Attention via $\log \sum \exp$ energy

Alexander Tschantz

January 30, 2025

Overview

$$\log \sum \exp \text{ framework}$$

$\log \sum \exp$ energy

We consider energy functions that map a set of parents $\{\boldsymbol{v}_p \in \mathbb{R}^{d_p} : p \in P\}$ to a set of children $\{\boldsymbol{v}_c \in \mathbb{R}^{d_c} : c \in C\}$.

Each energy function defines a similarity function (with parameters θ) that measures agreement between a parent and child vector:

$$\mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p) : \mathbb{R}^{d_c} \times \mathbb{R}^{d_p} \to \mathbb{R}.$$

The $\log \sum \exp$ energy function is then given by:

$$E(\{\boldsymbol{v}_c\}, \{\boldsymbol{v}_p\}, \theta) = - \sum_{c \in C} \ln\Big(\sum_{p \in P} \exp\big(\mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)\big) \Big).$$

$\log \sum \exp$ derivatives

We define *attention* as:

$$\alpha_{c,p} = \mathsf{softmax}_p\big(\mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)\big) = \frac{\exp\big(\mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)\big)}{\sum_{p' \in P} \exp\big(\mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_{p'})\big)}.$$

The derivatives with are then given by:

$$-\frac{\partial E}{\partial \boldsymbol{v}_c} = \sum_{p \in P} \alpha_{c,p} \frac{\partial \mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)}{\partial \boldsymbol{v}_c},$$

$$-\frac{\partial E}{\partial \boldsymbol{v}_p} = \sum_{c \in C} \alpha_{c,p} \frac{\partial \mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)}{\partial \boldsymbol{v}_p},$$

$$-\frac{\partial E}{\partial \theta} = \sum_{c \in C} \sum_{p \in P} \alpha_{c,p} \frac{\partial \mathrm{sim}_\theta(\boldsymbol{v}_c, \boldsymbol{v}_p)}{\partial \theta}.$$

log $\sum$ exp derivatives

The gradient of the energy function can be interpreted as an *expected value* over a discrete distribution $\alpha_{c,p} = P(p \mid c)$:

$$-\frac{\partial E}{\partial \mathbf{v}_c} = \mathbb{E}_{p \sim P(p|c)} \left[\frac{\partial \mathrm{sim}_\theta(\mathbf{v}_c, \mathbf{v}_p)}{\partial \mathbf{v}_c} \right] \tag{1}$$

While the child gradient is an exact expectation under $P(p \mid c)$, the parent gradient does not correspond exactly to an expectation under $P(c \mid p)$; it is instead proportional to it through Bayes rule.

$\log \sum \exp$ graphical models

We consider a single set of N nodes $\boldsymbol{v} = \{\boldsymbol{v}_i : i \in \{1, 2, \ldots, N\}\}$, where each node $\boldsymbol{v}_i \in \mathbb{R}^{d_i}$, and M energy functions $\{E_m : m \in \{1, 2, \ldots, M\}\}$.

Each E_m has a similairtiy function $\mathrm{sim}_m(\cdot)$ with parameters θ_m and defines a subset of nodes acting as *children* $C_m \subseteq \{1, 2, \ldots, N\}$ and a subset acting as *parents* $P_m \subseteq \{1, 2, \ldots, N\}$, which may overlap.

$$E_m\big(\{\boldsymbol{v}_c\}, \{\boldsymbol{v}_p\}, \theta_m\big) = -\sum_{c \in C_m} \ln\Big(\sum_{p \in P_m} \exp\big(\mathrm{sim}_m(\boldsymbol{v}_c, \boldsymbol{v}_p)\big)\Big).$$

The full energy is the sum over all energy functions:

$$E(\{\boldsymbol{v}_i\}, \{\theta_m\}) = \sum_{m=1}^{M} E_m\big(\{\boldsymbol{v}_c\}, \{\boldsymbol{v}_p\}, \theta_m\big).$$

log $\sum$ exp graphical models

For a single node $\mathbf{v}_a$, the gradient of the global energy E w.r.t. $\mathbf{v}_a$ decomposes into two terms. Let $\mathcal{M}_c(a) = \{m : a \in C_m\}$ denote the energy functions where $\mathbf{v}_a$ acts as a *child*, and $\mathcal{M}_p(a) = \{m : a \in P_m\}$ the energy functions where $\mathbf{v}_a$ acts as a *parent*. Then:

$$-\frac{\partial E}{\partial \mathbf{v}_a} = \underbrace{\sum_{m \in \mathcal{M}_c(a)} \sum_{p \in P_m} \mathsf{softmax}_p\Big(\mathrm{sim}(\mathbf{v}_a, \mathbf{v}_p)\Big) \frac{\partial}{\partial \mathbf{v}_a}\mathrm{sim}\big(\mathbf{v}_a, \mathbf{v}_p\big)}_{\mathbf{v}_a \text{ acting as a child}}$$

$$+ \underbrace{\sum_{m \in \mathcal{M}_p(a)} \sum_{c \in C_m} \mathsf{softmax}_a\Big(\mathrm{sim}(\mathbf{v}_c, \mathbf{v}_a)\Big) \frac{\partial}{\partial \mathbf{v}_a}\mathrm{sim}\big(\mathbf{v}_c, \mathbf{v}_a\big)}_{\mathbf{v}_a \text{ acting as a parent}}.$$

