Associative Memory Under the Probabilistic Lens: Improved Transformers & Dynamic Memory Creation

Rylan Schaeffer
Computer Science
Stanford University
rschaef@cs.stanford.edu

Mikail Khona
Physics
MIT
mikail@mit.edu

Nika Zahedi
Electrical Engineering
Stanford
nzahedi@stanford.edu

Ila Rani Fiete
Brain & Cognitive Sciences
MIT
fiete@mit.edu

Andrey Gromov
Physics & FAIR
UMD & Meta
gromovand@meta.com

Sanmi Koyejo
Computer Science
Stanford
sanmi@cs.stanford.edu

Abstract

Clustering is a fundamental unsupervised learning problem, and recent work showed modern continuous associative memory (AM) networks can learn to cluster data via a novel unconstrained continuous relaxation of the discrete clustering optimization problem. In this work, we demonstrate that the energy function of that AM network can be viewed as the scaled negative log likelihood of a Gaussian mixture model, and that the dynamics of the AM network can be viewed as performing expectation maximization via gradient ascent rather than via closed-form coordinate ascent. Based on this insight, we show that a widespread practical implementation choice - self-attention with pre-layer normalization - approximates clustering on the hypersphere with inhomogeneous von Mises-Fisher likelihoods, suggesting a future experiment to improve transformers. We additionally leverage this connection to propose a novel AM network with the ability to create new memories during learning, as necessitated by the data, by drawing on tools from combinatorial stochastic processes and Bayesian nonparametrics.

1 Introduction

Clustering is a ubiquitous unsupervised learning problem in which data are to be partitioned based on some notion of similarity. Recent work has shown the potential of modern continuous associative memory (AM) networks to adeptly cluster data [18], based on a continuous relaxation approach of the typically discrete clustering optimization problem. We show here that the energy function of the proposed AM network corresponds to the negative log likelihood of the data, and that the dynamics of the AM network's state and memories correspond one-to-one with the two steps of expectation maximization. By establishing this connection, we further discover that a defacto implementation choice in large-scale transformers - namely, self-attention with pre-layer normalization - approximates clustering on the hypersphere with inhomogeneous von Mises-Fisher likelihoods and non-uniform mixing proportions. Capitalizing on these insights, we additionally introduce a novel AM network imbued with the capability to form new memories during learning as necessitated by the data using ideas from bayesian nonparametrics. We are working to test our ideas numerically, but ran out of time & space for this workshop submission.

Associative Memory & Hopfield Networks in 2023. NeurIPS 2023 workshop.

2 Background

Clustering with K-Means. One common algorithm for data in Euclidean space $\boldsymbol{x}_1, ..., \boldsymbol{x}_N \in \mathbb{R}^D$ is the K-Means algorithm [14], which learns K centroids $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K \in \mathbb{R}^D$ by minimizing the following objective function:

$$\min_{\mu_1,...,\mu_K} \sum_x ||\boldsymbol{x} - \boldsymbol{\mu}_x||^2 \quad , \quad \boldsymbol{\mu}_x \overset{\text{def}}{=} \operatorname*{arg\,min}_{1,...,K} ||\boldsymbol{x} - \boldsymbol{\mu}_k||^2$$

Clustering with Associative Memory (CLAM) Networks. Saha et al. 2023 recently proposed a clustering algorithm using an associative memory (AM) network, termed CLAM [18]. The AM network's fixed points are collectively determined by the memories (in AM terminology) or centroids (in clustering terminology). Specifically, let $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K \in \mathbb{R}^D$ be K (fixed) memories. The energy function [13] for any state $\boldsymbol{v}$ is:

$$E(\boldsymbol{v}) \overset{\text{def}}{=} -\frac{1}{\beta} \log \left(\sum_k \exp\big(-\beta ||\boldsymbol{\mu}_k - \boldsymbol{v}||^2 \big) \right). \tag{1}$$

