convergence of Retrieval Dynamics $\mathcal{T}$)**.** Suppose $\mathcal{H}$ is given by (2.1) and $\mathcal{T}(\mathbf{x})$ is given by (2.5). For any sequence $\{\mathbf{x}_t\}_{t=0}^{\infty}$ defined by $\mathbf{x}_{t'+1} = \mathcal{T}(\mathbf{x}_{t'})$, all limit points of this sequence are stationary points if they are obtained by iteratively applying $\mathcal{T}$ to $\mathcal{H}$.

[4] Recall that a fixed point of $\mathcal{T}$ with respect to $\mathcal{H}$ is a point where $\mathbf{x} = \mathcal{T}(\mathbf{x})$, and a generalized fixed point is a point where $\mathbf{x} \in \mathcal{T}(\mathbf{x})$. For more details, refer to [Sriperumbudur and Lanckriet, 2009].

Proof Sketch. We verify and utilize Zangwill's global convergence theory [Zangwill, 1969] for iterative algorithms $\mathcal{T}$, to first show that all the limit points of $\{\mathbf{x}_t\}_{t=0}^{\infty}$ are generalized fixed points and $\lim_{t\to\infty} \mathcal{H}(\mathbf{x}_t) = \mathcal{H}(\mathbf{x}^\star)$, where $\mathbf{x}^\star$ are some generalized fixed points of $\mathcal{T}$. Subsequently, by [Sriperumbudur and Lanckriet, 2009, Lemma 5], we show that $\{\mathbf{x}^\star\}$ are also stationary points of $\mathrm{Min}_{\mathbf{x}}[\mathcal{H}]$, and hence $\mathcal{H}$ converges to local optimum. A detailed proof is shown in Appendix E.4. □

Intuitively, Lemma 2.2 indicates that the energy function converges to local optimum, i.e. $\lim_{t\to\infty} \mathcal{H}(\mathbf{x}_t) \to \mathcal{H}(\mathbf{x}^\star)$, where $\mathbf{x}^\star$ are stationary points of $\mathcal{H}$. Consequently, it offers formal justifications for the retrieval dynamics (2.5) to retrieve stored memory patterns $\{\boldsymbol{\xi}_\mu\}_{\mu\in[M]}$: for any query (initial point) $\mathbf{x}$, $\mathcal{T}$ monotonically and iteratively approaches stationary points of $\mathcal{H}$, where the memory patterns $\{\boldsymbol{\xi}_\mu\}_{\mu\in[M]}$ are stored. As for the retrieval error, we provide the following theorem stating that $\mathcal{T}$ achieves a lower retrieval error compared to its dense counterpart.

Theorem 2.2 (Retrieval Error)**.** Let $\mathcal{T}_{\text{Dense}}$ be the retrieval dynamics of the dense modern Hopfield model [Ramsauer et al., 2021]. It holds $\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ for all $\mathbf{x} \in S_\mu$. Moreover,

$$\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq m + d^{1/2} m\beta \left[\kappa \left(\operatorname*{Max}_{\nu\in[M]} \langle \boldsymbol{\xi}_\nu, \mathbf{x}\rangle - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right], \tag{2.6}$$

where $\left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}$ is the κth-largest element of $\boldsymbol{\Xi}^\mathsf{T}\mathbf{x} \in \mathbb{R}^M$ following the sparsemax definition (2.3).

Proof. A detailed proof is shown in Appendix E.3. □

Interestingly, (2.6) is a sparsity dependent bound[5]. By denoting $n := \|\mathbf{x}\|$, the second term on the RHS of (2.6) is dominated by the sparsity dimension κ as it can be expressed as $\kappa\left(1 - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}/(nm)\right) \propto \alpha\kappa$ with a constant $0 \leq \alpha \leq 2$. When $\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}$ is sparse (i.e. κ is small), the bound is tighter, vice versa.

Remark 2.3 (Faster Convergence)**.** Computationally, Theorem 2.2 implies that $\mathcal{T}$ requires fewer iterations to reach fixed points with the same amount of error tolerance compared to $\mathcal{T}_{\text{dense}}$. Namely, $\mathcal{T}$ retrieves stored memory patterns faster and therefore more efficiently, as evidenced in Figure 2.

Remark 2.4 (Noise-Robustness)**.** Moreover, in cases of contaminated patterns with noise $\boldsymbol{\eta}$, i.e. $\widetilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\eta}$ (noise in query) or $\widetilde{\boldsymbol{\xi}}_\mu = \boldsymbol{\xi}_\mu + \boldsymbol{\eta}$ (noise in memory), the impact of noise $\boldsymbol{\eta}$ on the sparse retrieval error (2.6) is linear, while its effect on the dense retrieval error (2.7) is exponential. This suggests the robustness advantage of the sparse Hopfield model, as evidenced in Figure 1.

2.3 Sparse Hopfield Layers for Deep Learning

The sparse Hopfield model can serve as a versatile component for deep learning frameworks, given its continuity and differentiability with respect to parameters. Corresponding to three types of Hopfield Layers proposed in [Ramsauer et al., 2021], we introduce their sparse analogs: **(1)** `SparseHopfield`, **(2)** `SparseHopfieldPooling`, **(3)** `SparseHopfieldLayer`. Layer `SparseHopfield` has memory (stored or key) patterns $\boldsymbol{\Xi}$ and query (state) pattern $\mathbf{x}$ as inputs, and associates these two sets of patterns via the sparse retrieval dynamics (2.5). This layer regards the transformer attention layer as its one-step approximation, while utilizing the sparsemax [Martins and Astudillo, 2016] on attention matrix. Layer `SparseHopfieldPooling` and Layer `SparseHopfieldLayer` are two variants of `SparseHopfield`, whose input patterns are memory patterns and query patterns from previous layers or external plugin, respectively. `SparseHopfieldPooling`, whose query patterns are learnable parameters, can be interpreted as performing a pooling operation over input memory patterns. `SparseHopfieldLayer`, by contrast, has learnable memory patterns that maps query patterns to hidden states with sparsemax activation. Thus it can substitute a fully connected layer within deep learning architectures. See (D.12) and the implementation Algorithm 1 in Appendix D, and [Ramsauer et al., 2021, Section 3] for more details of these associations. In Section 4, we apply these layers and compare them with their dense counterparts in [Ramsauer et al., 2021] and other baseline machine learning methods.

3 Fundamental Limits of Memory Capacity of Sparse Hopfield Models

How many patterns can be stored and reliably retrievable in the proposed model? We address this by decomposing it into to two sub-questions and answering them separately:

(A) What is the condition for a pattern $\boldsymbol{\xi}_\mu$ considered well stored in $\mathcal{H}$, and correctly retrieved?

[5] Notably, $\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ is also upper-bounded by a sparsity-independent but M, β-dependent bound

$$\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq 2m(M-1)\exp\left\{-\beta\left(\langle\boldsymbol{\xi}_\mu, \mathbf{x}\rangle - \operatorname*{Max}_{\nu\in[M]} \langle\boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu\rangle\right)\right\}. \tag{2.7}$$

(B) What is the number, in expectation, of the the patterns satisfying such condition?

For (A), we first introduce the notion of separation of patterns following [Ramsauer et al., 2021],

Definition 3.1 (Separation of Patterns)**.** The separation of a memory pattern $\boldsymbol{\xi}_\mu$ from all other memory patterns $\boldsymbol{\Xi}$ is defined as its minimal inner product difference to any other patterns:

$$\Delta_\mu \coloneqq \underset{\nu,\nu\neq\mu}{\mathrm{Min}} \left[\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu \rangle - \langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu \rangle\right] = \langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu \rangle - \underset{\nu,\nu\neq\mu}{\mathrm{Max}} \left[\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu \rangle\right]. \tag{3.1}$$

Similarly, the separation of $\boldsymbol{\xi}_\mu$ at a given $\mathbf{x}$ from all memory patterns $\boldsymbol{\Xi}$ is given by

$$\widetilde{\Delta}_\mu \coloneqq \underset{\nu,\nu\neq\mu}{\mathrm{Min}} \left[\langle \mathbf{x}, \boldsymbol{\xi}_\mu \rangle - \langle \mathbf{x}, \boldsymbol{\xi}_\nu \rangle\right]. \tag{3.2}$$

and then the well-separation condition for a pattern being well-stored and retrieved.

Theorem 3.1 (Well-Separation Condition)**.** Given the definition of stored and retrieved memories in Definition 2.2, suppose the memory patterns $\{\boldsymbol{\xi}_\mu\}_{\mu\in[M]}$ are located within the sphere $S_\mu \coloneqq \left\{\mathbf{x} \mid \|\mathbf{x} - \boldsymbol{\xi}_\mu\| \le R\right\}$, where the radius R is finite and defined as $R \coloneqq \frac{1}{2}\,\mathrm{Min}_{\mu,\nu\in[M]} \|\boldsymbol{\xi}_\mu - \boldsymbol{\xi}_\nu\|$ for all μ. Then, the retrieval dynamics $\mathcal{T}$ maps the sphere S_μ onto itself under the following conditions:

1. The initial query $\mathbf{x}$ is located within the sphere S_μ, i.e., $\mathbf{x} \in S_\mu$.
2. The *well-separation* condition is satisfied, which is given by:

$$\Delta_\mu \ge mn + 2mR - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)} - \frac{1}{\kappa}\left(\frac{R - m - md^{1/2}}{m\beta d^{1/2}}\right).$$

Corollary 3.1.1. Let $\delta \coloneqq \|\mathcal{T}_{\text{Dense}} - \boldsymbol{\xi}_\mu\| - \|\mathcal{T} - \boldsymbol{\xi}_\mu\|$. The well-separation condition can be expressed as $\Delta_\mu \ge \frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR$, which reduces to that of the dense Hopfield model when $\delta = 0$.

Proof Sketch. The proofs proceed by connecting Δ_μ with $\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$. To do so, we utilize Theorem 2.2 to incorporate the Δ_μ-dependent bound on the retrieval error of both sparse and dense Hopfield models [Ramsauer et al., 2021]. A detailed proof is shown in Appendix E.5. □

Together with Lemma 2.2, the well-separated condition serves as the necessary condition for pattern $\boldsymbol{\xi}_\mu$ to be well-stored at the stationary points of $\mathcal{H}$, and can be retrieved with at most $\epsilon = R$ by $\mathcal{T}$, as per Definition 2.2. We make the following three observations about the blessings from sparsity.

1. In general, to appreciate the blessings of sparsity, we rearrange the well-separation condition as

$$\Delta_\mu \ge 2mR + \underbrace{\left(mn - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right)}_{\coloneqq \alpha nm \quad \text{with } 0\le\alpha\le 2} - \frac{1}{\kappa}\left(\frac{R - m - md^{1/2}}{m\beta d^{1/2}}\right), \tag{3.3}$$

   and observe the two competing terms, αnm and $(R-m-md^{1/2})/(\kappa m\beta d^{1/2})$. Sparsity proves advantageous when the latter term surpasses the former, i.e. the sparse well-separation condition is consistently lower than its dense counterpart. The condition under which sparsity benefits are more likely to emerge (i.e., when the well-separation condition is more readily satisfied) is thereby:

$$\frac{1}{2}\underset{\mu,\nu\in[M]}{\mathrm{Min}} \|\boldsymbol{\xi}_\mu - \boldsymbol{\xi}_\nu\| \ge md^{1/2}\left(1 + \alpha\beta nm\kappa\right) + m, \quad \text{with } 0 \le \alpha \le 2. \tag{3.4}$$

   Intuitively, the sparser $\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}$ is, the easier it is for the above condition to be fulfilled.

2. **Large M limit:** For large M, the dense well-separation condition (Corollary 3.1.1) explodes while the sparse one (Theorem 3.1) saturates to the first three M-independent terms. This suggests that the hardness of distinguishing patterns can be tamed by the sparsity, preventing an increase of Δ_μ with M as observed in the dense Hopfield model. We numerically confirm this in Figure 1.

3. $\beta \to \infty$ **Limit:** In the region of low temperature, where $\beta \to \infty$ and hence all patterns can be *error-free* retrieved as per (2.7), we have $\Delta_\mu \ge 2mR + \alpha nm$ with $0 \le \alpha \le 2$. Here, the second term on the RHS represents the sparsity level of $\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}$, i.e. a smaller α indicates a higher degree of sparsity in $\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}$. Hence, the higher the sparsity, the easier it is to separate patterns.

For (B), equipped with Theorem 3.1 and Corollary 3.1.1, we provide a lower bound for the number of patterns being well-stored and can be *at least* R-retrieved in the next lemma[6]:

[6] Following the convention in memory capacity literature [Ramsauer et al., 2021, Demircigil et al., 2017, Krotov and Hopfield, 2016], we assume that all memory patterns $\{\boldsymbol{\xi}_\mu\}$ are sampled from a d-sphere of radius m.

Lemma 3.1 (Memory Capacity Lower Bound)**.** Let $1-p$ be the probability of successfully storing and retrieving a pattern. The number of patterns randomly sampled from a sphere of radius m that the sparse Hopfield model can store and retrieve is lower-bounded by

$$M \geq \sqrt{p}C^{\frac{d-1}{4}}, \tag{3.5}$$

where C is the solution to $C = {}^{b}/_{W_0(\exp\{a+\ln b\})}$ with $W_0(\cdot)$ being the principal branch of Lambert W function, $a := {}^{4}/_{d-1}\Big\{\ln\big[{}^{2m(\sqrt{p}-1)}/_{(R+\delta)}\big]+1\Big\}$ and $b := {}^{4m^2\beta}/_{5(d-1)}$. For sufficiently large β, the sparse Hopfield model exhibits a larger lower bound on the exponential memory capacity compared to its dense counterpart [Ramsauer et al., 2021]: $M \geq M_{\text{Dense}}$.

Proof Sketch. Our proof is built on [Ramsauer et al., 2021]. The high-level idea is to utilize the separation of random patterns sampled from spheres [Cai and Jiang, 2012, Brauchart et al., 2018] and the asymptotic expansion of the Lambert W function [Corless et al., 1996]. Firstly, we link the well-separation condition to cosine similarity distance, creating an inequality for the probability of a pattern being well-stored and retrieved. Next, we identify and prove conditions for the exponential memory capacity $M = \sqrt{p}C^{(d-1)/4}$ to hold. Finally, we analyze the scaling behaviors of C using its asymptotic expansion and show that $M \geq M_{\text{Dense}}$. A detailed proof is shown in Appendix E.6. □

Intuitively, the benefits of sparsity arises from the increased energy landscape separation provided by the sparse Hopfield energy function, which enables the separation of closely correlated patterns, resulting in a tighter well-separation condition for distinguishing such patterns and hence a larger lower bound on the memory capacity. Moreover, the sparse Hopfield model also enjoys the properties of fast convergence and exponentially suppressed retrieval error provided by the following corollary.

Corollary 3.1.2 (Fast Convergence and Exponentially Suppressed Retrieval Error)**.** For any query $\mathbf{x}$, $\mathcal{T}$ approximately retrieves a memory pattern $\boldsymbol{\xi}_\mu$ with retrieval error ϵ exponentially suppressed by Δ_μ: $\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq 2m(M-1)\exp\big\{-\beta\left(\Delta_\mu - 2m\,\mathrm{Max}\left[\|\mathbf{x}-\boldsymbol{\xi}_\mu\|, \|\mathbf{x}-\mathbf{x}_\mu^\star\|\right]\right)\big\}$.