(2)

$\log \sum \exp$ expectation maximisation

Expectation Step:

▶ Perform gradient descent on each $\mathbf{v}_i$ to find the optimal latent states:

$$\mathbf{v}_i^* = \arg \min_{\mathbf{v}_i} E(\{\mathbf{v}_i\}, \{\theta_m\}).$$

Maximisation Step:

▶ Given the updated $\mathbf{v}_i^*$, take a gradient step on each θ_m:

$$\theta_m = \theta_m - \eta \frac{\partial E}{\partial \theta_m}\Big|_{\mathbf{v}_i^*}.$$

log $\sum$ exp framework

```python
class Energy:
    def similarity(self, *args):
        raise NotImplementedError

    def __call__(self, *args):
        # energy function
        sim_matrix = self.similarity(*args)
        return -sum(logsumexp(sim_matrix, axis=1))

dx = grad(energy)(x, y)
```

log $\sum$ exp examples

Gaussian Mixture Model

We define a set of child nodes $\{x_i\}_{i=1}^{N}$ and parent means $\{\mu_k\}_{k=1}^{K}$. The covariance matrices $\{\Sigma_k\}_{k=1}^{K}$ are treated as parameters $\theta_k = \Sigma_k$ (note that we could have also included means in the similarity parameters).

The similarity function is given by:

$$\mathrm{sim}_k(x_i, \mu_k) = -\tfrac{1}{2}(x_i - \mu_k)^\top \Sigma_k^{-1}(x_i - \mu_k).$$

The energy function is:

$$E^{\mathrm{GMM}} = -\sum_{i=1}^{N} \ln\Big(\sum_{k=1}^{K} \exp\big(\mathrm{sim}_k(x_i, \mu_k)\big)\Big).$$

The similarity function can be interpreted as the log probability of x_i under the conditional Gaussian distribution $\mathcal{N}(\mu_k, \Sigma_k)$, up to a normalization constant.

Gaussian Mixture Model

The gradient for $\boldsymbol{\mu}_k$ is given by:

$$-\frac{\partial \boldsymbol{E}^{\mathrm{GMM}}}{\partial \boldsymbol{\mu}_k} = \sum_{i=1}^{N} \underbrace{\mathrm{softmax}_k\big(\mathrm{sim}(\boldsymbol{x}_i, \boldsymbol{\mu}_k)\big)}_{\alpha_{i,k}} \ \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k).$$

Solving for zero gives the fixed point update for $\boldsymbol{\mu}_k$:

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{N} \alpha_{i,k}\, \boldsymbol{x}_i}{\sum_{i=1}^{N} \alpha_{i,k}}.$$

Hopfield Attention

We define a set of child nodes $\{\boldsymbol{x}_i\}_{i=1}^N$, each $\boldsymbol{x}_i \in \mathbb{R}^d$, and a set of parent memory vectors $\{\boldsymbol{m}_\mu\}_{\mu=1}^K$, each $\boldsymbol{m}_\mu \in \mathbb{R}^d$.

The energy function is given by:

$$\mathrm{sim}(\boldsymbol{x}_i, \boldsymbol{m}_\mu) = \boldsymbol{x}_i^\top \boldsymbol{m}_\mu.$$

$$E^{\mathrm{Hopfield}} = -\sum_{i=1}^N \ln\Big(\sum_{\mu=1}^K \exp\big(\boldsymbol{x}_i^\top \boldsymbol{m}_\mu\big)\Big).$$

The gradients are:

$$\alpha_{i,\mu} = \mathrm{softmax}_\mu\big(\boldsymbol{x}_i^\top \boldsymbol{m}_\mu\big).$$

$$-\frac{\partial E}{\partial \boldsymbol{m}_\mu} = \sum_{i=1}^N \alpha_{i,\mu}\, \boldsymbol{x}_i, \quad -\frac{\partial E}{\partial \boldsymbol{x}_i} = \sum_{\mu=1}^K \alpha_{i,\mu}\, \boldsymbol{m}_\mu,$$

Slot Attention

We define a set of child (token) nodes $\{\boldsymbol{x}_j\}_{j=1}^N$ and a set of parent (slot) nodes $\{\boldsymbol{\mu}_i\}_{i=1}^S$. The parameters θ consist of two projection matrices $\boldsymbol{W}_K$ and $\boldsymbol{W}_Q$.