To determine the cluster assignment for datum $\boldsymbol{x} \in \mathbb{R}^D$, we initialize the AM network's state at time $t = 0$ to $\boldsymbol{v}(t = 0) = \boldsymbol{x}$, and run the network dynamics to minimize the energy function:

$$\tau \frac{d\boldsymbol{v}(t)}{dt} \overset{\text{def}}{=} -\frac{1}{2} \nabla_{\boldsymbol{v}} E(\boldsymbol{v}) = \sum_k (\boldsymbol{\mu}_k - \boldsymbol{v})\, \sigma\big(-\beta ||\boldsymbol{\mu}_k - \boldsymbol{v}||^2 \big), \tag{2}$$

where $\sigma(\cdot)$ denotes the softmax function. The energy is non-increasing since $\frac{d}{dt} E(\boldsymbol{v}) = \nabla_{\boldsymbol{v}} E(\boldsymbol{v}) \cdot \frac{d\boldsymbol{v}}{dt} = -(1/2) \nabla_{\boldsymbol{v}} E(\boldsymbol{v}) \cdot \nabla_{\boldsymbol{v}} E(\boldsymbol{v}) = -(1/2) ||\nabla_{\boldsymbol{v}} E(\boldsymbol{v})||^2 \leq 0$; consequently, the energy is guaranteed to converge to a local minima. Once the energy converges, we assign the datum to the nearest memory. To learn the memories, we perform gradient descent on the reconstruction error:

$$\sum_{n=1}^{n} \left| \left| \boldsymbol{x}_n - \boldsymbol{v}_n^{T, \{\mu_k\}} \right| \right|^2, \tag{3}$$

where $\boldsymbol{v}_n^{T,\{\mu_k\}}$ is the state of the AM network with memories $\{\mu_k\}_{k=1}^K$ having been initialized at $\boldsymbol{v}(t = 0) = \boldsymbol{x}_n$ and then following the CLAM dynamics (Eqn. 2) for T time.

Finite Mixture Models. The probabilistic approach to clustering is known as mixture modeling [6]. In a *finite* Gaussian mixture model, let $z \in [K]$ denote the latent variable cluster assignment and let θ denote the model parameters: $\{\pi_k\}_{k=1}^K$ are the mixing coefficients, and $\{\boldsymbol{\mu}_k, \Sigma_k\}_{k=1}^K$ are the means and covariances of the clusters. The data density is:

$$p(\boldsymbol{x}; \theta) \overset{\text{def}}{=} \sum_{k=1}^K p(\boldsymbol{x} | z = k; \theta)\, p(z = k; \theta) \overset{\text{def}}{=} \sum_{k=1}^K \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)\, \pi_k. \tag{4}$$

One can establish an equivalence between the energy function of the associative memory network and the negative log likelihood of the Gaussian mixture model by assuming uniform mixing coefficients $\pi_k = 1/K$ and shared isotropic covariance $\Sigma_k = 2\beta^{-1} I_D$ [6]:

$$-\log p(\boldsymbol{x}; \theta) \propto -\frac{1}{\beta} \log \left(\sum_k \exp(-\beta ||\boldsymbol{\mu}_k - \boldsymbol{x}||^2) \right) + C, \tag{5}$$

for irrelevant constants C, and therefore:

$$-\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}; \theta) = \sum_k (\boldsymbol{\mu}_k - \boldsymbol{x})\, \sigma\big(-\beta ||\boldsymbol{\mu}_k - \boldsymbol{x}||^2 \big). \tag{6}$$

We see from these two equations that CLAM's energy function is equal to the (inverse temperature-scaled) negative log likelihood, and the dynamics of minimizing the energy function via gradient descent are exactly equivalent to the dynamics of minimizing the negative log likelihood via gradient descent. Choosing non-uniform mixing proportions corresponds to Saha et al. 2023 [18]'s "weighted clustering," and choosing a von Mises-Fisher likelihood corresponds to their "spherical clustering"; one can, of course, choose other likelihoods e.g. Laplace, uniform, Lévy, etc.

3 Clustering with Associative Memory Networks is Expectation Maximization in a Probabilistic Mixture Model

Expectation Maximization (EM) [10] is a widely known coordinate ascent algorithm for latent variable models. When using EM to fit a Gaussian mixture model to data, EM alternates updates of two quantities: (1) the cluster assignments $p(z = k|\boldsymbol{x}; \theta)$, sometimes called posteriors or responsibilities, and (2) the parameters i.e. mixing proportions $\{\pi_k\}_k$, centroids $\{\boldsymbol{\mu}_k\}_k$ and covariances $\{\Sigma_k\}_k$. Although closed-form expressions exist for both in the case of a Gaussian mixture model, one can alternatively take gradient steps for both the E and M steps; this was called Generalized EM in the original work [10] and further explored in later work, e.g., [26, 15, 19]. In a Gaussian mixture model, holding the parameters θ fixed, the cluster assignment posterior is:

$$p(z = k|\boldsymbol{x}; \theta) = \frac{p(\boldsymbol{x}|z = k; \theta)\, p(z = k; \theta)}{\sum_{k'} p(\boldsymbol{x}|z = k'; \theta)\, p(z = k'; \theta)} = \frac{\mathcal{N}(\boldsymbol{x}; \mu_k, \Sigma_k)\pi_k}{\sum_{k'} \mathcal{N}(\boldsymbol{x}; \mu_{k'}, \Sigma_{k'})\pi_{k'}}. \tag{7}$$

