Comp-mech for logits:

Modular addition as a case study

Xavier Poncini

PIBBSS Symposium 2025

Outline:

1) (Just enough) Comp-mech

2) Modular addition

3) Results

4) Outlook

(Just enough) <u>Comp-Mech</u>

Hidden Markov model (HMM):

A HMM consists of:

* A set $\mathcal{X}$ $\overset{\text{"think"}}{\rightsquigarrow}$ vocabulary of emissions

* A collection of transition matrices $(T^{(x)})_{x \in \mathcal{X}}$ $\overset{\text{"think"}}{\rightsquigarrow}$ dynamic that determines emissions & hidden state transitions

  where $T_{ij}^{(x)} = P(X = x, S_j = s_j \mid S_i = s_i)$

Fix an initial dist. over hidden states $(P(s_i))$; the prob. of obs. $w = w_1 w_2 \dots w_n$ is:

$$P(w) = \sum_{i,j,\dots,z} P(s_i) P(w_1, s_j \mid s_i) P(w_2, s_k \mid s_j) \cdots P(w_n, s_z \mid s_y)$$

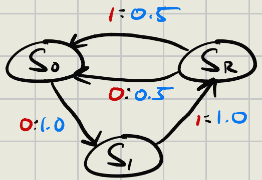
$$= \langle \pi | \underbrace{T^{(w_1)} T^{(w_2)} \cdots T^{(w_n)}}_{T^{(w)}} | 1 \rangle$$

where $\langle \pi | = [P(s_1) \ \dots \ P(s_{|S|})]$

$$| 1 \rangle = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

Example:

$\mathcal{X} = \{0, 1\}$



$$T^{(0)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \end{bmatrix}$$

$$T^{(1)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{2} & 0 & 0 \end{bmatrix}$$

What information about the past is relevant to predict the future?

Consider conditional probabilities:

$$P(w^{(f)} | w^{(p)}) = \frac{P(w^{(p)} w^{(f)})}{P(w^{(p)})} = \frac{\langle \eta | T^{(w^{(p)})} T^{(w^{(f)})} | 1 \rangle}{\langle \eta | T^{(w^{(p)})} | 1 \rangle}$$

The predictive vector $\langle \eta^{(w^{(p)})} | = \dfrac{\langle \eta | T^{(w^{(p)})}}{\langle \eta | T^{(w^{(p)})} | 1 \rangle}$ is relevant to all future predictions:

$$P(w^{(f)} | w^{(p)}) = \langle \eta^{(w^{(p)})} | T^{(w^{(f)})} | 1 \rangle$$

Observation 1: [Shai et al.]

The predictive vector of the HMM is often linearly decodable from the activations of a neural network trained on data from the HMM.

Observation 2:

Transformers (& other NNs) produce probabilities by passing logits through an "intense" non-linearity — the softmax function:
(Boltzmann dist.)

$$P(w) = \frac{e^{z(w)}}{\sum_{w} e^{z(w)}}$$

Question:

Can we develop a notion of a HMM for logits that admits an analogue of predictive vectors?

Energy-based hidden Markov model (EHMM):

A EHMM consists of:

* A set $\mathcal{X}$

* A collection of transition matrices $(H^{(x)})_{x \in \mathcal{X}}$

* An initial vector $\langle \eta |$ & a final vector $| \varphi \rangle$

$H^{(w_1)} H^{(w_2)} \cdots H^{(w_N)}$

such that $\langle \eta | H^{(w)} | \varphi \rangle \in \mathbb{R}$, for all $w \in \mathcal{X}^N$ & $N \in \mathbb{N}$

We can then associate these matrix elements with logits (energies)

$$z(w) = \langle \eta | H^{(w)} | \varphi \rangle \rightsquigarrow P(w) = \frac{e^{z(w)}}{\sum_w e^{z(w)}}$$

What information about the past is relevant to predict the future?