Proof. This results from Theorem 2.2, Lemma 2.2, and [Ramsauer et al., 2021, Theorem 4]. □

Corollary 3.1.2 suggests that, with a sufficient Δ_μ, $\mathcal{T}$ can approximately retrieve patterns after a single *activation*, allowing the integration of sparse Hopfield models into deep learning architectures similarly to [Hoover et al., 2023, Seidl et al., 2022, Fürst et al., 2022, Ramsauer et al., 2021].

4 Proof of Concept Experimental Studies

We demonstrate the validity of our theoretical results and method by testing them on various experimental settings with both synthetic and real-world datasets.

4.1 Experimental Validation of Theoretical Results

We conduct experiments to verify our theoretical findings, and report the results in Figure 1. For the memory capacity (the top row of Figure 1), we test the proposed sparse model on retrieving half-masked patterns comparing with the Dense (Softmax) and 10th order polynomial Hopfield models [Millidge et al., 2022, Krotov and Hopfield, 2016] on MNIST (high sparsity), Cifar10 (low sparsity) and ImageNet (low sparsity) datasets. For all Hopfield models, we set $\beta = 1$.[7] A query is regarded as correctly retrieved if its cosine similarity error is below a set threshold. In addition, for the robustness against noisy queries (the bottom row of Figure 1), we inject Gaussian noises with varying variances (σ) into the images. Plotted are the means and standard deviations of 10 runs. The results show that the proposed sparse Hopfield model excels when memory patterns exhibit a high degree of sparsity and the signal-to-noise ratio in patterns is low, aligning with our theoretical results.

4.2 Multiple Instance Learning Tasks

Ramsauer et al. [2021] point out that the memory-enhanced Hopfield layers present a promising approach for Multiple Instance Learning (MIL) tasks. Multiple Instance Learning (MIL) [Ilse et al., 2018, Carbonneau et al., 2018] is a variation of supervised learning where the training set consists of labeled bags, each containing multiple instances. The goal of MIL is to predict the bag labels based on the instances they contain, which makes it particularly useful in scenarios where labeling individual instances is difficult or impractical, but bag-level labels are available. Examples of such scenarios include medical imaging (where a bag could be an image, instances could be patches of the

[7] However, as pointed out in [Millidge et al., 2022], this is in fact *not* fair to compare modern Hopfield with $\beta = 1$ with higher order polynomial Hopfield models.

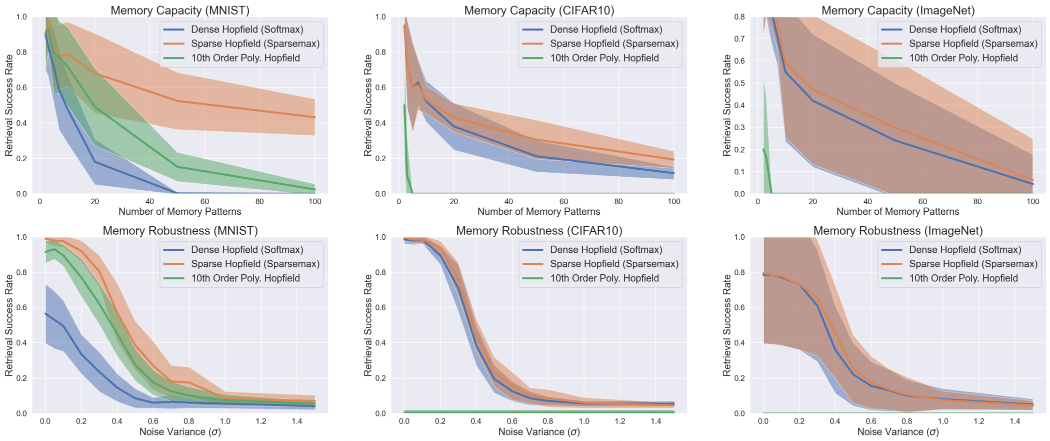

Figure 1: **Top:** Memory Capacity measured by successful half-masked retrieval rates. **Bottom:** Memory Robustness measured by retrieving patterns with varying levels of Gaussian noise. For all Hopfield models, we set $\beta = .01/0.1/0.1$ (for MNIST/CIFAR10/ImageNet) for better visualization. A query pattern is deemed correctly retrieved if its cosine similarity error is below a set threshold. For MNIST/CIFAR10/ImageNet datasets, we set the error thresholds to be 10/20/20% to cope with different sparse levels in data. Plotted are the means and standard deviations of 10 runs. The results suggest that the sparse Hopfield model excels when memory patterns exhibit a high degree of sparsity and the signal-to-noise ratio in patterns is low.

image, and the label could indicate the presence or absence of disease) and document classification (where a bag could be a document, instances could be the words or sentences in the document, and the label could indicate the topic or sentiment of the document). In this subsection, we implement our sparse Hopfield layers and applied them to MIL tasks on one synthetic and four real-world settings.

4.2.1 Synthetic Experiments

We use a synthetic MIL dataset, the bit pattern dataset, to demonstrate the effectiveness of the sparse Hopfield model. Each bag in this synthetic dataset contains a set of binary bit strings. The positive bag includes at least one of the positive bit patterns. We compare the performance of the `SparseHopfield` and `SparseHopfieldPooling` to their dense counterparts and vanilla attention [Vaswani et al., 2017]. We report the mean test accuracy of 10 runs. To demonstrate the effectiveness of sparse Hopfield model, we vary two hyperparameters of the bit pattern dataset corresponding to two perspectives: bag sparsity (sparsity in data) and bag size (number of memory patterns, M). For **bag sparsity**, we fix the bag size as 200, and inject from 2 to 80 positive patterns in a positive bag, results in 1 to 40 percent of positive patterns in each positive bag. For **bag size**, we fix the number of positive pattern in a bag to be 1, and vary bag size from 20 to 300. We report results of `SparseHopfieldPooling` in Table 1, and implementation details in Appendix H.1.1. A more complete version of Table 1, including the results of `Hopfield` and attention, is in Appendix G. The sparse Hopfield model demonstrates a better performance across all sparsity and all bag sizes.

Table 1: **Top (Bag Size):** Accuracy comparison on bit pattern dataset for sparse and dense Hopfield model. We report the average accuracy over 10 runs. The results suggest that the sparse Hopfield model demonstrates a better performance when facing a bag size increase. **Bottom (Bag Sparsity):** Performance comparison on bit pattern dataset for sparse and dense Hopfield model with varying bag sparsity. We report the average accuracy over 10 runs. The results suggest that the sparse Hopfield model demonstrates a better performance across all sparsity.

Bag Size	20	50	100	150	200	300
Dense Hopfield Pooling	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	76.44 ± 0.23	49.13 ± 0.01	52.88 ± 0.01
Sparse Hopfield Pooling	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	**99.76 ± 0.00**	**99.76 ± 0.00**	**99.76 ± 0.00**

Bag Sparsity	1%	5%	10%	20%	40%
Dense Hopfield Pooling	49.20 ± 0.00	85.58 ± 0.10	100.0 ± 0.00	100.0 ± 0.00	99.68 ± 0.00
Sparse Hopfield Pooling	**73.40 ± 0.06**	**99.68 ± 0.00**	100.0 ± 0.00	100.0 ± 0.00	**100.0 ± 0.00**

Convergence Analysis. In Figure 2, we numerically examine the convergence of the sparse and dense Hopfield models, plotting their loss and accuracy for the **bag size** tasks in above on the bit pattern dataset. We include multiple bag sizes to assess the effect of increasing memory patterns (i.e. M) on the loss curve. The plotted are the loss and accuracy curves of `SparseHopfieldPooling`. We refer results of `Hopfield` and more details to Appendix G.3. The results (Figure 2) show that, sparse Hopfield model surpasses its dense counterpart in all bag sizes. Moreover, for the same bag size, the sparse Hopfield model always reaches the minimum validation loss faster than dense Hopfield model, validating our Theorem 2.2.

Sparsity Generalization. We also evaluate the models' generalization performance with shifting information sparsity, by training dense and sparse Hopfield models with a specific bag sparsity and testing them on the other. We report the results in Table 5 and refer more details to Appendix G.3.

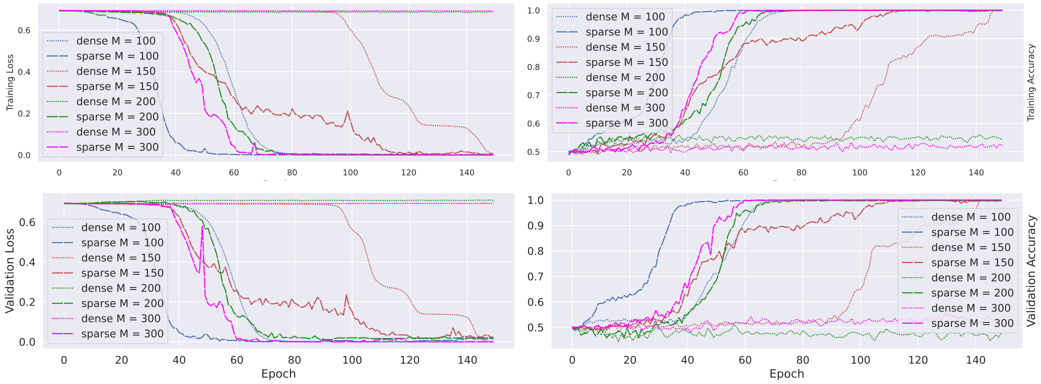

Figure 2: **Top:** The training loss and accuracy curve of dense and sparse Hopfield models with different bag sizes. **Bottom:** The validation loss and accuracy curve of dense and sparse Hopfield models with different bag sizes. The plotted are the mean of 10 runs. The results indicate that the sparse Hopfield model converges faster than the dense model and also yields superior accuracy.

4.2.2 Real-World MIL Tasks

Next, we demonstrate that the proposed method achieves near-optimal performance on four realistic (*non-sparse*) MIL benchmark datasets: Elephant, Fox and Tiger for image annotation [Ilse et al., 2018], UCSB breast cancer classification [Kandemir et al., 2014]. We use `Hopfield` and `SparseHopfield` to construct a similar model architecture proposed in [Ramsauer et al., 2021] and a detailed description of this experiment as well as its training and evaluating process can be found in Appendix H.1.2. As shown in Table 2, both Sparse and Dense Hopfield achieve near-best results on Tiger, Elephant and UCSB datasets, despite the low sparsity in data. The sparse Hopfield model outperforms the dense Hopfield model by a small margin on three out of four datasets.

5 Conclusion

We present a sparse Hopfield model with a memory-retrieval dynamics that corresponds to the sparse-structured attention mechanism. This model is capable of merging into deep learning architectures with data-dependent sparsity. Theoretically, we introduce a principled construction for modern Hopfield models, based on the convex conjugate of different entropy regularizers. It allows us to easily recover the dense modern Hopfield model [Ramsauer et al., 2021]] using Gibbs entropy. Moreover, we introduce the sparse Hopfield model using the Gini entropic regularizer, and explore its theoretical advantages, delineating conditions that favor its use. Empirically, we demonstrate our theoretical results and methodology to be effective on various synthetic and realistic settings. This work extends the correspondence between artificial and biological neural networks to sparse domain, potentially paving the way for future Hopfield-based methodologies and bio-inspired computing systems.

Table 2: Results for MIL benchmark datasets in terms of AUC score. The baselines are Path encoding [Küçükaşçı and Baydoğan, 2018], MInD [Cheplygina et al., 2015], MILES [Chen et al., 2006], APR [Dietterich et al., 1997], Citation-KNN [Wang and Zucker, 2000] and DD [Maron and Lozano-Pérez, 1997]. Results for baselines are taken from [Ramsauer et al., 2021]. The results suggest the proposed model achieves near-optimal performance even when the data is not sparse.

Method	Tiger	Fox	Elephant	UCSB
Dense Hopfield	0.878 ± 0.028	0.600 ± 0.011	0.907 ± 0.022	0.880 ± 0.013
Sparse Hopfield	0.892 ± 0.021	0.611 ± 0.010	0.912 ± 0.016	0.877 ± 0.009
Path encoding	0.910 ± 0.010	0.712 ± 0.014	0.944 ± 0.007	0.880 ± 0.022
MInD	0.853 ± 0.011	0.704 ± 0.016	0.936 ± 0.009	0.831 ± 0.027
MILES	0.872 ± 0.017	0.738 ± 0.016	0.927 ± 0.007	0.833 ± 0.026
APR	0.778 ± 0.007	0.541 ± 0.009	0.550 ± 0.010	—
Citation-kNN	0.855 ± 0.009	0.635 ± 0.015	0.896 ± 0.009	0.706 ± 0.032
DD	0.841	0.631	0.907	—

Acknowledgments

JH would like to thank to Tim Tsz-Kit Lau, Robin Luo and Andrew Chen for enlightening discussions on related topics, and Jiayi Wang for invaluable support in facilitating experimental deployments. The authors would like to thank the anonymous reviewers and program chairs for constructive comments. JH is partially supported by the Walter P. Murphy Fellowship. HL is partially supported by NIH R01LM1372201, NSF CAREER1841569 and a NSF TRIPODS1740735. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. Deep equilibrium models. *Advances in Neural Information Processing Systems*, 32, 2019. URL https://arxiv.org/abs/1909.01377.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. URL https://arxiv.org/abs/2004.05150.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, et al. Findings of the 2017 conference on machine translation (wmt17). Association for Computational Linguistics, 2017.

Johannes Brandstetter. Blog post: Hopfield networks is all you need, 2021. URL https://ml-jku.github.io/hopfield-layers/. Accessed: April 4, 2023.

Johann S Brauchart, Alexander B Reznikov, Edward B Saff, Ian H Sloan, Yu Guang Wang, and Robert S Womersley. Random point sets on the sphere—hole radii, covering, and separation. *Experimental Mathematics*, 27(1):62–81, 2018. URL https://arxiv.org/abs/1512.07470.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. URL https://arxiv.org/abs/1512.07470.

T Tony Cai and Tiefeng Jiang. Phase transition in limiting distributions of coherence of high-dimensional random matrices. *Journal of Multivariate Analysis*, 107:24–39, 2012. URL https://arxiv.org/abs/1102.2926.

Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*, 77:329–353, 2018. URL https://arxiv.org/abs/1612.03365.

Minghao Chen, Houwen Peng, Jianlong Fu, and Haibin Ling. Autoformer: Searching transformers for visual recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12270–12280, 2021. URL https://arxiv.org/abs/2107.00651.

Yixin Chen, Jinbo Bi, and James Ze Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):1931–1947, 2006.

Veronika Cheplygina, David MJ Tax, and Marco Loog. Dissimilarity-based ensembles for multiple instance learning. *IEEE transactions on neural networks and learning systems*, 27(6):1379–1391, 2015. URL https://arxiv.org/abs/1402.1349.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019. URL https://arxiv.org/abs/1904.10509.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. URL https://arxiv.org/abs/2204.02311.

Robert M Corless, Gaston H Gonnet, David EG Hare, David J Jeffrey, and Donald E Knuth. On the lambert w function. *Advances in Computational mathematics*, 5:329–359, 1996.

Gonçalo M Correia, Vlad Niculae, and André FT Martins. Adaptively sparse transformers. *arXiv preprint arXiv:1909.00015*, 2019. URL https://arxiv.org/abs/1909.00015.

Herbert A David and Haikady N Nagaraja. *Order statistics*. John Wiley & Sons, 2004.

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*, 168:288–299, 2017. URL https://arxiv.org/abs/1702.01929.

Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997.

Michael Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*, volume 2. Springer, 2010.

Peter Földiak. Forming sparse representations by local anti-hebbian learning. *Biological cybernetics*, 64(2):165–170, 1990.