Assuming uniform mixing proportions $\pi_k = 1/K$, and assuming identical isotropic cluster covariances $\Sigma_k = 2\beta^{-1}I$, the cluster assignment posterior simplifies to:

$$p(z = k|\boldsymbol{x}; \theta) = \frac{\exp\big(-\beta||\boldsymbol{\mu}_k - \boldsymbol{x}||^2\big)}{\sum_{k'} \exp\big(-\beta||\boldsymbol{\mu}_{k'} - \boldsymbol{x}||^2\big)} = \sigma(-\beta||\boldsymbol{\mu}_k - \boldsymbol{x}||^2). \tag{8}$$

The fixed points of the CLAM dynamics are given by a weighted combination of the centroids, where the weights are given by the cluster assignment posteriors. Specifically, if we initialize $\boldsymbol{v}(0) = \boldsymbol{x}$, then $\boldsymbol{v}^* \overset{\text{def}}{=} \sum_k p(z = k|\boldsymbol{x}; \theta)\boldsymbol{\mu}_k$ is a fixed point:

$$\frac{d\boldsymbol{v}^*}{dt} = \frac{1}{\tau}\left(\underbrace{\sum_k \boldsymbol{\mu}_k \sigma\big(-\beta||\boldsymbol{\mu}_k - \boldsymbol{v}^*||^2\big)}_{\overset{\text{def}}{=}\boldsymbol{v}^*} - \boldsymbol{v}^* \underbrace{\sum_k \sigma\big(-\beta||\boldsymbol{\mu}_k - \boldsymbol{v}^*||^2\big)}_{=1} \right) = \frac{1}{\tau}(\boldsymbol{v}^* - \boldsymbol{v}^*) = 0.$$

EM's two alternating phases correspond to CLAM's two alternating phases. EM's expectation step prescribes minimizing the negative log likelihood (Eqn. 5) with respect to the cluster assignment posterior probabilities by performing gradient descent (Eqn. 6), which corresponds to CLAM minimizing the energy function (Eqn. 1) with respect to the particle by rolling out the dynamics (Eqn. 2). The fixed points of the dynamics implicitly contain the probabilistic cluster assignments. Then, EM's maximization step prescribes minimixing the negative log likelihood by taking a gradient step with respect to the parameters θ, which corresponds to CLAM shaping the energy landscape by taking a gradient step with respect to the parameters θ.

4 Self-Attention with Pre-Layer Normalization Approximates Clustering on the Hypersphere

By making this connection between clustering with associative memory networks and probabilistic inference in mixture models, we discover a way to understand the interaction between self-attention and pre-layer normalization. The well-known equation for self-attention [23] is $\boldsymbol{z} = V\sigma(K\boldsymbol{q})$. Previous work [17] connected this to Hopfield networks. However, in practice, transformers are not purely stacked self-attention layers; among many components, layer norm [4] plays a crucial role. Layer norm transforms a vector $\boldsymbol{x} \in \mathbb{R}^D$ by computing its mean $\boldsymbol{m} = (\sum_d \boldsymbol{x}_d)/D$ and its variance $\sigma^2 = (\sum_d (\boldsymbol{x}_d - \boldsymbol{m})^2)/D$, then shifting and scaling by learnable parameters $\gamma \in \mathbb{R}$ and $\boldsymbol{\delta} \in \mathbb{R}^D$:

$$LN_{\gamma,\boldsymbol{\delta}}(\boldsymbol{x}) \quad \overset{\text{def}}{=} \quad \gamma\frac{\boldsymbol{x} - \boldsymbol{m}}{\sqrt{\sigma^2 + \epsilon}} + \boldsymbol{\delta}.$$

ϵ is a small constant for numerical stability. Practitioners have found that applying layer norm *before* self-attention layers in transformers (called "pre-layer norm") yields significantly better performance [5, 9, 24, 25]. What effect does composition of pre-layer norm and self-attention have? We show that the two together approximate clustering on the hypersphere using a mixture of inhomogeneous von