Consider "conditional logits" i.e. $z(w^{(f)}|w^{(p)})$ such that:

$$P(w^{(f)}|w^{(p)}) = \frac{e^{z(w^{(f)}|w^{(p)})}}{\sum_{w^{(f)}} e^{z(w^{(f)}|w^{(p)})}}$$

Claim: $z(w^{(f)}|w^{(p)}) = z(w^{(p)}w^{(f)})$ (proof is easy)

Expressing conditional logits in terms of EHMM objects:

$$z(w^{(f)}|w^{(p)}) = z(w^{(p)}w^{(f)}) = \langle \eta | H^{(w^{(p)})} H^{(w^{(f)})} | e \rangle$$

The predictive vector $\langle \eta^{(w^{(p)})}| = \langle \eta | H^{(w^{(p)})}$ is relevant to all future predictions:

$$z(w^{(f)}|w^{(p)}) = \langle \eta^{(w^{(p)})} | H^{(w^{(f)})} | 1 \rangle$$

Summary:

Process	Output	Predictive vector
HMM	$P(w) = \langle \eta \vert T^{(w)} \vert 1 \rangle$	$\langle \eta^{(w^{(p)})} \vert = \dfrac{\langle \eta \vert T^{(w^{(p)})}}{\langle \eta \vert T^{(w^{(p)})} \vert 1 \rangle}$
EHMM	$Z(w) = \langle \eta \vert H^{(w)} \vert 1 \rangle$	$\langle \eta^{(w^{(p)})} \vert = \langle \eta \vert H^{(w^{(p)})}$

Questions:

* Do neural networks represent the predictive vector of the EHMM?

* Do neural networks prefer the predictive vector of the HMM over the EHMM when given the chance?

Modular addition

$$a + b = c \mod p$$

Cyclic group (C_p):

The group C_p is generated by r subject to:

$$r^p = id$$

E.g. $r^{a+b} = r^c \iff c = a+b \mod p$.

A C_p-action describes the action of C_p on a set:

$$\alpha : S \times C_p \longrightarrow S$$

That respects the group structure i.e.

$$\alpha(v, r^{a+b}) = \alpha(\alpha(v, r^a), r^b).$$

Consider two C_p-actions: [Chughtai et al.]

$$\begin{array}{ccccccc} [\,0 & \cdots & 0 & 1 & 0 & \cdots & 0\,] \\ 0^{th} & & & i^{th} & & & (p-1)^{th} \end{array}$$

1) $S_p \times C_p \longrightarrow S_p$, $\quad S_p = \{\langle e_i | \;/\; i = 0, 1, \ldots, p-1\}$

on the vertices of a $(p-1)$-simplex S_p. Inducing:

$$e : C_p \longrightarrow Mat_{p \times p}(\{0,1\}), \quad e(r) = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & & 0 \\ 0 & & & 0 & 1 \\ 1 & 0 & \cdots & & 0 \end{bmatrix}$$

$$[\cos(\tfrac{2\pi w i}{p}) \quad \sin(\tfrac{2\pi w i}{p})]$$

2) $V_p^{(w)} \times C_p \longrightarrow V_p^{(w)}$, $\quad V_p^{(w)} = \{\langle v_i^{(w)} | \;/\; i = 0, 1, \ldots, p-1\}$

on the vertices of a p-gon V_p. Inducing:

$$e_w : C_p \longrightarrow Mat_{2 \times 2}(\mathbb{R}), \qquad e(r) = \begin{bmatrix} \cos(\frac{2\pi w}{p}) & \sin(\frac{2\pi w}{p}) \\ -\sin(\frac{2\pi w}{p}) & \cos(\frac{2\pi w}{p}) \end{bmatrix}$$

Random-Random Mod p (RRModp):

Vocabulary: $\mathcal{X} = \{0, 1, \dots, p-1\}$

Hidden states: $S = \{s_0^{(0)}\} \cup \{s_0^{(1)}, \dots, s_{p-1}^{(1)}\} \cup \{s_0^{(2)}, \dots s_{p-1}^{(2)}\}$

Intuitively, the RRModp HMM is given by:

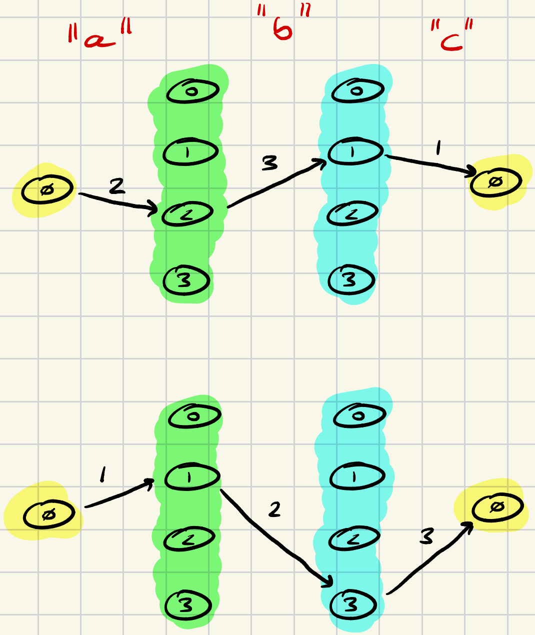
0) Process initialised in state $s_0^{(0)}$