Andreas Fürst, Elisabeth Rumetshofer, Johannes Lehner, Viet T Tran, Fei Tang, Hubert Ramsauer, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto, et al. Cloob: Modern hopfield networks with infoloob outperform clip. *Advances in neural information processing systems*, 35: 20450–20468, 2022. URL https://arxiv.org/abs/2110.11316.

Bolin Gao and Lacra Pavel. On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*, 2017. URL https://arxiv.org/abs/1704.00805.

Asela Gunawardana, William Byrne, and Michael I Jordan. Convergence theorems for generalized alternating minimization procedures. *Journal of machine learning research*, 6 (12), 2005. URL https://www.jmlr.org/papers/volume6/gunawardana05a/gunawardana05a.pdf.

Benjamin Hoover, Yuchen Liang, Bao Pham, Rameswar Panda, Hendrik Strobelt, Duen Horng Chau, Mohammed J Zaki, and Dmitry Krotov. Energy transformer. *arXiv preprint arXiv:2302.07253*, 2023. URL https://arxiv.org/abs/2302.07253.

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.

John J Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the national academy of sciences*, 81(10):3088–3092, 1984.

Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International conference on machine learning*, pages 2127–2136. PMLR, 2018. URL https://arxiv.org/abs/1802.04712.

Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37 (15):2112–2120, 2021. URL https://www.biorxiv.org/content/10.1101/2020.09.17.301879v1.

Melih Kandemir, Chong Zhang, and Fred A Hamprecht. Empowering multiple instance histopathology cancer diagnosis by cell graphs. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014: 17th International Conference, Boston, MA, USA, September 14-18, 2014, Proceedings, Part II 17*, pages 228–235. Springer, 2014.

Leo Kozachkov, Ksenia V Kastanenka, and Dmitry Krotov. Building transformers from neurons and astrocytes. *Proceedings of the National Academy of Sciences*, 120(34):e2219150120, 2023. URL https://www.biorxiv.org/content/10.1101/2022.10.12.511910v1.

Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. *Advances in neural information processing systems*, 29, 2016. URL https://arxiv.org/abs/1606.01164.

Dmitry Krotov and John J. Hopfield. Large associative memory problem in neurobiology and machine learning. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2008.06996.

Emel Şeyma Küçükaşcı and Mustafa Gökçe Baydoğan. Bag encoding strategies in multiple instance learning problems. *Information Sciences*, 467:559–578, 2018.

YC Lee, Gary Doolen, HH Chen, GZ Sun, Tom Maxwell, and HY Lee. Machine learning using a higher order correlation network. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States); Univ. of Maryland, College Park, MD (United States), 1986.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. URL https://arxiv.org/abs/2103.14030.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. URL https://arxiv.org/abs/1711.05101.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1), 2010. URL https://arxiv.org/abs/0908.0050.

Alireza Makhzani and Brendan J Frey. Winner-take-all autoencoders. *Advances in neural information processing systems*, 28, 2015. URL https://arxiv.org/abs/1409.2752.

Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. *Advances in neural information processing systems*, 10, 1997.

Andre Martins and Ramon Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International conference on machine learning*, pages 1614–1623. PMLR, 2016. URL https://arxiv.org/abs/1602.02068.

Beren Millidge, Tommaso Salvatori, Yuhang Song, Thomas Lukasiewicz, and Rafal Bogacz. Universal hopfield networks: A general framework for single-shot associative memory models. In *International Conference on Machine Learning*, pages 15561–15583. PMLR, 2022. URL https://arxiv.org/abs/2202.04557.

Charles M Newman. Memory capacity in neural network models: Rigorous lower bounds. *Neural Networks*, 1(3):223–238, 1988.

Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

Frank WJ Olver, Daniel W Lozier, Ronald F Boisvert, and Charles W Clark. *NIST handbook of mathematical functions hardback and CD-ROM*. Cambridge university press, 2010.

Fabian Paischer, Thomas Adler, Vihang Patil, Angela Bitto-Nemling, Markus Holzleitner, Sebastian Lehner, Hamid Eghbal-Zadeh, and Sepp Hochreiter. History compression via language models in reinforcement learning. In *International Conference on Machine Learning*, pages 17156–17185. PMLR, 2022. URL https://arxiv.org/abs/2205.12258.

Günther Palm. Neural associative memories and sparse coding. *Neural Networks*, 37:165–171, 2013.

Pierre Peretto and Jean-Jacques Niez. Long term memory storage capacity of multiconnected neural networks. *Biological Cybernetics*, 54(1):53–63, 1986.

Ben Peters, Vlad Niculae, and André FT Martins. Sparse sequence-to-sequence models. *arXiv preprint arXiv:1905.05702*, 2019. URL https://arxiv.org/abs/1905.05702.

Jiezhong Qiu, Hao Ma, Omer Levy, Scott Wen-tau Yih, Sinong Wang, and Jie Tang. Blockwise self-attention for long document understanding. *arXiv preprint arXiv:1911.02972*, 2019. URL https://arxiv.org/abs/1911.02972.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2008.02217.

Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, 2010.

Philipp Seidl, Philipp Renz, Natalia Dyubankova, Paulo Neves, Jonas Verhoeven, Jorg K Wegner, Marwin Segler, Sepp Hochreiter, and Gunter Klambauer. Improving few-and zero-shot reaction template prediction using modern hopfield networks. *Journal of chemical information and modeling*, 62(9):2111–2120, 2022.

Bharath K Sriperumbudur and Gert RG Lanckriet. On the convergence of the concave-convex procedure. In *Advances in neural information processing systems*, volume 9, pages 1759–1767, 2009. URL https://papers.nips.cc/paper_files/paper/2009/file/8b5040a8a5baf3e0e67386c2e3a9b903-Paper.pdf.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. Sparse sinkhorn attention. In *International Conference on Machine Learning*, pages 9438–9447. PMLR, 2020.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *ACM Computing Surveys*, 55(6):1–28, 2022. URL https://arxiv.org/abs/2009.06732.

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. URL https://arxiv.org/abs/1706.03762.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Jun Wang and Jean-Daniel Zucker. Solving multiple-instance problem: A lazy learning approach. 2000.

Michael Widrich, Bernhard Schäfl, Milena Pavlović, Hubert Ramsauer, Lukas Gruber, Markus Holzleitner, Johannes Brandstetter, Geir Kjetil Sandve, Victor Greiff, Sepp Hochreiter, et al. Modern hopfield networks and attention for immune repertoire classification. *Advances in Neural Information Processing Systems*, 33:18832–18845, 2020. URL https://arxiv.org/abs/2007.13505.

Yongyi Yang, Zengfeng Huang, and David Wipf. Transformers from an optimization perspective. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=VT0Y4PlV2m0.

Alan L Yuille and Anand Rangarajan. The concave-convex procedure (cccp). *Advances in neural information processing systems*, 14, 2001. URL https://proceedings.neurips.cc/paper_files/paper/2001/file/a012869311d64a44b5a0d567cd20de04-Paper.pdf.

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

Willard I Zangwill. *Nonlinear programming: a unified approach*, volume 52. Prentice-Hall Englewood Cliffs, NJ, 1969.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vSVLM2j9eie.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021. URL https://arxiv.org/abs/2012.07436.

Tian Zhou, Ziqing Ma, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, et al. Film: Frequency improved legendre memory model for long-term time series forecasting. *Advances in Neural Information Processing Systems*, 35:12677–12690, 2022. URL https://arxiv.org/abs/2205.08897.

Appendix

A Nomenclature Table

We summarize our notations in the following table for easy reference.

Table 3: Mathematical Notations and Symbols

Symbol	Description
$\langle \mathbf{a}, \mathbf{b} \rangle$	Inner product for vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$
$[I]$	Index set $\{1, \cdots, I\}$, where $I \in \mathbb{N}^+$
$\lVert \cdot \rVert_2$	Spectral norm, equivalent to the l_2-norm when applied to a vector
d	Dimension of patterns
M	Number of stored memory patterns
β	A scaling factor of the energy function that controls the learning dynamics
$\mathbf{x}$	State/configuration/query pattern in $\mathbb{R}^d$
$\boldsymbol{\xi}$	Memory patterns (keys) in $\mathbb{R}^d$
$\boldsymbol{\Xi}$	Shorthand for M stored memory (key) patterns $\{\boldsymbol{\xi}_\mu\}_{\mu \in [M]}$ in $\mathbb{R}^{d \times M}$
$\boldsymbol{\Xi}^\mathsf{T} \mathbf{x}$	M-dimensional overlap vector $(\langle \boldsymbol{\xi}_1, \mathbf{x} \rangle, \cdots, \langle \boldsymbol{\xi}_\mu, \mathbf{x} \rangle, \cdots, \langle \boldsymbol{\xi}_M, \mathbf{x} \rangle)$ in $\mathbb{R}^M$
$\left[\boldsymbol{\Xi}^\mathsf{T} \mathbf{x}\right]_\kappa$	The κ-th element of $\boldsymbol{\Xi}^\mathsf{T} \mathbf{x}$
n	Norm of $\mathbf{x}$, denoted as $n := \lVert \mathbf{x} \rVert$
m	Largest norm of memory patterns, denoted as $m := \mathrm{Max}_{\mu \in [M]} \lVert \boldsymbol{\xi}_\mu \rVert$
κ	The number of non-zero element of $\mathrm{Sparsemax}$, defined in (2.3)
R	The minimal Euclidean distance across all possible pairs of memory patterns, $R := \frac{1}{2} \mathrm{Min}_{\mu, \nu \in [M]} \lVert \boldsymbol{\xi}_\mu - \boldsymbol{\xi}_\nu \rVert$
S_μ	The sphere centered at the memory pattern $\boldsymbol{\xi}_\mu$ with finite radius R
$\mathbf{x}_\mu^\star$	The fixed point of $\mathcal{T}$ covered by S_μ, i.e. $\mathbf{x}_\mu^\star \in S_\mu$
Δ_μ	The separation of a memory pattern $\boldsymbol{\xi}_\mu$ from all other memory patterns $\boldsymbol{\Xi}$, defined in (3.1)
$\widetilde{\Delta}_\mu$	The separation of $\boldsymbol{\xi}_\mu$ at a given $\mathbf{x}$ from all memory patterns $\boldsymbol{\Xi}$, defined in (3.2)

B Broader Impacts and Future Directions: Brain Science and Foundation Models

The primary theme of our research is to perceive any data representation (set of patterns) as analogous to the neural responses of a global brain reacting to a vast range of external stimuli (queries). This perspective presents exciting opportunities to study large generative foundational models, such as large language models, within a rigorous scientific framework inspired by contemporary brain science research.

We believe this work could be impactful in several respects, even though it is foundational research and not tied to specific applications: **(Cognition.)** This research could contribute to our understanding of a memory-enhanced model's predictive capacity when given either in-context input (like historical data) or external stimuli (such as real-time events). **(Memory.)** It may also shed light on the inherent limits of artificial neural networks' memorization capabilities and how to augment them with external memory modules for rapid responses to potential external stimuli. **(Network.)** This research could enable models to better assess the intricate network of cross-sectional brain activity among different variables and infer its dynamic structural alterations to identify possible systematic properties.

C Related Works and Limitations

Sparse Hopfield Models. Our work is closely related to and motivated by [Földiak, 1990], which proposes a local anti-Hebbian learning rule for sparse representations in associative memory networks. This rule enhances storage capacity and retrieval capabilities but has limitations: (i) fixed sparsity based on local similarity of receptive fields, (ii) difficulty in scaling up and integration with modern DNNs [Makhzani and Frey, 2015], (iii) lack of a solid theoretical foundation for convergence and stability, and (iv) inherently unsupervised retrieval dynamics, limiting its applicability for supervised learning or other paradigms like reinforcement learning or semi-supervised learning. On the other hand, another line of related work, not specifically focusing on sparsifying Hopfield models, centers on sparse coding [Palm, 2013, Olshausen and Field, 1997], introducing sparsity to associative memory models through thresholding memory patterns. These studies offer insights into the relationship between the sparseness of the stored memory patterns and the robustness of the network but sufferers from the issues related to scalability, sparsity level bias, and noise vulnerability [Mairal et al., 2010, Rubinstein et al., 2010, Elad, 2010, Olshausen and Field, 1997]. In contrast, our approach is theoretically grounded and has data-dependent sparsity leading to better scalability, more meaningful and robust representations of patterns and allows the model to focus on the most relevant information for each specific instance.

Hopfield Models and Connection to Attention. Hopfield Models [Hopfield, 1984, 1982, Krotov and Hopfield, 2016] have seen renewed interest in the machine learning community due to advances in memory storage capacity understanding [Krotov and Hopfield, 2016, Demircigil et al., 2017], architectural innovations [Hoover et al., 2023, Seidl et al., 2022, Fürst et al., 2022, Ramsauer et al., 2021], and biological plausibility [Kozachkov et al., 2023, Krotov and Hopfield, 2021]. Notably, Modern Hopfield Networks [Ramsauer et al., 2021][8], a new subclass, highlight the equivalence[9] between their memory retrieval dynamics and attention mechanisms in transformers. With this hindsight, it becomes clear that transformers and modern Hopfield models share some high-level similarities, as well as differences. Both architectures are designed for denoising input, with transformers typically pre-trained on masked-token tasks and the modern Hopfield model aimed at completing incomplete or contaminated patterns. However, the modern Hopfield models are recurrent networks with a global energy function that ensures convergence to a fixed-point attractor, while transformers are generally viewed as feed-forward networks without such dynamics. It is natural to ask whether such equivalence is fundamental. Although, apart from Hopfield-side investigations [Hoover et al., 2023, Krotov and Hopfield, 2021, Ramsauer et al., 2021], there have been studies viewing transformers as dynamical systems, including the deep equilibrium models [Bai et al., 2019], and unfolded optimization [Yang et al., 2022], none exhibit similar converge-to-memory dynamics as in Hopfield models (hence missing the connection between dynamical memory retrieval and transformers), nor do they address sparsity. Building on the established equivalence in [Ramsauer et al., 2021], our work serves as an initial attempt to push such equivalence toward sparse models, both theoretically and empirically. It lays the groundwork for future Hopfield-based methodologies, architecture designs and biological computers (as in [Kozachkov et al., 2023]).

Sparse Attention. Attention-based seq2seq models excel in various applications like large language models [Chowdhery et al., 2022, Brown et al., 2020], time series prediction [Zhou et al., 2022, 2021], and biomedical science [Ji et al., 2021], primarily due to their versatility in framing tasks as source-to-target sequence transformations with potentially varying lengths. However, the original transformer architecture utilizes a dense, quadratic attention score matrix, which can be computationally demanding (with $\mathcal{O}(n^2)$ complexity for input sequence length n), memory-intensive, and challenging to interpret for long sequences. To combat these issues, there is a large amount of literature works leverages various sparsifying methods for attention and transformers to enhance computational efficiency while preserving the models' expressiveness, see [Tay et al., 2022] for an overview. Here, we classify sparse Transformers into two distinct categories based on the different kinds of sparsities. The first category focuses on structured-sparsity [Beltagy et al., 2020, Qiu et al., 2019, Child et al., 2019], which involves creating a sparse attention score matrix in a pre-determined manner. In these approaches, each token in the sequence attends to a fixed subset of other tokens, rather than the entire sequence. The second category obtains sparsity through the sparsity-inducing normalization maps [Peters et al., 2019, Correia et al., 2019, Krotov and Hopfield, 2016] that encourage the models to focus on a subset of relevant input elements, thereby fostering sparsity, scalability and interpretability. Compared to the first category, while these approaches still have $\mathcal{O}(n^2)$ space complexity, they offer the advantage of producing sparsity patterns that are more adaptive to the data. Our work is closely related to the second and utilizes *sparsity-inducing alternatives* to the softmax function in modern Hopfield models.