The energy function is:

$$\text{sim}(\boldsymbol{x}_j, \boldsymbol{\mu}_i) = (\boldsymbol{W}_K \boldsymbol{x}_j)^\top (\boldsymbol{W}_Q \boldsymbol{\mu}_i).$$

$$E^{\text{Slot}} = -\sum_{j=1}^N \ln\Big(\sum_{i=1}^S \exp\big(\text{sim}(\boldsymbol{x}_j, \boldsymbol{\mu}_i)\big)\Big).$$

The gradients are:

$$\alpha_{j,i} = \text{softmax}_i\big(\text{sim}(\boldsymbol{x}_j, \boldsymbol{\mu}_i)\big).$$

$$-\frac{\partial E}{\partial \boldsymbol{\mu}_i} = \sum_{j=1}^N \alpha_{j,i}\, \boldsymbol{W}_Q^\top \boldsymbol{W}_K\, \boldsymbol{x}_j, \quad -\frac{\partial E}{\partial \boldsymbol{x}_j} = \sum_{i=1}^S \alpha_{j,i}\, \boldsymbol{W}_K^\top \boldsymbol{W}_Q\, \boldsymbol{\mu}_i.$$

Self-Attention

We define a set of nodes $\{x_i\}_{i=1}^{N}$, where each x_i serves as both a child (query) and a parent (key). The parameters θ consist of projection matrices W^Q and W^K, forming:

$$q_i = W^Q x_i, \quad k_i = W^K x_i.$$

The energy function is:

$$\mathrm{sim}(x_c, x_p) = q_c^\top k_p.$$

$$E^{\mathrm{SA}} = -\sum_{c=1}^{N} \ln\Big(\sum_{p=1}^{N} \exp(q_c^\top k_p)\Big).$$

The gradients are:

$$\alpha_{c,p} = \mathrm{softmax}_p(q_c^\top k_p).$$

$$-\frac{\partial E}{\partial x_i} = \underbrace{\sum_{p=1}^{N} \alpha_{i,p} W^{Q\top} W^K x_p}_{\text{Child side}} + \underbrace{\sum_{c=1}^{N} \alpha_{c,i} W^{K\top} W^Q x_c}_{\text{Parent side}}.$$

Linear Mixture Model

We define a set of child nodes $\{\boldsymbol{x}_i\}_{i=1}^{N}$ and a set of parent nodes $\{\boldsymbol{z}_k\}_{k=1}^{K}$. The parameters $\boldsymbol{\theta} = \{(\boldsymbol{A}_k, \boldsymbol{b}_k, \Sigma_k)\}_{k=1}^{K}$ define a linear mapping:

$$\boldsymbol{x}_i = \boldsymbol{A}_k \boldsymbol{z}_k + \boldsymbol{b}_k + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \Sigma_k).$$

The energy function is:

$$\mathrm{sim}_k(\boldsymbol{x}_i, \boldsymbol{z}_k) = -\tfrac{1}{2}(\boldsymbol{x}_i - \boldsymbol{A}_k \boldsymbol{z}_k - \boldsymbol{b}_k)^{\top} \Sigma_k^{-1} (\boldsymbol{x}_i - \boldsymbol{A}_k \boldsymbol{z}_k - \boldsymbol{b}_k).$$

$$E^{\mathrm{LM}} = -\sum_{i=1}^{N} \ln\Big(\sum_{k=1}^{K} \exp\big(\mathrm{sim}_k(\boldsymbol{x}_i, \boldsymbol{z}_k)\big)\Big).$$

Note this is the energy function for the switching linear dynamical system (SLDS) model, and our simple latent attention (SLA) model.

Linear Mixture Model

The attention weights are:

$$\alpha_{i,k} = \text{softmax}_k\big(\text{sim}_k(\mathbf{x}_i, \mathbf{z}_k)\big).$$

The gradient w.r.t. $\mathbf{x}_i$ is:

$$-\frac{\partial E}{\partial \mathbf{x}_i} = \sum_{k=1}^{K} \alpha_{i,k} \mathbf{\Sigma}_k^{-1}(\mathbf{x}_i - \mathbf{A}_k \mathbf{z}_k - \mathbf{b}_k).$$

The fixed-point update for $\mathbf{x}_i$ is:

$$\mathbf{x}_i^* = \sum_{k=1}^{K} \alpha_{i,k} \big(\mathbf{A}_k \mathbf{z}_k + \mathbf{b}_k\big).$$

Non-Linear Mixture Model

We define a set of child nodes $\{\boldsymbol{x}_i\}_{i=1}^{N}$ and a set of parent nodes $\{\boldsymbol{z}_k\}_{k=1}^{K}$. The parameters $\boldsymbol{\theta} = \{\boldsymbol{A}_k, \boldsymbol{b}_k, \sigma\}$ define a non-linear mapping:

$$f(\boldsymbol{z}_k) = \sigma(\boldsymbol{A}_k \boldsymbol{z}_k + \boldsymbol{b}_k),$$

where $\sigma(\cdot)$ is a non-linearity.