Mises-Fisher (vMF) distributions. Recall that the vMF density function, with unit vector $\boldsymbol{m}_i$ and concentration $\kappa_i \geq 0$ is:

$$p(\boldsymbol{x}; \boldsymbol{m}_i, \kappa_i) \propto \exp(\kappa_i \, \boldsymbol{m}_i \cdot \boldsymbol{x}).$$

Define $\tilde{\boldsymbol{q}}$ as the pre-shifted and scaled query i.e., $\boldsymbol{q} \overset{\text{def}}{=} \gamma \tilde{\boldsymbol{q}} + \boldsymbol{\delta}$, with $||\tilde{\boldsymbol{q}}||_2 \approx 1$. The i^{th} element in the numerator of the softmax is:

$$\exp(\boldsymbol{k}_i^T \boldsymbol{q}) = \exp(\boldsymbol{k}_i^T (\gamma \tilde{\boldsymbol{q}} + \boldsymbol{\delta})) = \exp\left(\underbrace{\gamma ||\boldsymbol{k}_i||}_{= \kappa_i} \underbrace{\frac{\boldsymbol{k}_i}{||\boldsymbol{k}_i||}}_{= \boldsymbol{m}_i} \cdot \tilde{\boldsymbol{q}} \right) \cdot \underbrace{\exp(\boldsymbol{k}_i \cdot \boldsymbol{\delta})}_{= \tilde{\pi}_i}. \tag{9}$$

Pre-norm followed by self-attention is equivalent to clustering with inhomogeneous vMF likelihoods and with unnormalized mixing proportions. A related commentary about pre-layer norm and self-attention has been made before [8], albeit in a non-clustering context. This also suggests a limited expressivity: the concentrations and the mixing proportions are inextricably tied via $\boldsymbol{k}_i$, whereas one might want them to be independent; future work can easily test separating them.

5 Infinite Clustering with Associative Memory Networks

By establishing this connection between CLAM and probabilistic inference in a Gaussian mixture model, one can draw upon the wealth of methods in probabilistic clustering to create new associative memory networks with new capabilities. One capability is creating new memories (new clusters) as necessitated by the data. This is interesting both biologically and computationally. Biologically, animals create new memories throughout their lives, and the process by which these processes occur are fundamental topics in experimental and computational neuroscience alike. Computationally, in the context of clustering, choosing the right number of clusters is a perennial dilemma.

To equip an AM network with the ability to create new memories, we propose leveraging Bayesian nonparametrics using combinatorial stochastic processes [16]. Specifically, we will use the Dirichlet Process (DP), and its generalization the Pitman-Yor Process (PYP). These two processes both define a probability distribution over partitions of set, that can then be used as an “infinite"-dimensional prior over the number of partitions, i.e. clusters. For notational convenience, we will instead work with the (1-parameter) Chinese Restaurant Process [1] (CRP) and its generalization, the 2-parameter Chinese Restaurant Process, corresponding to the DP and PYP, respectively. The CRP allows one to specify a prior over which cluster the n^{th} data point will be assigned to, conditioned on where the preceding $n-1$ data were assigned and hyperparameter $\alpha > 0$ specifying the (unnoramlized) probability of creating a new cluster:

$$p(c_n = c | c_{<n}, \alpha) \overset{\text{def}}{=} \frac{1}{\alpha + n - 1} \begin{cases} \sum_{n' < n} I(c_{n'} = c) & \text{if } 1 \leq c \leq C_{n-1} \\ \alpha & \text{if } c = C_{n-1} + 1 \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$CRP(\alpha)$ produces logarithmetically many clusters in n, which is non-ideal for data with power-law tails. The 2-parameter CRP introduces a second hyperparameter $d \in [0, 1)$ that accelerates how quickly new clusters form and decelerates how quickly existing clusters accumulate mass.

$$p(c_n = c | c_{<n}, \alpha, d) \overset{\text{def}}{=} \frac{1}{n - 1 + \alpha} \begin{cases} -d + \sum_{n' < n} I(c_n = c) & \text{if } 1 \leq c \leq C_{n-1} \\ \alpha + C_{n-1} \cdot d & \text{if } c = C_{n-1} + 1 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

We propose using these to define a novel AM network with the ability to create new clusters:

$$E(\boldsymbol{v}) \overset{\text{def}}{=} -\frac{1}{\beta} \log \left((\alpha + dK) \exp\left(-(\beta^{-1} + \rho^{-1})^{-1} ||\boldsymbol{v}||^2 \right) + \sum_{k=1}^{K} (\pi_k - d) \cdot \exp\left(-\beta ||\boldsymbol{\mu}_k - \boldsymbol{v}||^2 \right) \right) \tag{12}$$

For discussion, related work and future directions, see App. A.