C.1 Limitations

Since our model aligns with sparsemax attention, it also grapples with $\mathcal{O}(d^2)$ complexity, a characteristic typical of the sparsity-inducing normalization map category of sparse attention. In addition, we opt not to impose any assumptions on the data (patterns) to maintain the general applicability of

[8] Also see the well-written blog post [Brandstetter, 2021].

[9] While this equivalence only holds when the retrieval dynamics is applied exactly once, as originally shown in [Ramsauer et al., 2021] and later emphasized in [Krotov and Hopfield, 2021], it allows us to view modern Hopfield models as generalized attentions with additional functionalities and hence opens new avenues for Hopfield-based architecture designs.

our model. This decision, however, prevents us from providing a rigorous characterization of how data-dependent sparsity explicitly impacts retrieval error, the well-separation condition, and memory capacity. Specifically, a detailed analysis of $\left[\Xi^\mathsf{T}\mathbf{x}\right]_{(\kappa)}$ is a problem of order statistics [David and Nagaraja, 2004] that hinges on the distribution of patterns. Instead, we offer qualitative discussions in Section 3 to provide insights into the behavior of the sparse model under various conditions, aiding in a better understanding and application of the model.

D Modern Hopfield Model and Its Connection to Attention Mechanism

Ramsauer et al. [2021] generalize the exponential-interaction-based energy function proposed in [Demircigil et al., 2017] to continuous patterns and states with a strong link the attention mechanism. In this section, we provide an overview for both of them and then draw the connection of the modern Hopfield model to attention mechanism.

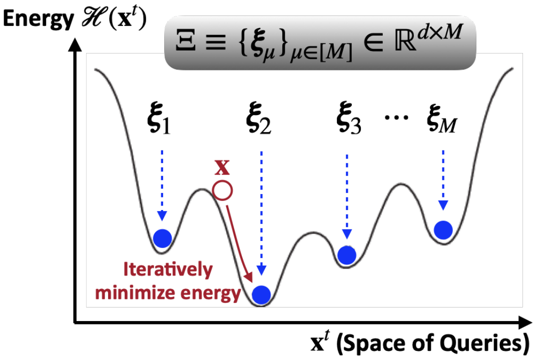
Energy $\mathscr{H}(\mathbf{x}^t)$
$\Xi \equiv \{\boldsymbol{\xi}_\mu\}_{\mu\in[M]} \in \mathbb{R}^{d\times M}$
$\boldsymbol{\xi}_1$
$\boldsymbol{\xi}_2$
$\boldsymbol{\xi}_3$ ··· $\boldsymbol{\xi}_M$
$\mathbf{x}$
Iteratively
minimize energy
$\mathbf{x}^t$ (Space of Queries)

Figure 3: **Visualizing Hopfield Models.** Let $\mathbf{x} \in \mathbb{R}^d$ represent the query pattern, and let $\Xi := (\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_M) \in \mathbb{R}^{d\times M}$ denote the memory patterns. The objective of the Hopfield models is to store the memory patterns Ξ and then retrieve a specific memory pattern $\boldsymbol{\xi}_\mu$ based on a given query $\mathbf{x}$. They achieve these by embedding the memories Ξ in the energy landscape $\mathcal{H}(\mathbf{x})$ of a physical system (e.g., the Ising model in [Hopfield, 1982] and its higher-order generalizations [Lee et al., 1986, Peretto and Niez, 1986, Newman, 1988]), where each memory $\boldsymbol{\xi}_\mu$ corresponds to a local minimum. When a query $\mathbf{x}$ is presented, the model initiates energy-minimizing retrieval dynamics $\mathcal{T}$ at the query $\mathbf{x}$, which then navigate the energy landscape to find the nearest local minimum, effectively retrieving the memory most similar to the query.

D.1 Modern Hopfield Model

We first introduce the log-sum-exponential (lse) function for any given vector $\mathbf{z} = (z_1, \cdots, z_M)$ and $\beta > 0$:

$$\text{lse}\left(\beta, \mathbf{z}\right) := \frac{1}{\beta}\log\left(\sum_{\mu=1}^{M}\exp\{\beta z_\mu\}\right), \tag{D.1}$$

which is an important representation of the softmax function which can be derived by considering the "argmax function" under entropy regularization, see [Gao and Pavel, 2017] and references therein.

Exponential Binary Hopfield Model. With $\text{lse}(\cdot)$, the exponential-Hopfield model for binary patterns $\boldsymbol{\xi}, \mathbf{x} \in \{\pm 1\}^d$ proposed in [Demircigil et al., 2017] can be written as, denoting $\Xi := \left(\boldsymbol{\xi}^1, \cdots, \boldsymbol{\xi}^M\right)$,

$$\mathcal{H}(\mathbf{x}) = -\sum_{\mu=1}^{M}\exp\{\langle\boldsymbol{\xi}^\mu, \boldsymbol{\sigma}\rangle\} = -\exp\left\{\text{lse}\left(1, \Xi^\mathsf{T}\boldsymbol{\sigma}\right)\right\}, \tag{D.2}$$

which leads to the super-linear memory capacity of $M \propto 2^{d/2}$.

Modern Hopfield Model. For continuous patterns $\mathbf{x}, \{\boldsymbol{\xi}^\mu\} \in \mathbb{R}^d$, Ramsauer et al. [2021] propose the continuous[10] modern Hopfield model

$$\mathcal{H}(\mathbf{x}) \coloneqq -\operatorname{lse}\left(1, \boldsymbol{\Xi}^\top \mathbf{x}\right) + \frac{1}{2}\langle \mathbf{x}, \mathbf{x}\rangle + \frac{1}{\beta}\log M + \frac{1}{2}m^2, \tag{D.3}$$

with retrieval dynamics

$$\mathbf{x}^{\text{new}} = \mathcal{T}_{\text{Dense}}(\mathbf{x}) = \boldsymbol{\Xi} \cdot \operatorname{Softmax}\left(\beta \boldsymbol{\Xi}^\top \mathbf{x}\right), \tag{D.4}$$

where $\frac{1}{2}\langle \mathbf{x}, \mathbf{x}\rangle$ is a regularizer introduced for ensuring configuration vector $\mathbf{x}$ being finite, and $m \coloneqq \max_\mu \|\boldsymbol{\xi}^\mu\|$ is the largest norm of memory patterns. Moreover, they show that (i) the modern Hopfield (D.3) has an exponential memory capacity in d, (ii) the retrieval dynamics (D.4) can consistently retrieve patterns with high accuracy with only one step, and (iii) surprisingly, the retrieval dynamics (D.4) is connected to the attention mechanism in transformer giving rise to a new methodology — the Hopfield DNN layer.

D.2 Memory Retrieval Dynamics $\mathcal{T}_{\text{Dense}} \leftrightarrow$ Self-Attention Mechanism

Following [Ramsauer et al., 2021, Brandstetter, 2021], we say $\mathbf{X}$ and $\boldsymbol{\Xi}$ are in the associative space (embedded space), as they are mapped from the *raw* query $\mathbf{R}$ and $\mathbf{Y}$ memory patterns, respectively, via

$$\mathbf{X}^\top = \mathbf{R}\mathbf{W}_Q \coloneqq \mathbf{Q}, \tag{D.5}$$

$$\boldsymbol{\Xi}^\top = \mathbf{Y}\mathbf{W}_K \coloneqq \mathbf{K}, \tag{D.6}$$

with some $\mathbf{W}_Q$ and $\mathbf{W}_K$. Therefore, we can express $\mathcal{T}_{\text{Dense}}$ as

$$\left(\mathbf{Q}^{\text{new}}\right)^\top = \mathbf{K}^\top \operatorname{Softmax}\left(\beta \mathbf{K}\mathbf{Q}^\top\right). \tag{D.7}$$

Taking transpose to above, we have

$$\mathbf{Q}^{\text{new}} = \operatorname{Softmax}\left(\beta \mathbf{Q}\mathbf{K}^\top\right)\mathbf{K}. \tag{D.8}$$

Projecting $\mathbf{K}$ to $\mathbf{V}$ with $\mathbf{W}_V$, we have

$$\boldsymbol{Z} \coloneqq \mathbf{Q}^{\text{new}}\mathbf{W}_V = \operatorname{Softmax}\left(\beta \mathbf{Q}\mathbf{K}^\top\right)\mathbf{K}\mathbf{W}_V \tag{D.9}$$

$$= \operatorname{Softmax}\left(\beta \mathbf{Q}\mathbf{K}^\top\right)\mathbf{V}, \tag{D.10}$$

which leads to the self-attention mechanism.

Plugging back the raw patterns $\mathbf{R}$ and $\mathbf{Y}$, we arrive the foundation of the Hopfield layer,

$$\boldsymbol{Z} = \operatorname{Softmax}\left(\beta \mathbf{R}\mathbf{W}_Q\mathbf{W}_K^\top\mathbf{Y}^\top\right)\mathbf{Y}\mathbf{W}_K\mathbf{W}_V. \tag{D.11}$$

The same construction applies to the sparse retrieval dynamics (2.5),

$$\boldsymbol{Z}' = \operatorname{Sparsemax}\left(\beta \mathbf{R}\mathbf{W}'_Q\mathbf{W}'^\top_K\mathbf{Y}^\top\right)\mathbf{Y}\mathbf{W}'_K\mathbf{W}'_V. \tag{D.12}$$

resulting in a sparse Hopfield layer that can be seamlessly integrated into deep learning architectures.

D.3 Algorithm of Multi-Step `SparseHopfield` Layer

Here, we present an algorithm for implementing the `SparseHopfield` layer with multi-step updates (i.e. multiple iterative retrievals). The algorithm, summarized in Algorithm 1 below, outlines the process for U update steps. Similar to [Ramsauer et al., 2021], the `SparseHopfield` takes as input the matrices $\mathbf{R}$, $\mathbf{Y}$, and the weight matrices $\mathbf{W}'_Q, \mathbf{W}'_K, \mathbf{W}'_V$.

Here we explain the usage of the above algorithm w.r.t. different settings.

1. **Memory Retrieval.** The memory retrieval is a learning-free setting. Thus, we can exclude the use of weight matrices $\mathbf{W}_K, \mathbf{W}_Q, \mathbf{W}_V$ (by setting them to identity matrices). And let the input (corrupted image) to be our $\mathbf{R}$, stored patterns as $\mathbf{Y}$ for retrieval.

[10] Note that, there are also many continuous Hopfield models prior than [Ramsauer et al., 2021], including [Krotov and Hopfield, 2016, Hopfield, 1984].

Algorithm 1 Multi-Step `SparseHopfield` Layer

```
Require: $U \in \mathbb{R} \geq 1, \mathbf{R}, \mathbf{Y}$.
  $\mathbf{Q} \leftarrow \mathbf{R}\mathbf{W}'_Q$
  for $i \rightarrow 1$ to $U$ do
      $\mathbf{Q}^{\text{new}} \leftarrow \text{Sparsemax}\left(\beta \mathbf{Q} \mathbf{W}'^T_K \mathbf{Y}^T\right) \mathbf{Y}\mathbf{W}'_V \mathbf{W}'_K$        Hopfield Update as D.12
      $\mathbf{Q} \leftarrow \mathbf{Q}^{\text{new}}$
  end for
  return $\mathbf{Q}$
```

2. `SparseHopfield`**.** The `SparseHopfield` has two inputs, $\mathbf{R}, \mathbf{Y}$. Since the `SparseHopfield` can be used to replace attention mechanism in models, we make the weight matrices $\mathbf{W}'_K, \mathbf{W}'_Q, \mathbf{W}'_V$ learnable, and $\mathbf{R}$, $\mathbf{Y}$, $\mathbf{Y}$ be the source of query, key, value, respectively. Note that the self-attention-liked mechanism can be realized by setting $\mathbf{R} = \mathbf{Y}$.
3. `SparseHopfieldPooling`**.** The `SparseHopfieldPooling` layer has one input, $\mathbf{Y}$, where $\mathbf{Q}$ is the learnable **prototype pattern** and fixed during inference, and $\mathbf{Y}$ is the stored patterns we want to perform pooling over. Note that the $\mathbf{Q}$ here is independent from the input and can be seen as part of the learnable parameter of the `SparseHopfieldPooling` layer. Here since we replace the query pattern ($\mathbf{R}\mathbf{W}'_Q$) with a static **prototype pattern** $\mathbf{Q}$, the learnable weight matrices here will only be $\mathbf{W}'_K, \mathbf{W}'_V$.
4. `SparseHopfieldLayer`**.** The `SparseHopfieldLayer` layer has one input, $\mathbf{R}$. Where $\mathbf{R}$ is the query pattern. And we have learnable weight matrices $\mathbf{W}'_K, \mathbf{W}'_V$ served as our stored patterns and pattern projections, leading our key and value independent to the input. In other words, following the notation in Algorithm 1, $\mathbf{Y}$ can be seen as an identity matrix.

E Proofs of Main Text

E.1 Theorem 2.1

Proof of Theorem 2.1.

$$\begin{aligned}
\underset{\mathbf{p}\in\Delta^d}{\text{Max}}\left[\langle\mathbf{p},\mathbf{z}\rangle-\frac{1}{2}\|\mathbf{p}\|^2+\frac{1}{2}\right] &= \underset{\mathbf{p}\in\Delta^d}{\text{Max}}\left[\frac{1}{2}\|\mathbf{z}\|^2+\langle\mathbf{p},\mathbf{z}\rangle-\frac{1}{2}\|\mathbf{p}\|^2-\frac{1}{2}\|\mathbf{z}\|^2+\frac{1}{2}\right] && \text{(E.1)}\\
&= \underset{\mathbf{p}\in\Delta^d}{\text{Max}}\left[\frac{1}{2}\|\mathbf{z}\|^2+\frac{1}{2}-\frac{1}{2}\|\mathbf{p}-\mathbf{z}\|^2\right] && \text{(E.2)}\\
&= \frac{1}{2}\|\mathbf{z}\|^2+\frac{1}{2}-\underset{\mathbf{p}\in\Delta^d}{\text{Min}}\left[\frac{1}{2}\|\mathbf{p}-\mathbf{z}\|^2\right] && \text{(E.3)}\\
&= \frac{1}{2}\|\mathbf{z}\|^2-\frac{1}{2}\|\mathbf{p}^\star-\mathbf{z}\|^2+\frac{1}{2}=\Psi^\star(\mathbf{z}), && \text{(E.4)}
\end{aligned}$$

with $\mathbf{p}^\star$ given by (2.3). □

E.2 Lemma 2.1

Proof of Lemma 2.1. To show monotonic decreasing property of the energy (2.1), we first derive the sparse retrieval dynamics by utilizing the aforementioned Theorem 2.1, Corollary 2.1.1, along with the convex-concave procedure [Yuille and Rangarajan, 2003, 2001]. Then, we show the monotonicity of $\mathcal{H}$ by constructing an iterative upper bound of $\mathcal{H}$ which is convex in $\mathbf{x}_{t+1}$ and thus, lowered iteratively by the CCCP method.

By convex conjugate, Ψ^*, the conjugate convex of Ψ, is always convex, and hence $-\Psi^*$ is a concave function. Therefore, the energy function $\mathcal{H}$ is by construction the sum of the convex function $\mathcal{H}_1(\mathbf{x}) := \frac{1}{2}\langle\mathbf{x},\mathbf{x}\rangle$ and the concave function $\mathcal{H}_2(\mathbf{x}) := -\Psi^\star\left(\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)$. In addition, $\mathcal{H}$ is differentiable by definition.