The similarity function is:

$$\operatorname{sim}_k(\boldsymbol{x}_i, \boldsymbol{z}_k) = -\tfrac{1}{2}\|\boldsymbol{x}_i - f(\boldsymbol{z}_k)\|^2.$$

The energy function is:

$$E^{\mathrm{NL}} = -\sum_{i=1}^{N} \ln\Big(\sum_{k=1}^{K} \exp\big(\operatorname{sim}_k(\boldsymbol{x}_i, \boldsymbol{z}_k)\big)\Big).$$

Non-Linear Mixture Model: Gradients

The attention weights are:

$$\alpha_{i,k} = \text{softmax}_k\big(\text{sim}_k(\boldsymbol{x}_i, \boldsymbol{z}_k)\big).$$

The gradient w.r.t. $\boldsymbol{x}_i$ is:

$$-\frac{\partial E}{\partial \boldsymbol{x}_i} = \sum_{k=1}^{K} \alpha_{i,k}\,(\boldsymbol{x}_i - f(\boldsymbol{z}_k)).$$

The gradient w.r.t. $\boldsymbol{z}_k$ is:

$$-\frac{\partial E}{\partial \boldsymbol{z}_k} = \sum_{i=1}^{N} \alpha_{i,k}\,\frac{\partial f(\boldsymbol{z}_k)}{\partial \boldsymbol{z}_k}\,(\boldsymbol{x}_i - f(\boldsymbol{z}_k)).$$

Note these are these are predictive coding updates, weighted by the attention $\alpha_{i,k}$.

Kronecker $\log \sum \exp$

We define a single set of child nodes $\{\boldsymbol{x}_i\}_{i=1}^{N}$ and *two* sets of parents:

$$\{\boldsymbol{z}_k\}_{k=1}^{K}, \quad \{\boldsymbol{u}_\ell\}_{\ell=1}^{L}.$$

We combine these factors *multiplicatively* via a *Kronecker* structure, giving a single $\log \sum \exp$ term over all pairs (k, ℓ).

Similarity: For each child $\boldsymbol{x}_i$ and each parent pair (k, ℓ), define

$$\mathrm{sim}\big(\boldsymbol{x}_i; \boldsymbol{z}_k, \boldsymbol{u}_\ell\big) = \mathrm{sim}_z(\boldsymbol{x}_i, \boldsymbol{z}_k) + \mathrm{sim}_u(\boldsymbol{x}_i, \boldsymbol{u}_\ell),$$

$$E^{\mathrm{Kron}} = -\sum_{i=1}^{N} \ln\Big(\sum_{k=1}^{K}\sum_{\ell=1}^{L} \exp\big(\mathrm{sim}(\boldsymbol{x}_i; \boldsymbol{z}_k, \boldsymbol{u}_\ell)\big)\Big).$$

Kronecker $\log \sum \exp$

The resulting attention is now a *joint* softmax over all parent pairs:

$$\alpha_{i,k,\ell} = \text{softmax}_{k,\ell}\big(\text{sim}(\boldsymbol{x}_i; \boldsymbol{z}_k, \boldsymbol{u}_\ell)\big)$$

$$= \frac{\exp\big(\text{sim}(\boldsymbol{x}_i; \boldsymbol{z}_k, \boldsymbol{u}_\ell)\big)}{\sum_{k'=1}^{K} \sum_{\ell'=1}^{L} \exp\big(\text{sim}(\boldsymbol{x}_i; \boldsymbol{z}_{k'}, \boldsymbol{u}_{\ell'})\big)}.$$

Gradients: For a child $\boldsymbol{x}_i$,

$$-\frac{\partial E}{\partial \boldsymbol{x}_i} = \sum_{k=1}^{K} \sum_{\ell=1}^{L} \alpha_{i,k,\ell} \frac{\partial}{\partial \boldsymbol{x}_i} \text{sim}(\boldsymbol{x}_i; \boldsymbol{z}_k, \boldsymbol{u}_\ell).$$

Similarly, for a parent $\boldsymbol{z}_k$:

$$-\frac{\partial E}{\partial \boldsymbol{z}_k} = \sum_{i=1}^{N} \sum_{\ell=1}^{L} \alpha_{i,k,\ell} \frac{\partial}{\partial \boldsymbol{z}_k} \text{sim}_z(\boldsymbol{x}_i, \boldsymbol{z}_k),$$

Kronecker $\log \sum \exp$ (Discrete Case)

Setup: We define a single discrete child variable $\boldsymbol{x} \in \{1, \ldots, D\}$ and two discrete parent variables:

$$\boldsymbol{z} \in \{1, \ldots, K\}, \quad \boldsymbol{u} \in \{1, \ldots, L\}.$$

The parameters $\boldsymbol{\theta} \in \mathbb{R}^{D \times K \times L}$ represent a joint categorical distribution.

Energy We define a single similarity function that depends on all three variables:

$$\mathrm{sim}(x, z, u) = \ln \theta_{x,z,u}.$$

$$E^{\mathrm{Kron}} = -\sum_{i=1}^{N} \ln \Big(\sum_{k=1}^{K} \sum_{\ell=1}^{L} \exp \big(\mathrm{sim}(x_i, z_k, u_\ell) \big) \Big).$$

Kronecker $\log \sum \exp$ (Discrete Case)

Tensor Attention: This induces a joint softmax over both parent variables:

$$\alpha_{i,k,\ell} = \frac{\exp\big(\text{sim}(x_i, z_k, u_\ell)\big)}{\sum_{k'=1}^{K} \sum_{\ell'=1}^{L} \exp\big(\text{sim}(x_i, z_{k'}, u_{\ell'})\big)}.$$

Interpretation:

- Parent variables (z_k, u_ℓ) conspire to jointly explain child variables x_i.
- The attention $\alpha_{i,k,\ell}$ generalizes categorical mixture models to factorial structures.