References

[1] David J Aldous, Illdar A Ibragimov, Jean Jacod, and David J Aldous. *Exchangeability and related topics*. Springer, 1985.

[2] Nick Alonso and Jeff Krichmar. A sparse quantized hopfield network for online-continual memory. *arXiv preprint arXiv:2307.15040*, 2023.

[3] Louis Annabi, Alexandre Pitti, and Mathias Quoy. On the relationship between variational inference and auto-associative memory. *Advances in Neural Information Processing Systems*, 35:37497–37509, 2022.

[4] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[5] Alexei Baevski and Michael Auli. Adaptive input representations for neural language modeling. *arXiv preprint arXiv:1809.10853*, 2018.

[6] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[7] Trenton Bricken, Xander Davies, Deepak Singh, Dmitry Krotov, and Gabriel Kreiman. Sparse distributed memory is a continual learner. In *The Eleventh International Conference on Learning Representations*, 2022.

[8] Trenton Bricken and Cengiz Pehlevan. Attention approximates sparse distributed memory. *Advances in Neural Information Processing Systems*, 34:15301–15315, 2021.

[9] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[10] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

[11] Ruth Fuentes-García, Ramsés H Mena, and Stephen G Walker. Modal posterior clustering motivated by hopfield's network. *Computational Statistics & Data Analysis*, 137:92–100, 2019.

[12] Sheena A Josselyn and Susumu Tonegawa. Memory engrams: Recalling the past and imagining the future. *Science*, 367(6473):eaaw4325, 2020.

[13] Mikail Khona and Ila R Fiete. Attractor and integrator networks in the brain. *Nature Reviews Neuroscience*, 23(12):744–766, 2022.

[14] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[15] Radford M Neal and Geoffrey E Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer, 1998.

[16] Jim Pitman. *Combinatorial stochastic processes: Ecole d'eté de probabilités de saint-flour xxxii-2002*. Springer, 2006.

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

[18] Bishwajit Saha, Dmitry Krotov, Mohammed J Zaki, and Parikshit Ram. End-to-end differentiable clustering with associative memories. *arXiv preprint arXiv:2306.03209*, 2023.

[19] Ruslan Salakhutdinov, Sam T Roweis, and Zoubin Ghahramani. Optimization with em and expectation-conjugate-gradient. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 672–679, 2003.

[20] Rylan Schaeffer, Blake Bordelon, Mikail Khona, Weiwei Pan, and Ila Rani Fiete. Efficient online inference for nonparametric mixture models. In *Uncertainty in Artificial Intelligence*, pages 2072–2081. PMLR, 2021.

[21] Rylan Schaeffer, Yilun Du, Gabrielle K Liu, and Ila Fiete. Streaming inference for infinite feature models. In *International Conference on Machine Learning*, pages 19366–19387. PMLR, 2022.

[22] Rylan Schaeffer, Gabrielle Kaili-May Liu, Yilun Du, Scott Linderman, and Ila R Fiete. Streaming inference for infinite non-stationary clustering. In *Conference on Lifelong Learning Agents*, pages 310–326. PMLR, 2022.

[23] 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.

[24] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao. Learning deep transformer models for machine translation. *arXiv preprint arXiv:1906.01787*, 2019.

[25] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR, 2020.

[26] Lei Xu and Michael I Jordan. On convergence properties of the em algorithm for gaussian mixtures. *Neural computation*, 8(1):129–151, 1996.

A Discussion

Biological agents learn from radically different data than most deep learning models. Deep neural networks typically work best when trained on (self)supervised stationary data presented in an offline/aggregate manner. In contrast, most biological agents learn from unsupervised nonstationary data presented in an online/streaming manner. This observation has led many to design algorithms for unsupervised learning on non-stationary streaming data, from both probabilistic [20, 21, 22] and associative memory network approaches [7, 2]. Interestingly, the probabilistic algorithms share some striking similarities with memory engrams [12], an exciting new area of experimental neuroscience. Others have explored the connection between modern continuous Hopfield networks and probabilistic modeling [11] [3]. We are especially excited to combine insights from AM networks and probabilistic modeling to create better algorithms and to model phenomona in cognitive science and neuroscience.