Applying the convex-concave procedure to $\mathcal{H}$ gives

$$\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_1\left(\mathbf{x}_{t+1}\right)=-\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2\left(\mathbf{x}_t\right), \tag{E.5}$$

which leads to

$$\mathbf{x}_{t+1}=\boldsymbol{\nabla}_\mathbf{x}\Psi\left(\boldsymbol{\Xi}\mathbf{x}_t\right)=\boldsymbol{\Xi}\,\text{Sparsemax}\left(\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}_t\right), \tag{E.6}$$

by Theorem 2.1 and Corollary 2.1.1.

Following [Yuille and Rangarajan, 2003, 2001], we show the monotonic decreasing of (2.1) over t with by considering the problem of energy minimization:

$$\underset{\mathbf{x}}{\text{Min}}\left[\mathcal{H}(\mathbf{x})\right] \quad = \quad \underset{\mathbf{x}}{\text{Min}}\left[\mathcal{H}_1(\mathbf{x})+\mathcal{H}_2(\mathbf{x})\right], \tag{E.7}$$

which, in the convex-concave procedure, is solved by iteratively computing

$$\mathbf{x}_{t+1} \quad \in \quad \underset{\mathbf{x}}{\text{ArgMin}}\left[\mathcal{H}_1(\mathbf{x})+\langle\mathbf{x},\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2\left(\mathbf{x}_t\right)\rangle\right], \tag{E.8}$$

for all t. The intuition behind this is to linearize the concave $\mathcal{H}_2$ around the current iteration's solution $\mathbf{x}_t$, making $\mathcal{H}_1(\mathbf{x}_{t+1})+\langle\mathbf{x}_{t+1},\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2(\mathbf{x}_t)\rangle$ convex in $\mathbf{x}_{t+1}$.

By convexity and concavity of $\mathcal{H}_1$ and $\mathcal{H}_2$, we have

$$\begin{aligned}
\mathcal{H}_1(\mathbf{x}) &\geq \mathcal{H}_1(\mathbf{y})+\langle(\mathbf{x}-\mathbf{y}),\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_1(\mathbf{y})\rangle, && \text{(E.9)}\\
\mathcal{H}_2(\mathbf{x}) &\leq \mathcal{H}_2(\mathbf{y})+\langle(\mathbf{x}-\mathbf{y}),\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2(\mathbf{y})\rangle, && \text{(E.10)}
\end{aligned}$$

for all $\mathbf{x},\mathbf{y}$. Therefore, it holds

$$\begin{aligned}
\mathcal{H}(\mathbf{x}) &= \mathcal{H}_1(\mathbf{x})+\mathcal{H}_2(\mathbf{x}) && \text{(E.11)}\\
&\leq \mathcal{H}_1(\mathbf{x})+\mathcal{H}_2(\mathbf{y})+\langle(\mathbf{x}-\mathbf{y}),\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2(\mathbf{y})\rangle := \mathcal{H}_U\left(\mathbf{x},\mathbf{y}\right), && \text{(E.12)}
\end{aligned}$$

where $\mathcal{H}_U$ is the upper bound of $\mathcal{H}$. Then, for each iteration t, we have

$$\mathbf{x}_{t+1}\in\underset{\mathbf{x}}{\text{ArgMin}}\left[\mathcal{H}_U(\mathbf{x},\mathbf{x}_t)\right]=\underset{\mathbf{x}}{\text{ArgMin}}\left[\mathcal{H}_1(\mathbf{x})+\langle\mathbf{x},\boldsymbol{\nabla}_\mathbf{x}\mathcal{H}_2(\mathbf{x}_t)\rangle\right], \tag{E.13}$$

which lowers the upper bound $\mathcal{H}_U$ iteratively and hence decreases the value of $\mathcal{H}$ monotonically, i.e.

$$\begin{aligned}
\mathcal{H}(\mathbf{x}_{t+1}) &\leq \mathcal{H}_U(\mathbf{x}_{t+1}, \mathbf{x}_t) && \text{(By (E.12))}\\
&\leq \mathcal{H}_U(\mathbf{x}_t, \mathbf{x}_t) && \text{(Set } \mathbf{x}=\mathbf{y} \text{ in (E.12))}\\
&= \mathcal{H}(\mathbf{x}_t), && \text{(E.14)}
\end{aligned}$$

for all t. This completes the proof that $\mathcal{H}$ can be monotonically decreased by $\mathcal{T}(\mathbf{x})$ given by (2.5). □

E.3 Theorem 2.2

Proof of Theorem 2.2. Let $\mathcal{T}_{\text{Dense}}$ be the retrieval dynamics of the dense modern Hopfield model [Ramsauer et al., 2021], and $\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ and $\|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ be the retrieval error of sparse and dense Hopfield model, respectively.

We observe

$$\begin{aligned}
&\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| - \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|\\
&= \left\|\sum_{\nu=1}^{\kappa} \boldsymbol{\xi}_\nu \left[\mathrm{Sparsemax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)\right]_\nu - \boldsymbol{\xi}_\mu\right\| - \left\|\sum_{\nu=1}^{\kappa} \boldsymbol{\xi}_\nu \left[\mathrm{Softmax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)\right]_\nu - \boldsymbol{\xi}_\mu\right\| && \text{(E.15)}\\
&\leq \left\|\sum_{\nu=1}^{\kappa} \left[\mathrm{Sparsemax}(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x})\right]_\nu \boldsymbol{\xi}_\nu\right\| - \left\|\sum_{\nu=1}^{\kappa} \left[\mathrm{Softmax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)\right]_\nu \boldsymbol{\xi}_\nu\right\| && \text{(E.16)}\\
&\leq 0, && \text{(E.17)}
\end{aligned}$$

which gives

$$\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \quad\leq\quad \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|. \tag{E.18}$$

Next, we provide an upper bound of the sparse retrieval error for a query $\mathbf{x} \in S_\mu$ given memory patterns $\{\boldsymbol{\xi}_\nu\}_{\nu\in[M]}$.

According to the (2.3), it holds

$$[\mathrm{Sparsemax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)]_\mu \leq \left[\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_\mu - \left[\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)} + \frac{1}{\kappa}, \tag{E.19}$$

for all $\mu \in [M]$. Then, the sparse retrieval error is

$$\begin{aligned}
\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}^\mu\| = \left\|\boldsymbol{\Xi}\,\mathrm{Sparsemax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right) - \boldsymbol{\xi}^\mu\right\| &= \left\|\sum_{\nu=1}^{\kappa} \boldsymbol{\xi}_{(\nu)} \left[\mathrm{Sparsemax}\left(\beta\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right)\right]_{(\nu)} - \boldsymbol{\xi}^\mu\right\|\\
&\leq m + m\beta\left\|\sum_{\nu=1}^{\kappa}\left(\left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\nu)} - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)} + \frac{1}{\beta\kappa}\right)\frac{\boldsymbol{\xi}_{(\nu)}}{m}\right\| && \text{(By (E.19))}\\
&= m + d^{1/2}m\beta\left[\sum_{\nu=1}^{\kappa}\left(\left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\nu)} - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)} + \frac{1}{\beta\kappa}\right)\right] && \text{(E.20)}\\
&\leq m + d^{1/2}m\beta\left[\kappa\left(\underset{\nu\in[M]}{\mathrm{Max}}\,\langle\boldsymbol{\xi}_\nu, \mathbf{x}\rangle - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right]. && \text{(E.21)}
\end{aligned}$$

□

E.4 Lemma 2.2

In order to prove Lemma 2.2, we need the following two auxiliary lemmas.

Lemma E.1 ([Gunawardana et al., 2005], Proposition 7)**.** Let $\mathbf{x}_t \in \mathcal{X}_t$ and $\mathbf{x}_{t+1} \in \mathcal{X}_{t+1}$. Given a real-valued continuous function $\mathcal{H}_U$ on $\mathcal{X}_t \times \mathcal{X}_{t+1}$, define the point-to-set map $\mathcal{T} : \mathcal{X}_t \to \mathcal{X}_{t+1}$ by

$$\begin{aligned}
\mathcal{T}(\mathbf{x}_t) \coloneqq \underset{\mathbf{x}'_{t+1}\in\mathcal{X}_{t+1}}{\mathrm{ArgMin}}\ \mathcal{H}_U(\mathbf{x}_t, \mathbf{x}'_{t+1}) && \text{(E.22)}\\
= \{\mathbf{x}_{t+1} \mid \mathcal{H}_U(\mathbf{x}_t, \mathbf{x}_{t+1}) \leq \mathcal{H}_U(\mathbf{x}_t, \mathbf{x}'_{t+1}), \forall \mathbf{x}'_{t+1} \in \mathcal{X}_{t+1}\}. && \text{(E.23)}
\end{aligned}$$

Then $\mathcal{T}$ is a closed map at $\mathbf{x}_t$ if $\mathcal{T}(\mathbf{x})$ is non-empty.

Lemma E.2 ([Sriperumbudur and Lanckriet, 2009], Lemma 5)**.** Recall a fixed point of $\mathcal{T}$ w.r.t. $\mathcal{H}$ is a point for which $\mathbf{x} = \mathcal{T}(\mathbf{x})$, and a generalized fixed point is a point for which $\mathbf{x} \in \mathcal{T}(\mathbf{x})$. Suppose $\mathbf{x}^\star$ is a generalized fixed point of $\mathcal{T}$, then, $\mathbf{x}^\star$ is a stationary point of the minimization problem (E.7).

Proof of Lemma 2.2. From Zangwill global convergence theory for iterative algorithms [Zangwill, 1969], all limit points of $\{\mathbf{x}_t\}_{t=0}^\infty$ are fixed points[11], if the following three conditions are satisfied for the energy function $\mathcal{H}$ and the retrieval dynamics $\mathcal{T}$.

(i) For any sequence $\{\mathbf{x}_t\}_{t=0}^\infty$ with starting point $\mathbf{x}_0 \in S_\mu$, all points are in the same compact set S_μ.

(ii) $\mathcal{H}$ is monotonically decreased by $\mathcal{T}(\mathbf{x})$, i.e. $\mathcal{H}(\mathbf{x}_{t+1}) \leq \mathcal{H}(\mathbf{x}_t), \forall \mathbf{x}_{t+1} = \mathcal{T}(\mathbf{x}_t)$.

(iii) For all t, if $\mathcal{H}(\mathbf{x}_{t+1}) < \mathcal{H}(\mathbf{x}_t)$, $\mathcal{T}$ is closed at $\mathbf{x}_t$.

Furthermore, $\lim_{t\to\infty} \mathcal{H}\left(\mathbf{x}_t\right) = \mathcal{H}\left(\mathbf{x}^\star\right)$ for all limit points $\mathbf{x}^\star$.

From Definition 2.2, since S_μ with finite radius R is bounded and closed, every S_μ is a compact set. Namely, for any sequence $\{\mathbf{x}_t\}_{t=0}^\infty$, all points are embedded in the sphere S_μ, which is a compact set. Therefore, condition (i) is automatically satisfied. Then condition (ii), the monotonic descent property of $\{\mathbf{x}_t\}_{t=0}^\infty$, has been analyzed in the original paper of CCCP [Yuille and Rangarajan, 2003]. By our definition on $\mathcal{H}_1$ and $\mathcal{H}_2$, we have $\mathcal{H}_U\left(\mathbf{x}, \mathbf{y}\right) \coloneqq \mathcal{H}_1(\mathbf{x}) + \mathcal{H}_2(\mathbf{y}) + \langle (\mathbf{x} - \mathbf{y}), \boldsymbol{\nabla}_\mathbf{x} \mathcal{H}_2(\mathbf{y}) \rangle$ is continuous in $\mathbf{x}$ and $\mathbf{y}$. Consequently, by Lemma E.1, the non-empty assumption of the point-to-set map $\mathcal{T}$ guarantees that $\mathcal{T}$ is closed at $\mathbf{x}_t$ and so satisfies condition (iii) for generalized fixed points. Therefore, by Zangwill global convergence theory, all the limit points of $\{\mathbf{x}_t\}_{t=0}^\infty$ are generalized fixed points and $\lim_{t\to\infty} \mathcal{H}\left(\mathbf{x}_t\right) = \mathcal{H}\left(\mathbf{x}^\star\right)$, where $\mathbf{x}^\star$ are some generalized fixed points of $\mathcal{T}$. Furthermore, based on the results of Lemma E.2, $x^\star$ are also the stationary points of the minimization problem (E.7). Therefore, the energy function is ensured to converge to local optimum. □

[11] Recall that, a fixed point of $\mathcal{T}$ is defined as $\mathbf{x}^\star \coloneqq \{\mathbf{x} \mid \mathbf{x} = \mathcal{T}(\mathbf{x}^\star)\}$.

E.5 Theorem 3.1 and Corollary 3.1.1

Proof of Theorem 3.1. Recall $n \coloneqq \|\mathbf{x}\|$. By Definition 3.1, we have

$$\mathop{\mathrm{Max}}_{\mu\in[M]} \langle \boldsymbol{\xi}_\mu, \mathbf{x}\rangle = \langle \boldsymbol{\xi}_\nu, \mathbf{x}\rangle - \widetilde{\Delta}_\nu, \tag{E.24}$$

thereby obtaining

$$\begin{aligned} \|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| &\leq m + d^{1/2} m\beta \left[\kappa\left(\mathop{\mathrm{Max}}_{\nu\in[M]} \langle \boldsymbol{\xi}_\nu, \mathbf{x}\rangle - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right] && \text{(E.25)} \\ &= m + d^{1/2} m\beta \left[\kappa\left(\langle \boldsymbol{\xi}_\mu, \mathbf{x}\rangle - \widetilde{\Delta}_\mu - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right] && \text{(E.26)} \end{aligned}$$

Since $n \coloneqq \|\mathbf{x}\|$ and $m \coloneqq \max_\mu \|\boldsymbol{\xi}^\mu\|$, we have

$$\begin{aligned} & m + d^{1/2} m\beta \left[\kappa\left(\langle \boldsymbol{\xi}_\mu, \mathbf{x}\rangle - \widetilde{\Delta}_\mu - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right] && \text{(E.27)} \\ & \leq m + d^{1/2} m\beta \left[\kappa\left(mn - \widetilde{\Delta}_\mu - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right] && \text{(E.28)} \end{aligned}$$

Then, by the Cauchy-Schwartz inequality

$$|\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu\rangle - \langle \mathbf{x}, \boldsymbol{\xi}_\mu\rangle| \leq \|\boldsymbol{\xi}_\mu - \mathbf{x}\| \cdot \|\boldsymbol{\xi}_\mu\| \leq \|\boldsymbol{\xi}_\mu - \mathbf{x}\| m, \quad \forall \mu \in [M], \tag{E.29}$$

we observe that $\widetilde{\Delta}_\mu$ can be expressed in terms of Δ_μ:

$$\begin{aligned} \widetilde{\Delta}_\mu &= \mathop{\mathrm{Min}}_{\nu,\nu\neq\mu} \left[\langle \mathbf{x}, \boldsymbol{\xi}_\mu\rangle - \langle \mathbf{x}, \boldsymbol{\xi}_\nu\rangle + \left(\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu\rangle - \langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu\rangle\right) - \left(\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu\rangle - \langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu\rangle\right)\right] && \text{(E.30)} \\ &\geq \mathop{\mathrm{Min}}_{\nu,\nu\neq\mu} \left[\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu\rangle - \langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu\rangle + \left(\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\nu\rangle - \langle \mathbf{x}, \boldsymbol{\xi}_\nu\rangle\right) - \left(\langle \boldsymbol{\xi}_\mu, \boldsymbol{\xi}_\mu\rangle - \langle \mathbf{x}, \boldsymbol{\xi}_\mu\rangle\right)\right] \\ & && \text{(By Cauchy-Schwarz)} \\ &= \Delta_\mu - 2\|\boldsymbol{\xi}_\mu - \mathbf{x}\| m = \Delta_\mu - 2mR, && \text{(By } \mathbf{x} \in S_\mu\text{)} \end{aligned}$$

where R is radius of the sphere S_μ. Inserting the bound on $\widetilde{\Delta}_\mu$, we obtain

$$\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq m + d^{1/2} m\beta \left[\kappa\left(mn - \Delta_\mu + 2mR - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right]. \tag{E.31}$$

For $\mathcal{T}$ to be a mapping from S_μ to S_μ, we obtain the inequality:

$$m + d^{1/2} m\beta \left[\kappa\left(mn - \Delta_\mu + 2mR - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)}\right) + \frac{1}{\beta}\right] \leq R, \tag{E.32}$$

which gives

$$\Delta_\mu \geq mn + 2mR - \left[\boldsymbol{\Xi}^\mathsf{T}\mathbf{x}\right]_{(\kappa)} - \frac{1}{\kappa}\left(\frac{R - m - md^{1/2}}{m\beta d^{1/2}}\right). \tag{E.33}$$

Therefore, as long as Δ_μ satisfies this inequality, $\mathcal{T}$ is a mapping from S_μ onto itself. □

Proof of Corollary 3.1.1. Let $\mathcal{T}_{\text{Dense}}$ be the retrieval dynamics of the dense modern Hopfield model [Ramsauer et al., 2021], and $\epsilon_{\text{Sparsemax}} \coloneqq \|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ and $\epsilon_{\text{Dense}} \coloneqq \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|$ be the retrieval error of sparse and dense Hopfield model, respectively.