Multi log $\sum$ exp

We define a set of child nodes $\{\mathbf{x}_i\}_{i=1}^{N}$ and a set of parent nodes $\{\mathbf{z}_k\}_{k=1}^{K}$. Each parent k is associated with multiple similarity terms, indexed by different factors.

Similarity functions: Each similarity term contributes independently to the energy function:

$$\mathrm{sim}_1(\mathbf{x}_i, \mathbf{z}_k), \quad \mathrm{sim}_2(\mathbf{x}_i, \mathbf{z}_k), \quad \ldots, \quad \mathrm{sim}_M(\mathbf{x}_i, \mathbf{z}_k).$$

Energy function:

$$E^{\mathrm{Multi}} = -\sum_{i=1}^{N} \ln\Big(\sum_{k=1}^{K} \exp\big(\textstyle\sum_{m=1}^{M} \mathrm{sim}_m(\mathbf{x}_i, \mathbf{z}_k)\big)\Big).$$

Here, all similarity terms are summed before the softmax, leading to a *joint* mixture model over multiple factors.

Multi log $\sum$ exp

The resulting attention weight is:

$$\alpha_{i,k} = \text{softmax}_k\Big(\sum_{m=1}^{M} \text{sim}_m(\boldsymbol{x}_i, \boldsymbol{z}_k)\Big).$$

With gradients:

$$-\frac{\partial E}{\partial \boldsymbol{x}_i} = \sum_{k=1}^{K} \alpha_{i,k} \sum_{m=1}^{M} \frac{\partial}{\partial \boldsymbol{x}_i} \text{sim}_m(\boldsymbol{x}_i, \boldsymbol{z}_k).$$

$$-\frac{\partial E}{\partial \boldsymbol{z}_k} = \sum_{i=1}^{N} \alpha_{i,k} \sum_{m=1}^{M} \frac{\partial}{\partial \boldsymbol{z}_k} \text{sim}_m(\boldsymbol{x}_i, \boldsymbol{z}_k).$$

This formulation is useful when multiple similarties contribute to the similarity between a child and parent.

$$\log \sum \exp \text{ graphical models}$$

Block-slot Attention

Setup: We define a set of child nodes $\{\boldsymbol{x}_i\}_{i=1}^{N}$, a set of slot parent nodes $\{\boldsymbol{z}_k\}_{k=1}^{K}$, and a set of memory parent nodes $\{\boldsymbol{m}_\mu\}_{\mu=1}^{M}$. The parameters $\boldsymbol{\theta} = \{\boldsymbol{W}_K, \boldsymbol{W}_Q\}$ consist of projection matrices.

Energy: Each child $\boldsymbol{x}_i$ is compared to slot parents $\boldsymbol{z}_k$ and memory parents $\boldsymbol{m}_\mu$:

$$\mathrm{sim}_z(\boldsymbol{x}_i, \boldsymbol{z}_k) = (\boldsymbol{W}_K \boldsymbol{x}_i)^\top (\boldsymbol{W}_Q \boldsymbol{z}_k),$$

$$\mathrm{sim}_m(\boldsymbol{x}_i, \boldsymbol{m}_\mu) = \boldsymbol{x}_i^\top \boldsymbol{m}_\mu.$$

$$E^{\mathrm{BlockSlot}} = -\sum_{i=1}^{N} \ln \left(\sum_{k=1}^{K} \exp\big(\mathrm{sim}_z(\boldsymbol{x}_i, \boldsymbol{z}_k)\big) \right)$$

$$-\sum_{i=1}^{N} \ln \left(\sum_{\mu=1}^{M} \exp\big(\mathrm{sim}_m(\boldsymbol{x}_i, \boldsymbol{m}_\mu)\big) \right).$$

Energy Transformer

Setup: We define a set of nodes $\{x_i\}_{i=1}^N$ and a set of memory nodes $\{m_\mu\}_{\mu=1}^M$. The parameters $\theta = \{W^Q, W^K\}$ consist of projection matrices for queries and keys.

Energy: Each node x_i attends to itself (self-attention) and to memory nodes m_μ (Hopfield attention):

$$\mathrm{sim}_{\mathsf{self}}(x_i, x_j) = (W^Q x_i)^\top (W^K x_j),$$

$$\mathrm{sim}_{\mathsf{memory}}(x_i, m_\mu) = x_i^\top m_\mu.$$

$$E^{\mathrm{ET}} = -\sum_{i=1}^N \ln \left(\sum_{j=1}^N \exp\big(\mathrm{sim}_{\mathsf{self}}(x_i, x_j)\big) \right)$$

$$-\sum_{i=1}^N \ln \left(\sum_{\mu=1}^M \exp\big(\mathrm{sim}_{\mathsf{memory}}(x_i, m_\mu)\big) \right).$$

Atari Model

Setup: We define a sequence of child nodes $\{x_t\}_{t=1}^T$, a sequence of latent parent nodes $\{z_t\}_{t=1}^T$, and a set of mode parameters $\{A_k, b_k, \Sigma_k\}_{k=1}^K$. The parameters $\theta = \{A_k, b_k, \Sigma_k\}$ define the mappings.