1) Sample a from $\mathcal{X}$ & transition $s_0^{(0)} \xrightarrow{a} s_a^{(1)}$

2) Sample b from $\mathcal{X}$ & transition $s_a^{(1)} \xrightarrow{b} s_{a+b \bmod p}^{(2)}$

3) Transition $s_{a+b \bmod p}^{(2)} \xrightarrow{c} s_0^{(0)}$ where $c = a+b \bmod p$

Examples: $p = 4$



Formally, the RRMod$_p$ HMM is defined:

$$\mathcal{X} = \{0, 1, \ldots, p-1\}, \quad T^{(i)} = \begin{array}{c} \\ s^{(0)} \\ s^{(1)} \\ s^{(2)} \end{array} \!\!\! \begin{array}{c} \begin{array}{ccc} s^{(0)} & s^{(1)} & s^{(2)} \end{array} \\ \left[\begin{array}{c|c|c} 0 & \frac{1}{p}\langle e_i| & \underset{\sim}{0} \\ \hline \underset{\sim}{0} & \underset{\sim}{0} & \frac{1}{p} e(r^i) \\ \hline |e_i\rangle & \underset{\sim}{0} & \underset{\sim}{0} \end{array}\right] \end{array}$$

$$\langle \eta| = [1 \;\; \underset{\sim}{0} \;\; \underset{\sim}{0}]$$

Probabilities:

$$P(a) = \langle \eta | T^{(a)} | 1 \rangle = [0 \;\; \tfrac{1}{p}\langle e_a| \;\; \underset{\sim}{0}] \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \frac{1}{p} \qquad \langle e_a | e(r^b) \rangle$$

$$P(ab) = \langle \eta | T^{(a)} T^{(b)} | 1 \rangle = [0 \;\; \underset{\sim}{0} \;\; \tfrac{1}{p^2}\langle e_{a+b \bmod p}|] \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \frac{1}{p^2}$$

$$P(abc) = \langle \eta | T^{(a)} T^{(b)} T^{(c)} | 1 \rangle = [\tfrac{1}{p^2}\langle e_{a+b \bmod p} | e_c \rangle \;\; \underset{\sim}{0} \;\; \underset{\sim}{0}] \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \frac{\delta_{a+b \bmod p,\, c}}{p^2}$$

Predictive vectors of RRModp:

Recall the HMM predictive vector: $\langle \eta^{(w)}| = \dfrac{\langle \eta | T^{(w)}}{\langle \eta | T^{(w)} | 1 \rangle}$

Reusing parts of previous calculations:

$$\langle \eta^{(a)}| = \frac{\langle \eta | T^{(a)}}{\langle \eta | T^{(a)} | 1 \rangle} = [\,0 \;\; \langle e_a| \;\; \underset{\sim}{0}\,]$$

$$\langle \eta^{(ab)}| = \frac{\langle \eta | T^{(a)} T^{(b)}}{\langle \eta | T^{(a)} T^{(b)} | 1 \rangle} = [\,0 \;\; \underset{\sim}{0} \;\; \langle e_{a+b \bmod p}|\,]$$

$$\langle \eta^{(abc)}| = \frac{\langle \eta | T^{(a)} T^{(b)} T^{(c)}}{\langle \eta | T^{(a)} T^{(b)} T^{(c)} | 1 \rangle} = \begin{cases} \langle \eta| , & c = a+b \bmod p \\ \text{undefined} , & \text{else} \end{cases}$$

Projecting out the zero entries, the set of predictive vectors is given by:

$$\underset{\substack{\uparrow \\ 0^{th} \qquad i^{th} \qquad (p-1)^{th}}}{[\,0 \cdots 0 \;\; 1 \;\; 0 \cdots 0\,]}$$

$$S_p = \{ \langle e_i| \;/\; i = 0, 1, \ldots, p-1 \}$$

vertices of a $(p-1)$-simplex

Soft Random – Random Mod p (sRRModp):