First, let's recall Theorem 2.2, which states that

$$\epsilon_{\text{Sparsemax}} = \|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \quad \leq \quad \epsilon_{\text{Dense}} = \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|. \tag{E.34}$$

Next we want to find the lower bound of separation Δ_μ such that $\mathcal{T}$ is a mapping from S_μ onto S_μ.

To link Δ_μ to $\mathcal{T}$, we first bound ϵ_{Dense} via [Ramsauer et al., 2021, Lemma A.4]:

$$
\begin{aligned}
\epsilon_{\text{Dense}} &= \left\|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\right\| \qquad &\text{(E.35)}\\
&= \left\|\boldsymbol{\xi}_\mu - \sum_{\nu=1}^{M}[\mathrm{Softmax}(\beta\Xi^{\mathsf{T}}\mathbf{x})]_\nu \boldsymbol{\xi}_\nu\right\| &\text{(E.36)}\\
&= \left\|\left(1-\left[\mathrm{Softmax}(\beta\Xi^{\mathsf{T}}\mathbf{x})\right]_\mu\right)\boldsymbol{\xi}_\mu - \sum_{\nu=1,\nu\neq\mu}^{M}\left[\mathrm{Softmax}(\beta\Xi^{\mathsf{T}}\mathbf{x})\right]_\nu \boldsymbol{\xi}_\nu\right\| &\text{(E.37)}\\
&\leq \widetilde{\epsilon}\|\boldsymbol{\xi}_\mu\| + \frac{\widetilde{\epsilon}}{M-1}\sum_{\nu=1,\nu\neq\mu}^{M}\|\boldsymbol{\xi}_\nu\| &\text{(E.38)}\\
&\leq \widetilde{\epsilon}\left(m + \frac{1}{M-1}\sum_{\nu=1,\nu\neq\mu}^{M} m\right) &\text{(E.39)}\\
&\leq 2\widetilde{\epsilon}m, &\text{(E.40)}
\end{aligned}
$$

where $\widetilde{\epsilon} := (M-1)\exp\left\{-\beta\widetilde{\Delta}_\mu\right\} = (M-1)\exp\left\{-\beta\left(\langle\boldsymbol{\xi}_\mu,\mathbf{x}\rangle - \mathrm{Max}_{\nu\in[M]}\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\nu\rangle\right)\right\}$ and the inequality

$$
\left[\mathrm{Softmax}(\beta\Xi^{\mathsf{T}}\mathbf{x})\right]_\nu = \frac{\exp\{\beta\left(\langle\mathbf{x},\boldsymbol{\xi}_\nu\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\mu\rangle\right)\}}{1+\sum_{\nu'\neq\mu}\exp\{\beta\left(\langle\mathbf{x},\boldsymbol{\xi}_{\nu'}\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\mu\rangle\right)\}} \leq \exp\left\{-\beta\widetilde{\Delta}_\mu\right\}, \tag{E.41}
$$

is used in the fourth line.

Then, by the Cauchy-Schwartz inequality

$$
\left|\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\mu\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\mu\rangle\right| \leq \|\boldsymbol{\xi}_\mu - \mathbf{x}\|\cdot\|\boldsymbol{\xi}_\mu\| \leq \|\boldsymbol{\xi}_\mu - \mathbf{x}\|m, \quad \forall\mu\in[M], \tag{E.42}
$$

we observe that $\widetilde{\Delta}_\mu$ can be expressed in terms of Δ_μ:

$$
\begin{aligned}
\widetilde{\Delta}_\mu &= \mathop{\mathrm{Min}}_{\nu,\nu\neq\mu}\left[\langle\mathbf{x},\boldsymbol{\xi}_\mu\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\nu\rangle + \left(\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\mu\rangle - \langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\nu\rangle\right) - \left(\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\mu\rangle - \langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\nu\rangle\right)\right] \qquad &\text{(E.43)}\\
&\geq \mathop{\mathrm{Min}}_{\nu,\nu\neq\mu}\left[\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\mu\rangle - \langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\nu\rangle + \left(\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\nu\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\nu\rangle\right) - \left(\langle\boldsymbol{\xi}_\mu,\boldsymbol{\xi}_\mu\rangle - \langle\mathbf{x},\boldsymbol{\xi}_\mu\rangle\right)\right] &\\
& &\text{(By Cauchy-Schwarz)}\\
&= \Delta_\mu - 2\|\boldsymbol{\xi}_\mu - \mathbf{x}\|m = \Delta_\mu - 2mR, &\text{(By } \mathbf{x}\in S_\mu\text{)}
\end{aligned}
$$

where R is radius of the sphere S_μ.

Hence, combining the bound from (E.40) with (E.34) results in

$$
\begin{aligned}
\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| &\leq \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq 2\widetilde{\epsilon}m \qquad &\text{(E.44)}\\
&= 2(M-1)\exp\left\{-\beta\widetilde{\Delta}_\mu\right\}m &\text{(E.45)}\\
&\leq 2(M-1)\exp\{-\beta\left(\Delta_\mu - 2mR\right)\}m. &\text{(E.46)}
\end{aligned}
$$

Therefore, given $\delta := \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| - \|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq 0$, we have

$$
\|\mathcal{T}(\mathbf{x}) - \boldsymbol{\xi}_\mu\| \leq 2(M-1)\exp\{-\beta\left(\Delta_\mu - 2mR + \delta\right)\}m - \delta \leq \|\mathcal{T}_{\text{Dense}}(\mathbf{x}) - \boldsymbol{\xi}_\mu\|. \tag{E.47}
$$

For $\mathcal{T}$ to be a mapping from S_μ onto S_μ, it is sufficient to have

$$
2(M-1)\exp\{-\beta(\Delta_\mu - 2mR)\}m - \delta \leq R, \tag{E.48}
$$

which leads to

$$
\Delta_\mu \geq \frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR. \tag{E.49}
$$

□

E.6 Lemma 3.1

We begin with a helper lemma.

Lemma E.3 ([Ramsauer et al., 2021])**.** Given real numbers $a, b \in \mathbb{R}$. If the equation

$$ac + c \ln c - b = 0, \tag{E.50}$$

holds, then the solution is

$$c = \frac{b}{W_0(\exp(a + \ln b))}. \tag{E.51}$$

Proof. Starting from the given equation, we can rearrange and solve for c as follows:

$$\begin{aligned}
ac + c \ln c - b &= 0, \\
a + \ln c &= \frac{b}{c}, \\
\frac{b}{c} + \ln\left(\frac{b}{c}\right) &= a + \ln b, \\
\frac{b}{c} \exp\left(\frac{b}{c}\right) &= \exp(a + \ln b), \\
\frac{b}{c} &= W_0(\exp(a + \ln b)), \\
c &= \frac{b}{W_0(\exp(a + \ln b))}.
\end{aligned}$$

This completes the proof. □

Then we present the proof.

Proof of Lemma 3.1. Equipped with $\Delta_\mu \geq \frac{1}{\beta} \ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR$ from Corollary 3.1.1, we first write down the probability of success storage and retrieval, i.e. minimal separation $\Delta_{\min}$ satisfies well-separation condition.

Let $\Delta_{\min} = \text{Min}_{\mu \in [M]} \Delta_\mu$, and $\theta_{\mu\nu}$ be the angle between two patterns $\boldsymbol{\xi}^\mu$ and $\boldsymbol{\xi}^\nu$. Intuitively, $\theta_{\mu\nu} \in [0, \pi]$ represent the pairwise correlation of two patterns the two patterns.

We have

$$\Delta_{\min} \geq \frac{1}{\beta} \ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR, \tag{E.52}$$

and

$$\Delta_{\min} = \underset{1 \leq \mu \leq \nu \leq M}{\text{Min}} \left[m^2 \left(1 - \cos(\theta_{\mu\nu})\right)\right] = m^2 \left[1 - \cos(\theta_{\min})\right], \tag{E.53}$$

where $\theta_{\min} := \text{Min}_{1 \leq \mu \leq \nu \leq M} \theta_{\mu\nu} \in [0, \pi]$. Then, it holds

$$m^2 \left[1 - \cos(\theta_{\min})\right] \geq \frac{1}{\beta} \ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR. \tag{E.54}$$

With Corollary 3.1.1 , we write down the probability of success storage and retrieval as

$$P\left(\Delta_\mu \geq \frac{1}{\beta} \ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right) = 1 - p. \tag{E.55}$$

By (E.54), we have

$$P\left(m^2 \left[1 - \cos(\theta_{\min})\right] \geq \frac{1}{\beta} \ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right) = 1 - p. \tag{E.56}$$

By [Olver et al., 2010, (4.22.2)], for $0 \leq \cos(\theta_{\min}) \leq 1$, $\cos(\theta_{\min})$ can be upper bounded by:

$$\cos(\theta_{\min}) \leq 1 - \frac{\theta_{\min}^2}{5}. \tag{E.57}$$

It holds

$$P\left(\frac{m^2\theta_{\min}^2}{5} \geq \frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right) = 1-p, \tag{E.58}$$

which can be rewritten as

$$P\left(M^{\frac{2}{d-1}}\theta_{\min} \geq \frac{\sqrt{5}M^{\frac{2}{d-1}}}{m}\left[\frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right]^{\frac{1}{2}}\right) = 1-p. \tag{E.59}$$

Here, $M^{2/d-1}$ is introduced for later convenience.

Let

$$\omega_d := \frac{2\pi^{d+1/2}}{\Gamma\left(\frac{d+1}{2}\right)}, \tag{E.60}$$

be the surface area of a d-dimensional unit sphere is, where $\Gamma(\cdot)$ represents the gamma function.

By [Brauchart et al., 2018, Lemma 3.5], we obtain

$$\begin{aligned} P\left(M^{\frac{2}{d-1}}\theta_{\min} \geq \frac{\sqrt{5}M^{\frac{2}{d-1}}}{m}\left[\frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right]^{\frac{1}{2}}\right) &= 1-p \\ &\geq 1 - \frac{1}{2}\gamma_{d-1}5^{\frac{d-1}{2}}M^2m^{-(d-1)}\left[\frac{1}{\beta}\ln\left(\frac{2(M-1)m}{R+\delta}\right) + 2mR\right]^{\frac{d-1}{2}}, \end{aligned} \tag{E.61}$$

where γ_d is defined as the ratio of surface areas of $(d-1)$- and d-dimensional unit sphere:

$$\gamma_d := \frac{1}{d}\frac{\omega_{d-1}}{\omega_d} = \frac{1}{d\sqrt{\pi}}\frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}. \tag{E.62}$$

Recall $d, M \in \mathbb{N}_+$, $p \in [0,1]$. With some real value $C \in \mathbb{R}$, it holds

$$M = \sqrt{p}C^{\frac{d-1}{4}}. \tag{E.63}$$

From (E.61), we have

$$5^{\frac{d-1}{2}}\left(\sqrt{p}C^{\frac{d-1}{4}}\right)^2 m^{-(d-1)}\left\{\frac{1}{\beta}\ln\left[\frac{2\left(\sqrt{p}C^{\frac{d-1}{4}}-1\right)m}{R+\delta}\right] + \frac{1}{\beta}\right\}^{\frac{d-1}{2}} - p \leq 0, \tag{E.64}$$

which leads to

$$5^{\frac{d-1}{2}}C^{\frac{d-1}{2}}m^{-(d-1)}\left\{\frac{1}{\beta}\ln\left[\frac{2\left(\sqrt{p}C^{\frac{d-1}{4}}-1\right)m}{R+\delta}\right] + \frac{1}{\beta}\right\}^{\frac{d-1}{2}} \leq 1. \tag{E.65}$$

To apply Lemma E.3, we first rearrange (E.65) as

$$\frac{5C}{m^2\beta}\left\{\ln\left[\frac{2\left(\sqrt{p}C^{\frac{d-1}{4}}-1\right)m}{R+\delta}\right] + 1\right\} - 1 \leq 0, \tag{E.66}$$

and then identify

$$a := \frac{4}{d-1}\left\{\ln\left[\frac{2m(\sqrt{p}-1)}{R+\delta}\right] + 1\right\}, \quad b := \frac{4m^2\beta}{5(d-1)}. \tag{E.67}$$

By Lemma E.3, we have the solution

$$C = \frac{b}{W_0(\exp\{a + \ln b\})}, \tag{E.68}$$

where $W_0(\cdot)$ is the upper branch of the Lambert W function. Since the domain of the Lambert W function is $x > -{}^{1}\!/_{e}$ and the fact $\exp\{a + \ln b\} > 0$, the solution exists.

When C satisfies inequality (E.65), we arrive a lower bound on the exponential storage capacity M:

$$M \geq \sqrt{p} C^{\frac{d-1}{4}}. \tag{E.69}$$

Notably, the above takes similar form as [Ramsauer et al., 2021, Theorem 3]. To see the blessings of sparsity, we consider the following asymptotic analysis and compare with results from the dense modern Hopfield model. To compare with results from dense modern Hopfield model, we denote the dense counterparts of a, b with $\widetilde{\cdot}$ notation, i.e.

$$\widetilde{a} \coloneqq \frac{2}{d-1}\left[1 + \ln\left(2\beta m^2 p\right)\right], \quad \widetilde{b} = b. \tag{E.70}$$

By [Corless et al., 1996], for sufficient large z, $W_0(z)$ is asymptotic to

$$W_0(z) \simeq \ln z - \ln \ln z + \mathcal{O}(1). \tag{E.71}$$

Therefore, for sufficient large β, we have

$$W_0\left(\exp\{a + \ln b\}\right) \simeq a + \ln b - \ln\left(a + \ln b\right) + \mathcal{O}(1), \tag{E.72}$$

which is dominated by a.