Similarity Functions: The child x_t is assigned to a mixture component z_t using:

$$\mathrm{sim}_{\mathsf{GMM}}(x_t, z_t) = -\tfrac{1}{2}(x_t - z_t)^\top \Sigma_k^{-1}(x_t - z_t).$$

The latent child z_t is explained by a linear mixture model:

$$\mathrm{sim}_{\mathsf{LM}}(z_t, z_{t-1}) = -\tfrac{1}{2}(z_t - A_k z_{t-1} - b_k)^\top \Sigma_k^{-1}(z_t - A_k z_{t-1} - b_k).$$

Atari Model

Energy:

$$E^{\mathrm{Atari}} = -\sum_{t=1}^{T} \ln \left(\sum_{k=1}^{K} \exp\big(\mathrm{sim}_{\mathsf{GMM}}(\boldsymbol{x}_t, \boldsymbol{z}_t)\big) \right)$$

$$-\sum_{t=1}^{T-1} \ln \left(\sum_{k=1}^{K} \exp\big(\mathrm{sim}_{\mathsf{LM}}(\boldsymbol{z}_t, \boldsymbol{z}_{t-1})\big) \right).$$

The gradient for the latent parent $\boldsymbol{z}_t$ combines both responsibilities:

$$-\frac{\partial E}{\partial \boldsymbol{z}_t} = \sum_{k=1}^{K} \alpha_{t,k}\, \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_t - \boldsymbol{z}_t) + \sum_{k=1}^{K} \beta_{t,k}\, \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{z}_t - \boldsymbol{A}_k \boldsymbol{z}_{t-1} - \boldsymbol{b}_k).$$

$$\log \sum \exp \text{ spatio-temporal}$$
$$\text{model}$$

Spatiotemporal Attention

Hierarchy: We define a hierarchy of L layers. Each layer l contains K_l latent variables, where each variable evolves over T time steps. Concretely, let:

$$\mathbf{x}_{k,t}^{(l)} \in \mathbb{R}^{D_l}$$

denote the k-th variable (or "slot") in layer l at time t, with D_l-dimensional representations. The number of variables K_l may differ across layers.

Structure:

▶ **Inter-layer (vertical):** Each variable $\mathbf{x}_{k,t}^{(l)}$ is influenced by variables from the layer below $(l-1)$ at the same time t.

▶ **Intra-layer (concurrent):** Variables within the same layer l interact at each time step t.

▶ **Temporal:** Each variable $\mathbf{x}_{k,t}^{(l)}$ is influenced by its own past states $\mathbf{x}_{k,t'}^{(l)}$ for $t' < t$.

Spatiotemporal Attention

Inter-layer (vertical):

$$\mathrm{sim}_v^{(l)}(\boldsymbol{x}, \boldsymbol{x}') = \big(\boldsymbol{W}_v^{K,(l)}\,\boldsymbol{x}\big)^\top \big(\boldsymbol{W}_v^{Q,(l)}\,\boldsymbol{x}'\big).$$

Intra-layer (concurrent):

$$\mathrm{sim}_c^{(l)}(\boldsymbol{x}, \boldsymbol{x}') = \big(\boldsymbol{W}_c^{Q,(l)}\,\boldsymbol{x}\big)^\top \big(\boldsymbol{W}_c^{K,(l)}\,\boldsymbol{x}'\big).$$

Temporal (past states):

$$\mathrm{sim}_t^{(l)}(\boldsymbol{x}, \boldsymbol{x}') = \big(\boldsymbol{W}_t^{Q,(l)}\,\boldsymbol{x}\big)^\top \big(\boldsymbol{W}_t^{K,(l)}\,\boldsymbol{x}'\big).$$

Spatiotemporal Attention

Inter-layer Energy (Slot attentiom):

$$E_v^{(l)} = -\sum_{t=1}^{T}\sum_{c=1}^{K_{l-1}} \ln\left(\sum_{k=1}^{K_l} \exp\big(\mathrm{sim}_v^{(l)}(\boldsymbol{x}_{c,t}^{(l-1)}, \boldsymbol{x}_{k,t}^{(l)})\big)\right).$$

Intra-layer Energy (Self attention):

$$E_c^{(l)} = -\sum_{t=1}^{T}\sum_{k=1}^{K_l} \ln\left(\sum_{\substack{k'=1 \\ k'\neq k}}^{K_l} \exp\big(\mathrm{sim}_c^{(l)}(\boldsymbol{x}_{k,t}^{(l)}, \boldsymbol{x}_{k',t}^{(l)})\big)\right).$$

Temporal Energy (Casual self attention):

$$E_t^{(l)} = -\sum_{k=1}^{K_l}\sum_{t=2}^{T} \ln\left(\sum_{t'<t} \exp\big(\mathrm{sim}_t^{(l)}(\boldsymbol{x}_{k,t}^{(l)}, \boldsymbol{x}_{k,t'}^{(l)})\big)\right).$$