For a, we have

$$\widetilde{a} \leq a, \tag{E.73}$$

and hence

$$W_0\left(\exp\left\{\widetilde{a} + \ln \widetilde{b}\right\}\right) \leq W_0\left(\exp\{a + \ln b\}\right). \tag{E.74}$$

Therefore, combining above with (E.68), we have

$$\widetilde{C} = \frac{b}{W_0\left(\exp\left\{\widetilde{a} + \ln \widetilde{b}\right\}\right)} \leq \frac{b}{W_0\left(\exp\{a + \ln b\}\right)} = C, \tag{E.75}$$

which states that the lower bound of the sparse capacity is larger than that of [Ramsauer et al., 2021]

$$M = \sqrt{p} C^{\frac{d-1}{4}} \geq \sqrt{p} \widetilde{C}^{\frac{d-1}{4}} = M_{\text{Dense}}. \tag{E.76}$$

□

F Auxiliary Theoretical Background

Remark F.1 (Remark on Definition 2.1)**.** To see the equivalence of the two optimization problems, we observe

$$\begin{aligned}
\underset{\mathbf{p}\in\Delta^d}{\operatorname{ArgMax}}\left[\langle \mathbf{p}, \mathbf{z}\rangle - \Psi(\mathbf{p})\right] &= \underset{\mathbf{p}\in\Delta^d}{\operatorname{ArgMax}}\left[\langle \mathbf{p}, \mathbf{z}\rangle - \frac{1}{2}\|\mathbf{p}\|^2\right] \\
&= \underset{\mathbf{p}\in\Delta^d}{\operatorname{ArgMin}}\left[-\frac{1}{2}\left(\|\mathbf{p}\|^2 + \|\mathbf{z}\|^2 - 2\left\langle \mathbf{p}, \mathbf{z}\right\rangle\right)\right] \\
&= \underset{\mathbf{p}\in\Delta^d}{\operatorname{ArgMin}}\left[\frac{1}{2}\|\mathbf{p} - \mathbf{z}\|^2\right],
\end{aligned} \tag{F.1}$$

where the last line is obtained by inserting $\|\mathbf{z}\|^2$ as a constant in (F.1).

G Additional Experiments

In order to highlight the benefits of the sparse Hopfield model, particularly under conditions of high data sparsity, we broaden our experimental studies with more models. These models include the `SparseHopfield`, `Hopfield`, the attention mechanism [Vaswani et al., 2017], and a attention-based MIL baseline, the gated-attention mechanism [Ilse et al., 2018].

G.1 Visualization of Experimental Validation of Theoretical Results

We provide visual demonstrations of Section 4.1 in Figure 4.

G.2 Bit Pattern MIL

To supplement Section 4.2.1, we conduct further numerical investigations on the same MIL tasks (**bag sparsity** and **bag size**) with `SparseHopfield`, `Hopfield`. In these experiments, we contrast the performance of `SparseHopfield` and `Hopfield` (and also `SparseHopfieldPooling` and `HopfieldPooling`) with the attention mechanism [Vaswani et al., 2017] and the gated-attention mechanism [Ilse et al., 2018]. For the **bag size**, we fix the number of positive pattern in a bag to be 1, and vary bag size from 20 to 300. For the **bag sparsity**, we fix the bag size as 200, and inject from 2 to 100 positive patterns in a positive bag, results in 1 to 50 percent of positive patterns in each positive bag. The results are reported in Table 4. For numerical experiments on synthetic datasets, we do not use hyperparameter search due to the simplicity of both model structure and data.

Table 4: **Top (Bag Size):** Accuracy comparison on bit pattern dataset for sparse and dense Hopfield model. We report the average accuracy over 10 runs. The results suggest that the sparse Hopfield model demonstrates a better performance when facing a bag size increase. **Bottom (Bag Sparsity):** Performance comparison on bit pattern dataset for sparse and dense Hopfield model with varying bag sparsity. We report the average accuracy over 10 runs. The results suggest that the sparse Hopfield model demonstrates a better performance across all sparsity.

Bag Size	20	50	100	150	200	300
Sparse Hopfield	98.82 ± 0.34	99.45 ± 0.19	97.13 ± 0.11	95.98 ± 0.12	94.17 ± 0.01	90.15 ± 0.30
Dense Hopfield	99.65 ± 0.70	99.51 ± 0.87	53.90 ± 0.00	49.51 ± 0.02	51.92 ± 0.12	53.83 ± 0.12
Sparse Hopfield Pooling	**99.71 ± 0.06**	100.0 ± 0.00	100.0 ± 0.00	**99.76 ± 0.00**	**99.76 ± 0.00**	**99.76 ± 0.00**
Dense Hopfield Pooling	99.68 ± 0.15	100.0 ± 0.00	100.0 ± 0.00	76.44 ± 0.23	49.13 ± 0.01	52.88 ± 0.01
Attention	87.01 ± 0.00	74.51 ± 0.01	45.19 ± 0.31	53.75 ± 0.76	46.63 ± 0.02	53.36 ± 0.03
Gated	87.88 ± 0.00	63.44 ± 0.04	75.38 ± 0.56	73.45 ± 0.70	71.05 ± 0.35	49.61 ± 1.78

Bag Sparsity	1%	2%	3%	5%	10%	20%	40%	50%
Sparse Hopfield	95.62 ± 0.01	95.98 ± 0.30	99.68 ± 0.01	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00
Dense Hopfield	51.44 ± 0.01	57.21 ± 0.01	75.48 ± 0.01	99.03 ± 0.11	99.51 ± 0.02	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00
Sparse Hopfield Pooling	**99.76 ± 0.00**	**99.68 ± 0.00**	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00
Dense Hopfield Pooling	49.20 ± 0.00	85.58 ± 0.10	100.0 ± 0.00	100.0 ± 0.00	99.68 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00
Attention	74.51 ± 0.01	78.81 ± 0.04	96.63 ± 0.02	100.0 ± 0.00	99.51 ± 0.01	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00
Gated	78.94 ± 0.41	95.28 ± 0.35	98.55 ± 0.00	99.03 ± 0.01	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00

G.3 Convergence Analysis

To supplement Section 4.2.1, we also analyze the convergence behavior of the `SparseHopfield` and `Hopfield` numerically. In Figure 5, we plot the loss and accuracy curve for both models on the bit pattern dataset for the **bag size** tasks mentioned in Section 4.2.1. We include various bag sizes in the plot to examine how the loss curve responds to an increase in bag size (i.e., the number of memory patterns, M). The results show that, `SparseHopfield` surpasses the `Hopfield` in nearly all bag sizes. Moreover, for the same bag size, `SparseHopfield` always reaches the minimum validation loss faster than `Hopfield`. This provides empirical support for our theoretical prediction outlined in Theorem 2.2. In conjunction with the findings illustrated in Figure 2, Figure 5 reinforces the benefits of utilizing the sparse Hopfield model. In particular, the evidence verifies the claim in Theorem 2.2, demonstrating that the convergence speed of the sparse and dense Hopfield models shows different dependencies on the bag size M in this experiment.

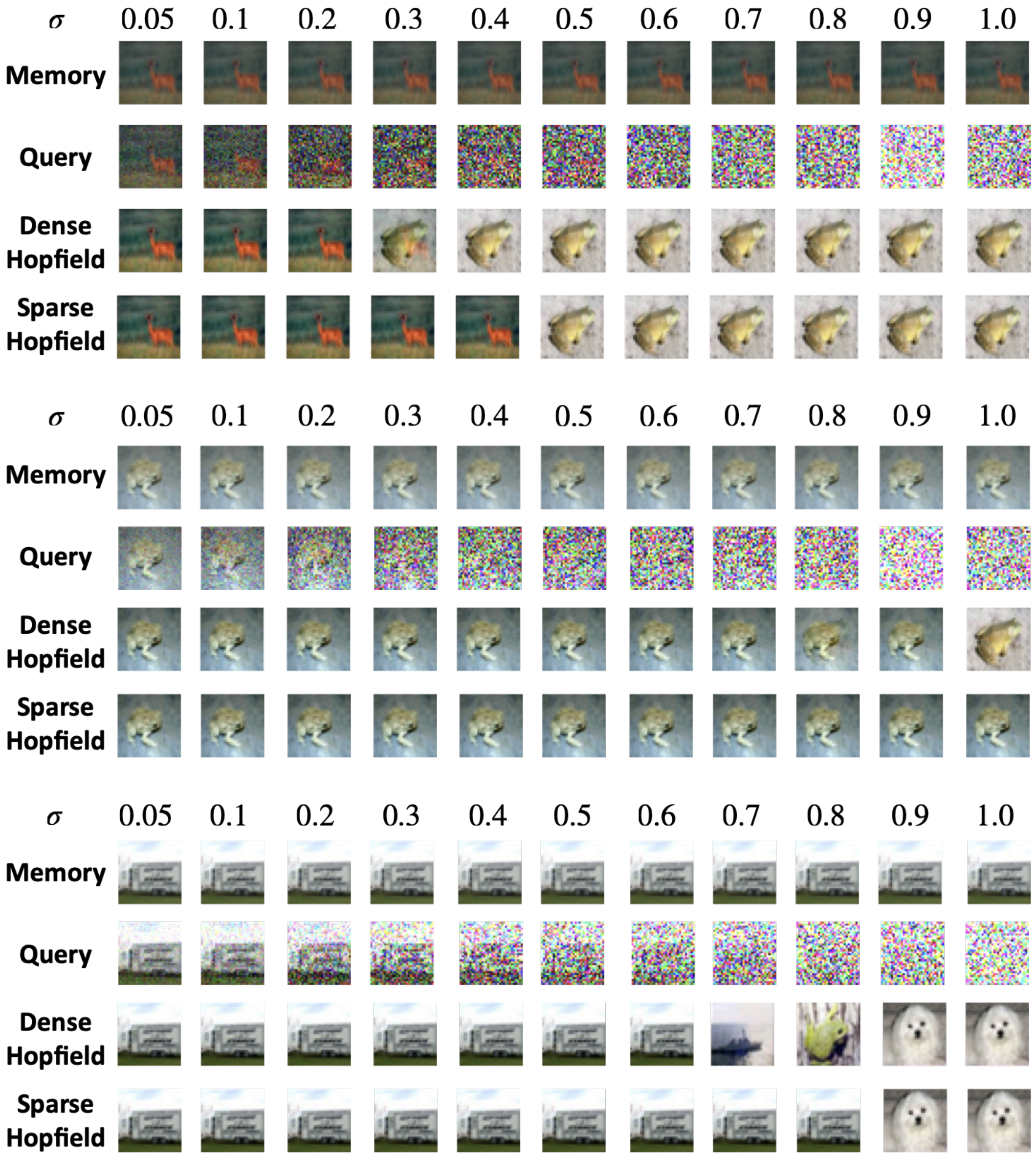

Figure 4: **Visualizing noise-robustness of sparse and dense Hopfield models (Figure 1 of Section 4.1).** We perform memory retrieval using both Dense and Sparse Hopfield models, with queries subjected to varying levels of noise. We randomly select an image from the CIFAR10 dataset to serve as the memory pattern. This selected image is then contaminated with different levels of random noise ($\mu = 0$ and $0.05 \leq \sigma \leq 1.5$) to generate query patterns. The results demonstrate that the Sparse Hopfield model is more effective in retrieving the original image, showcasing its superior robustness against noise.

G.4 Sparsity Generalization

To supplement Section 4.2.1, we explore another scenario where the bag sparsity shifts between training and test data. We train dense and sparse Hopfield models on a certain bag sparsity, and evaluate on another. The main goal of this setting is to investigate the generalization performance of dense and sparse Hopfield models when the information sparsity shift of training and test data distribution. We fix the bag size to 200, and then implant different number of positive signals to the training and test dataset range from 0.5 to 50 percent. We report the results of `HopfieldPooling`

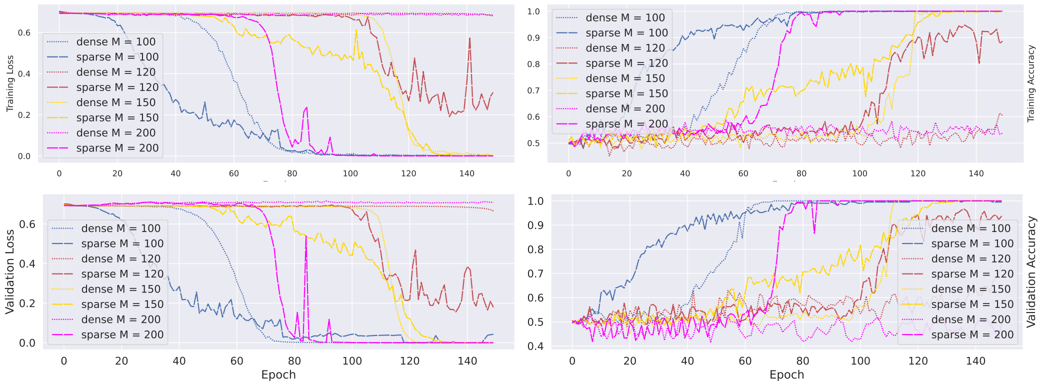

Figure 5: **Top:** The training loss and accuracy curve of `SparseHopfield` and `Hopfield` with different bag sizes. **Bottom:** The validation loss and accuracy curve of `SparseHopfield` and `Hopfield` with different bag sizes. The plotted are the mean of 10 runs. The results indicate that the sparse Hopfield model converges faster than the dense model and also yields superior validation/test accuracy.

and `SparseHopfieldPooling` in Table 5.[12] The result shows that for `HopfieldPooling`, while train on dense bags help its performance of evaluating on sparse bags, lacking ability of learning from sparse bags still affects its performance. Meanwhile `SparseHopfieldPooling` is more robust against sparsity shift, especially for the case where it was trained on dense bags and evaluate on sparse bags. However, both sparse and dense Hopfield models inevitably suffer from a performance drop when having a sparsity gap when train bags are much more sparse than test bags.

Table 5: **Accuracy comparison on bit pattern dataset for sparse and dense Hopfield Model when varying the train/test sparsity gap.** We report the average accuracy over 10 runs. The result shows that for `HopfieldPooling`, while train on dense bags help its performance of evaluating on sparse bags, lacking ability of learning from sparse bags still affects its performance. Meanwhile `SparseHopfieldPooling` is more robust against sparsity shift, especially for the case where it was trained on dense bags and evaluate on sparse bags. However, both sparse and dense Hopfield models inevitably suffer from a performance drop when having a sparsity gap when train bags are much more sparse than test bags.

	# of Train Positive Signal per Bag (Dense/Sparse)						
# of Test	1	2	10	20	40	80	100
1	46.63 / 99.76	48.55 / 94.31	53.84 / 74.52	59.61 / 81.73	66.82 / 81.25	72.07 / 81.73	72.59 / 81.25
2	47.59 / 52.40	51.44 / 98.18	58.17 / 95.19	62.01 / 95.67	69.23 / 95.67	72.59 / 95.67	72.11 / 95.67
10	99.51 / 100.0	99.51 / 100.0	99.51 / 100.0	99.51 / 100.0	99.51 / 100.0	99.51 / 100.0	99.51 / 100.0
20	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0
40	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0
80	100.0 / 97.03	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0
100	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0	100.0 / 100.0

G.5 Real-World Experiments

To examine the practical applicability of the proposed model, we implement it in two additional experiments that utilize transformer-based models for distinct tasks. These tasks include multivariate time series prediction [Zhang and Yan, 2022], and neural machine translation [Vaswani et al., 2017]. In these experiments, we substitute the existing attention mechanism with both the `Hopfield` and `SparseHopfield` layers.