Spatiotemporal Attention

$$-\frac{\partial E}{\partial \boldsymbol{x}_{k,t}^{(l)}} = \underbrace{-\frac{\partial E_v^{(l)}}{\partial \boldsymbol{x}_{k,t}^{(l)}}}_{\text{bottom-up}} + \underbrace{-\frac{\partial E_v^{(l+1)}}{\partial \boldsymbol{x}_{k,t}^{(l)}}}_{\text{top-down}} + \underbrace{-\frac{\partial E_c^{(l)}}{\partial \boldsymbol{x}_{k,t}^{(l)}}}_{\text{intra-layer}} + \underbrace{-\frac{\partial E_t^{(l)}}{\partial \boldsymbol{x}_{k,t}^{(l)}}}_{\text{temporal}}.$$

Bottom-up gradient

$$-\frac{\partial E_v^{(l)}}{\partial \boldsymbol{x}_{k,t}^{(l)}} = \sum_{c=1}^{K_{l-1}} \mathrm{softmax}_k\big(\boldsymbol{A}_{c,k}\big)\, \boldsymbol{W}_v^{Q,(l)\top}\, \boldsymbol{W}_v^{K,(l)} \boldsymbol{x}_{c,t}^{(l-1)}.$$

Top-down gradient

$$-\frac{\partial E_v^{(l+1)}}{\partial \boldsymbol{x}_{k,t}^{(l)}} = \sum_{p=1}^{K_{l+1}} \mathrm{softmax}_p\big(\boldsymbol{A}_{k,p}\big)\, \boldsymbol{W}_v^{K,(l+1)\top}\, \boldsymbol{W}_v^{Q,(l+1)} \boldsymbol{x}_{p,t}^{(l+1)}.$$

Spatiotemporal Attention

Intra-layer gradient

$$-\frac{\partial E_c^{(l)}}{\partial \mathbf{x}_{k,t}^{(l)}} = \sum_{k' \neq k} \mathrm{softmax}_k\big(\mathbf{A}_{k',k}\big)\, \mathbf{W}_c^{K,(l)\top}\, \mathbf{W}_c^{Q,(l)}\, \mathbf{x}_{k',t}^{(l)}$$
$$+ \sum_{k' \neq k} \mathrm{softmax}_{k'}\big(\mathbf{A}_{k,k'}\big)\, \mathbf{W}_c^{Q,(l)\top}\, \mathbf{W}_c^{K,(l)}\, \mathbf{x}_{k',t}^{(l)}.$$

Temporal gradient

$$-\frac{\partial E_t^{(l)}}{\partial \mathbf{x}_{k,t}^{(l)}} = \sum_{t' < t} \mathrm{softmax}_{t'}\big(\mathbf{A}_{t,t'}\big)\, \mathbf{W}_t^{Q,(l)\top}\, \mathbf{W}_t^{K,(l)}\, \mathbf{x}_{k,t'}^{(l)}.$$

Remark: By merging inter-layer, intra-layer, and temporal connections into a single large log-sum-exp term, we increase flexibility but at the cost of $\mathcal{O}((N + K)\, T)^2$ complexity.

Renormalised mixture model

Renormalised mixture models

We define a hierarchy of L layers, each containing K_l latent variables of dimension D_l, where K_l decreases with l ($K_1 > K_2 > \cdots > K_L$).

Each variable $\boldsymbol{x}_{k,t}^{(l)}$ at layer l receives input only from a local subset of variables in layer $l-1$, forming a *receptive field*:

$$\mathcal{R}_k^{(l)} \subseteq \{1, \ldots, K_{l-1}\}.$$

In this setting, intra-layer and temporal energy remain the same.

Renormalised Mixture Model

Each variable $x_{k,t}^{(l)}$ at layer l is explained only by a local receptive field $\mathcal{R}_k^{(l)}$ in layer $l-1$:

$$E_v^{(l)} = -\sum_{t=1}^{T}\sum_{k=1}^{K_l}\ln\left(\sum_{c\in\mathcal{R}_k^{(l)}}\exp\big(\mathrm{sim}_v^{(l)}(x_{c,t}^{(l-1)}, x_{k,t}^{(l)})\big)\right).$$

Define:

$$\boldsymbol{A}_{c,k} = \mathrm{sim}_v^{(l)}(x_{c,t}^{(l-1)}, x_{k,t}^{(l)}), \quad \boldsymbol{A}_{k,p} = \mathrm{sim}_v^{(l+1)}(x_{k,t}^{(l)}, x_{p,t}^{(l+1)}).$$

Then the bottom-up and top-down gradients are:

$$-\frac{\partial E_v^{(l)}}{\partial x_{k,t}^{(l)}} = \sum_{c\in\mathcal{R}_k^{(l)}}\mathrm{softmax}_c(\boldsymbol{A}_{c,k})\frac{\partial}{\partial x_{k,t}^{(l)}}\mathrm{sim}_v^{(l)}(x_{c,t}^{(l-1)}, x_{k,t}^{(l)}),$$

$$-\frac{\partial E_v^{(l+1)}}{\partial x_{k,t}^{(l)}} = \sum_{p\in\mathcal{R}_k^{(l+1)}}\mathrm{softmax}_p(\boldsymbol{A}_{k,p})\frac{\partial}{\partial x_{k,t}^{(l)}}\mathrm{sim}_v^{(l+1)}(x_{k,t}^{(l)}, x_{p,t}^{(l+1)}).$$