[12] For the ease of presentation, we exclude the standard deviations in Table 5 as they are all close to zero and less than 0.31%

G.5.1 Multivariate Time Series Prediction

For the multivariate time series prediction task, we implement two variants of the SOTA Crossformer model [Zhang and Yan, 2022], **Crossformer-DH** and **Crossformer-SH**, with `Hopfield` and `SparseHopfield` layers respectively. These models employ an architecture akin to the Swin-Transformer [Liu et al., 2021], utilizing shifting windows to extract information at multiple resolutions. The experiment results are showed in Table 6. Our results indicate that our proposed `SparseHopfield` not only consistently enhances transformer-based deep learning models but also achieves SOTA performance. In 60+% of 58 settings, the Sparse Hopfield model, Crossformer-SH, ranks first or second, with 28 top and 7 runner-up performances.

Datasets. We conduct the experiments on four multivariate time series real-world datasets: ETTh1 (Electricity Transformer Temperature-hourly), ETTm1 (Electricity Transformer Temperature-minutely), WTH (Weather), ILI (Influenza-Like Illness), ECL (Electricity Consuming Load), Traffic.

Baselines. We benchmark our method against the results of [Zhang and Yan, 2022] and other baselines (Tranformer [Vaswani et al., 2017], Informer [Zhou et al., 2021] and Autoformer [Chen et al., 2021]) therein.

Setup. We adopt the same setting as in [Zhang and Yan, 2022]: multivariate time series prediction task on various datasets. Following [Zhang and Yan, 2022], for each dataset, we evaluate our models with several different prediction horizons. As for hyperparameters, we simply adopt the optimized hyperparameter configuration used in [Zhang and Yan, 2022] obtained via grid search for both [Zhang and Yan, 2022] and all baselines. We report the average accuracy of 5 runs, evaluated using Mean Square Error (MSE) and Mean Absolute Error (MAE) metrics.

G.5.2 Neural Machine Translation

We showcase the application of the proposed sparse Hopfield model in the context of the classic neural machine translation task, as described in [Vaswani et al., 2017]. By substituting the attention mechanism in the transformer with a 1-step `Hopfield` and `SparseHopfield`, we compare the performance (BLEU score) of the transformer and Hopfield models on various language pairs. The results of this comparison can be found in Appendix G.4.

Datasets. We use the WMT17 [Bojar et al., 2017] machine translation task dataset. Which consists of sentence pairs of two different languages, where we consider the translation between German and English (EN-DE), Russian and English (RU-EN). The EN-DE setting has 5.91M pairs of training data, 3000 pairs of validation and 3000 pairs of test data. The EN-RU setting has 25.78M pairs of training data, 3000 pairs of validation and 3000 pairs of test data.

Baselines. For the baselines, we follow the architecture of base transformer in [Vaswani et al., 2017] which has 6 layers of encoder and decoder. The hidden dimension is 512 and the feed forward dimension is 2048. More details of configuration can be found in Table 8. Note that when switching the attention in base transformer to either `Hopfield` or `SparseHopfield`, no extra parameter was added in our experiment. Thus, the comparison in our setting is fair.

Setup. We follow the setup in [Vaswani et al., 2017] on the WMT-17 dataset. In this experiment, we consider the task of English to German (EN-DE), German to English (DE-EN), Russian to English (RU-EN) and English to Russian (EN-RU). We report the BLEU score on the test set. For WMT17, we follow the base-transformer in [Vaswani et al., 2017], and train the model with 50000 steps, and report the performance of the last checkpoint[13]. Our results show that our proposed `SparseHopfield` not only consistently improves upon transformer-based deep learning models but also surpasses the performance of the dense Hopfield model [Ramsauer et al., 2021].

[13] We follow the implementation in `https://github.com/OpenNMT/OpenNMT-py` for the NMT experiment.

Table 6: **Accuracy comparison for multivariate time series predictions on various datasets, using both the sparse and dense Hopfield models.** Based on SOTA prediction model Crossformer [Zhang and Yan, 2022], we implement two Crossformer variants, **Crossformer-DH** and **Crossformer-SH**, with `Hopfield` and `SparseHopfield` layers respectively. We report the average accuracy of 5 runs, evaluated using Mean Square Error (MSE) and Mean Absolute Error (MAE) metrics. We benchmark our method against the results of [Zhang and Yan, 2022] and other baselines (Tranformer [Vaswani et al., 2017], Informer [Zhou et al., 2021] and Autoformer [Chen et al., 2021]) therein. We evaluate each dataset with different prediction horizons (showed in the second column). We have the best results **bolded** and the second best results underlined. In 60+% of 58 settings, the Sparse Hopfield model, Crossformer-SH, ranks first or second, with 28 top and 7 runner-up performances. Our results indicate that our proposed `SparseHopfield` not only consistently enhances transformer-based deep learning models but also achieves SOTA or comparable performance.

Models		Transformer		Informer		Autoformer		Crossformer		**Crossformer-DH**		**Crossformer-SH**	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	24	0.620	0.577	0.577	0.549	0.439	0.440	0.305	0.367	0.299	0.365	**0.295**	**0.357**
	48	0.692	0.671	0.685	0.625	0.429	0.442	0.352	0.394	0.351	0.399	**0.346**	**0.392**
	168	0.947	0.797	0.931	0.752	0.493	0.479	**0.410**	**0.441**	0.412	0.443	0.425	0.455
	336	1.094	0.813	1.128	0.873	0.509	0.492	**0.440**	**0.461**	0.455	0.468	0.459	0.477
	720	1.241	0.917	1.215	0.896	0.539	0.537	0.519	0.524	0.523	0.529	**0.518**	**0.522**
ETTm1	24	0.306	0.371	0.323	0.369	0.410	0.428	0.310	0.371	0.199	0.285	**0.198**	**0.287**
	48	0.465	0.470	0.494	0.503	0.483	0.464	0.300	0.352	0.290	0.356	**0.276**	**0.340**
	96	0.681	0.612	0.678	0.614	0.502	0.476	0.320	0.373	0.344	0.398	**0.305**	**0.371**
	288	1.162	0.879	1.056	0.786	0.604	0.522	0.404	0.427	0.404	0.429	**0.373**	**0.406**
	672	1.231	1.103	1.192	0.926	0.607	0.530	0.569	0.528	0.568	0.523	**0.467**	**0.474**
WTH	24	0.349	0.397	0.335	0.381	0.363	0.396	**0.294**	**0.343**	**0.294**	**0.343**	**0.294**	0.344
	48	0.386	0.433	0.395	0.459	0.456	0.462	0.370	0.411	**0.369**	**0.408**	0.375	0.410
	168	0.613	0.582	0.608	0.567	0.574	0.548	0.473	0.494	**0.472**	**0.493**	0.480	0.499
	336	0.707	0.634	0.702	0.620	0.600	0.571	**0.495**	**0.515**	0.498	0.519	0.504	0.523
	720	0.834	0.741	0.831	0.731	0.587	0.570	**0.526**	**0.542**	0.528	0.546	0.536	0.544
ILI	24	3.954	1.323	4.588	1.462	3.101	1.238	**3.041**	1.186	3.428	1.279	3.124	**1.143**
	36	4.167	1.360	4.845	1.496	**3.397**	1.270	3.406	1.232	3.490	1.306	3.404	**1.192**
	48	4.746	1.463	4.865	1.516	**2.947**	**1.203**	3.459	1.221	3.600	1.277	3.509	1.205
	60	5.219	1.553	5.212	1.576	**3.019**	**1.202**	3.640	1.305	3.666	1.271	3.709	1.205
ECL	48	0.334	0.399	0.344	0.393	0.241	0.351	0.156	0.255	0.159	0.264	**0.154**	**0.254**
	168	0.353	0.420	0.368	0.424	0.299	0.387	0.231	0.309	0.290	0.316	**0.225**	**0.303**
	336	0.381	0.439	0.381	0.431	0.375	0.428	0.323	0.369	**0.318**	**0.363**	0.332	0.375
	720	0.391	0.438	0.406	0.443	**0.377**	0.434	0.404	0.423	0.397	**0.421**	0.414	0.429
	960	0.492	0.550	0.460	0.548	**0.366**	**0.426**	0.433	0.438	0.434	0.438	0.440	0.443
Traffic	24	0.597	0.332	0.608	0.334	0.550	0.363	0.491	0.274	**0.488**	**0.271**	0.496	0.280
	48	0.658	0.369	0.644	0.359	0.595	0.376	0.519	0.295	**0.513**	0.291	0.516	**0.290**
	168	0.664	0.363	0.660	0.391	0.649	0.407	0.513	0.289	0.516	0.289	**0.512**	**0.288**
	336	0.654	0.358	0.747	0.405	0.624	0.388	0.530	0.300	0.541	0.304	**0.529**	**0.297**
	720	0.685	0.370	0.792	0.430	0.674	0.417	0.573	0.313	0.557	0.307	**0.555**	**0.304**

H Experimental Details

All experiments are conducted on the platform with NVIDIA GEFORCE RTX 2080 Ti and INTEL XEON SILVER 4214 @ 2.20GHz.

H.1 Multiple Instance Learning (MIL)

H.1.1 Synthetic Dataset

Architectural Details. For the Hopfield model, the architecture is composed of 1 layer of either `Hopfield` or `HopfieldPooling` and 1 layer of fully connected output projection. For the attention model, the architecture is composed of 1 layer of attention layer and 1 layer of fully connected output projection. The dataset contains 50% of positive bags and 50% of negative bags.

Training Details. We use an AdamW [Loshchilov and Hutter, 2017] optimizer. For each bag size, we ran the experiment 10 times with different random seed. For all of our synthetic dataset

Table 7: **Results for the machine translation on the WMT17 dataset with language pairs of DE-EN, EN-DE, RU-EN, EN-RU.** We showcase the application of the proposed sparse Hopfield model in the context of the classic neural machine translation task on WMT17 dataset, as outlined in [Vaswani et al., 2017]. By substituting the attention mechanism in the transformer with a 1-step `Hopfield` and `SparseHopfield`, we compare the performance (BLEU score) of the transformer and Hopfield models. To ensure a fair comparison, all models (Transformer, Dense Hopfield, Sparse Hopfield) are of the same size. Our results show that our proposed `SparseHopfield` consistently improves upon transformer-based deep learning models

Dataset	**DE-EN**	**EN-DE**	**RU-EN**	**EN-RU**
Transformer	29.8	34.9	28.5	24.8
Dense Hopfield	33.6	**37.2**	28.5	**24.9**
Sparse Hopfield	**36.1**	37.1	**28.6**	24.8

Table 8: Hyperparameter of the NMT experiment.

parameter	**values**
batch size	4096
initial lr	2.0
vocab size (DE-EN)	36000
vocab size (EN-RU)	34776
num heads	8
hidden dimension	512
word vector dimension	512
feed forward dimension	2048
encoder layer	6
decoder layer	6
label smoothing	0.1
decay method	Noam
optimizer	Adam
warm up steps (DE-EN)	4000
warm up steps (EN-RU)	8000
train steps (DE-EN)	50000
train steps (EN-RU)	80000
max sequence length	96
beam size	5
tokenizer	sentencepiece

experiments, we use the exact same configuration, shown in Table 5. The coefficients of Adam optimizer, betas, are set to $(0.9, 0.999)$. As the number of training epochs, we use 150, and the evaluate the model on the testset with the last checkpoint. All the experiments done on the synthetic datasets follow the same architecture and training details.

Baselines. For our synthetic dataset, we consider two baselines. [Ilse et al., 2018] (**Gated**), where they proposed a gated-attention mechanism by inserting one extra linear layer on the attention weights before the softmax function. And they replaced the activation function of query to Tanh function. [Vaswani et al., 2017] (**Attn**), where they proposed a multi-head attention mechanism has been widely used in modern deep learning.

Table 9: Statistics of Bit Pattern Synthetic Dataset.

Unique Patterns	Pattern Length (bits)	Training	Test
4	4	800	200

Table 10: Hyperparameter of the Bit Pattern Dataset.

parameter	**values**
batch size	128
learning rates	10^{-3}
scaling factors	0.25
num heads	8
head dimension	8
max update steps	3
dropout	0.5

H.1.2 MIL Benchmark Datasets

The experiment is conducted on 4 popular MIL datasets. Elephant, Fox and Tiger are datasets for image annotation which are composed of preprocessed and segmented colored images. Each image is characterized by color, texture and shape descriptors. These datasets contain 100 positive images that contain the purposed animals and 100 negative images that are drawn from a pool of images of other animals. Furthermore, we tested our model on the UCSB breast cancer classification task. An instance in UCSB dataset represents a patch of a histopathological image of cancerous or normal tissue. The detailed statistics of datasets are summarized in Table 11.

Table 11: Statistics of MIL benchmark datasets

Name	Instances	Features	Bags	+bags	−bags
Elephant	1391	230	200	100	100
Fox	1302	230	200	100	100
Tiger	1220	230	200	100	100
UCSB Breast Cancer	2002	708	58	26	32

In detail, we used a similar architecture described in [Ramsauer et al., 2021] to perform the MIL tasks. Firstly, the instance embeddings are sent to fully connected linear embedding layers with ReLU activation. After that, we used a `SparseHopfield` which has Sparse Retrieval Dynamics to process the output of fully connected linear layers. Afterward, we flatten the output of `SparseHopfield` and use a linear network with ReLU activation can perform classification.

To avoid application bias, we follow the experiment setting in [Küçükaşcı and Baydoğan, 2018, Ramsauer et al., 2021] and utilize a stratified ten-fold cross-validation to demonstrate the success of proposed `SparseHopfield` and `Hopfield`. For each fold in cross-validation, we use a stratified sampling process to split folds for training into a training set and validation set with a 0.1 split rate. We train the models' parameters and tune hyperparameters via grid searching. Once the hyperparameters are selected and the parameters of models are tuned on the first train folds of the first seed, we apply the selected configuration of the model models on other test folds or folds of other random seeds. All reported ROC-AUC scores are the average results of 5 runs with different random seeds.

The grid search space is listed in Table 12. The embedding layers are the pre-HopfieldPooling linear network and the layer width of them is the number of hidden units. A dropout operation, also known as bag dropout, is applied to the attention matrix since it's easy to overfit on these benchmark datasets. All models are trained with the Adam optimizer for 50 epochs. To combat overfitting, we also use an early-stopper that chooses the best checkpoint on the validation set.

One thing that should be noticed is that [Ramsauer et al., 2021] uses the pooling layers `HopfieldPooling` for MIL tasks instead of associative layers `SparseHopfield` or `Hopfield`. We also conduct an ablation experiment in that the model uses the first two modules for MIL tasks following the model structure as well as its training and testing process presented above. As shown in Table 13, the pooling layers can reach comparative results with associative layers on Fox and Tiger datasets but have performance degradation on Elephant and UCSB datasets. Besides, the `SparseHopfieldPooling` can also perform better than `HopfieldPooling` on Tiger and Elephant datasets.

Table 12: Hyperparameter grid search space on the respective validation sets of the Elephant, Fox, Tiger and UCSB breast cancer datasets.

parameter	**values**
batch size	$\{4, 8, 16\}$
learning rates	$\{10^{-3}, 10^{-5}\}$
learning rate decay	$\{0.98, 0.96, 0.94\}$
embedding layers	$\{1, 2\}$
layer width	$\{32, 64, 128\}$
number of heads	$\{8, 12\}$
head dimensions	$\{16, 32\}$
scaling factors	$\{0.1, 1, 10\}$
bag dropout	$\{0.0, 0.75\}$

Table 13: Results for MIL benchmark datsets in terms of AUC score. The models use pooling layers `HopfieldPooling` and `SparseHopfieldPooling` instead.

Method	Tiger	Fox	Elephant	UCSB
w/ `HopfieldPooling`	0.871 ± 0.014	0.637 ± 0.035	0.876 ± 0.015	0.828 ± 0.068
w/ `SparseHopfieldPooling`	0.884 ± 0.007	0.610 ± 0.033	0.914 ± 0.016	0.796 ± 0.107