Bayesian model expansion

If a new data point $\boldsymbol{x}_{N+1}$ is not well explained by existing parent components, we introduce a new parent to explain it.
We evaluate the energy contribution of the new data point:

$$E_{N+1} = -\ln\left(\sum_{k=1}^{K} \exp\big(\mathrm{sim}(\boldsymbol{x}_{N+1}, \boldsymbol{z}_k,)\big)\right).$$

If $E_{N+1} > \tau$ (for some threshold τ), then $\boldsymbol{x}_{N+1}$ is not sufficiently explained, and we add a new parent component $\boldsymbol{z}_{K+1}$, where the new parent is initialized based on $\boldsymbol{x}_{N+1}$.

Coordinate Ascent

Energy Function:
We consider an energy of the form

$$E(\{\boldsymbol{x}_j\}, \{\boldsymbol{\mu}_i\}; \theta) = -\sum_{j=1}^{N} \ln\Big(\sum_{i=1}^{S} \exp\big(\mathrm{sim}_\theta(\boldsymbol{x}_j, \boldsymbol{\mu}_i)\big)\Big),$$

where θ denotes the parameters of the similarity function.

EM Procedure:

▶ **E-step:** Update latent variables $\{\boldsymbol{\mu}_i\}$ given fixed θ.

▶ **M-step:** Update parameters θ in closed form given current slots.

Slot Attention Example:

$$\theta = \{\boldsymbol{W}_K, \boldsymbol{W}_Q\}, \quad \mathrm{sim}_\theta(\boldsymbol{x}_j, \boldsymbol{\mu}_i) = (\boldsymbol{W}_K \boldsymbol{x}_j)^\top (\boldsymbol{W}_Q \boldsymbol{\mu}_i).$$

Define the attention matrix **A** with entries

$$A_{ji} = \mathrm{softmax}_i\big((\boldsymbol{W}_K \boldsymbol{x}_j)^\top (\boldsymbol{W}_Q \boldsymbol{\mu}_i)\big).$$

Goal: Perform an **E-step** (update slots $\boldsymbol{\mu}_i$) and **M-step** (update $\boldsymbol{W}_K, \boldsymbol{W}_Q$).

Coordinate Ascent

M-step: Update Parameters
Given fixed slots $\{\boldsymbol{\mu}_i\}$ and attention **A**, we update $\boldsymbol{W}_K, \boldsymbol{W}_Q$.
We consider minimizing

$$\min_{\boldsymbol{W}_K, \boldsymbol{W}_Q} \sum_{j=1}^{N} \sum_{i=1}^{S} A_{ji} \left\| \boldsymbol{W}_K \boldsymbol{x}_j - \boldsymbol{W}_Q \boldsymbol{\mu}_i \right\|^2.$$

Setting the gradient to zero yields a weighted least-squares problem in $\boldsymbol{W}_K$ and $\boldsymbol{W}_Q$.

Key idea:

- ▶ **A** is fixed (like responsibilities in EM).
- ▶ Solve for $\boldsymbol{W}_K, \boldsymbol{W}_Q$ in closed form.

Coordinate Ascent

Closed-Form Updates

Differentiate and set to zero:

$$W_K = \Big(\sum_{j,i} A_{ji}\, W_Q \boldsymbol{\mu}_i\, \mathbf{x}_j^\top\Big)\Big(\sum_{j,i} A_{ji}\, \mathbf{x}_j \mathbf{x}_j^\top\Big)^{-1}.$$

$$W_Q = \Big(\sum_{j,i} A_{ji}\, W_K \mathbf{x}_j\, \boldsymbol{\mu}_i^\top\Big)\Big(\sum_{j,i} A_{ji}\, \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top\Big)^{-1}.$$

Algorithm: Alternate <u>E-step</u> (update $\{\boldsymbol{\mu}_i\}$) and <u>M-step</u> (update W_K, W_Q) until convergence.

Open questions

- ▶ Hierarchical POMDPs and hierarchical mixture models differ only in their similairty functions.
- ▶ Crucially, hierarchical mixture models induce a discrete state space (in terms of their attention matrices).
- ▶ This is seen in SLDS, where we model the dynamics of the induced discrete state space (which mode is active at a given time) as a HMM.
- ▶ Is there a way to combine the two models, with continuous mixture models on the bottom and discrete mixture models on the top?

Model

$$-\sum_{t=1}^{T} \ln\Bigg(\sum_{m_1=1}^{M_1} \sum_{m_2=1}^{M_2} \sum_{m_3=1}^{M_3}$$

$$\exp\Big($$

$$\mathrm{sim}_{collison}(\boldsymbol{x}_t, m_1)$$

$$+\,\mathrm{sim}_{gmm}(\boldsymbol{x}_t, m_1) \tag{3}$$

$$+\,\mathrm{sim}_{mlp}(\boldsymbol{x}_t, m_3)$$

$$+\,\mathrm{sim}_{slds}(\boldsymbol{x}_t, \boldsymbol{x}_{t+1}, m_1, m_2, m_3)$$

$$\Big)\Bigg).$$