Vocabulary: $\mathcal{X} = \{0, 1, \ldots, p-1\}$

Vectors: $\langle v_i^{(\omega)}| = \left[\cos\left(\frac{2\pi\omega i}{p}\right) \quad \sin\left(\frac{2\pi\omega i}{p}\right)\right]$

Intuitively, the sRRModp EHMM is given by:

0) Process initialised in vector $\langle \eta| = [1 \ \underline{0} \ \underline{\underline{0}}]$

1) Sample a from $\mathcal{X}$ & transition $\langle \eta| \xrightarrow{a} \langle v_a^{(\omega)}|$

2) Sample b from $\mathcal{X}$ & transition $\langle v_a^{(\omega)}| \xrightarrow{b} \langle v_{a+b}^{(\omega)}|$

3) Transition $\langle v_{a+b}^{(\omega)}| \xrightarrow{c} \langle \eta|$ $\quad \arg\max P(* \mid ab) = c$

where $c = a+b \mod p$

Fix a tuple of frequencies $\underset{\sim}{\omega} = (\omega_1, \omega_2, \ldots, \omega_N)$ where $\omega_i \in \{1, 2, \ldots, \lfloor \frac{p}{2} \rfloor\}$.

Formally, the sRRModp EHMM is defined:

$$\mathcal{X} = \{0, 1, \ldots, p-1\},$$

$$H^{(i)} = \left[\begin{array}{c:c:c} 0 & \frac{1}{p}\bigoplus_{j=1}^{N} \langle v_i^{(\omega_j)}| & \underset{\sim}{0} \\ \hdashline \underset{\sim}{0} & \underset{\sim}{0} & \frac{1}{p}\bigoplus_{j=1}^{N} e_{\omega_j}(r^i) \\ \hdashline \bigoplus_{j=1}^{N} |v_i^{(\omega_j)}\rangle & \underset{\sim}{0} & \underset{\sim}{0} \end{array}\right]$$

$$\langle \eta | = [1 \;\; \underset{\sim}{0} \;\; \underset{\sim}{0}], \quad |\varphi\rangle = \begin{bmatrix} 1 \\ \underset{\sim}{0} \\ \underset{\sim}{0} \end{bmatrix}$$

Recall:

$$\langle v_k^{(\omega)}| = \left[\cos\left(\frac{2\pi\omega k}{p}\right) \;\; \sin\left(\frac{2\pi\omega k}{p}\right)\right]$$

$$e(r) = \begin{bmatrix} \cos\left(\frac{2\pi\omega}{p}\right) & \sin\left(\frac{2\pi\omega}{p}\right) \\ -\sin\left(\frac{2\pi\omega}{p}\right) & \cos\left(\frac{2\pi\omega}{p}\right) \end{bmatrix}$$

Logits:

$$z(a) = \langle \eta | H^{(a)} | \varphi \rangle = \left[0 \quad \frac{1}{p} \bigoplus_{j=1}^{N} \langle v_a^{(w_j)} | \quad \underset{\sim}{0} \right] \begin{bmatrix} 1 \\ \underset{\sim}{0} \\ \underset{\sim}{0} \end{bmatrix} = 0$$

$$z(ab) = \langle \eta | H^{(a)} H^{(b)} | \varphi \rangle = \left[0 \quad \underset{\sim}{0} \quad \frac{1}{p^2} \bigoplus_{j=1}^{N} \langle v_{a+b \bmod p}^{(w_j)} | \right] \begin{bmatrix} 1 \\ \underset{\sim}{0} \\ \underset{\sim}{0} \end{bmatrix} = 0$$

$$z(abc) = \langle \eta | H^{(a)} H^{(b)} H^{(c)} | \varphi \rangle$$

$$= \left[\frac{1}{p^2} \sum_{j=1}^{N} \langle v_{a+b \bmod p}^{(w_j)} | v_c^{(w_j)} \rangle \quad \underset{\sim}{0} \quad \underset{\sim}{0} \right] \begin{bmatrix} 1 \\ \underset{\sim}{0} \\ \underset{\sim}{0} \end{bmatrix} = \frac{1}{p^2} \sum_{j=1}^{N} \cos\left(\frac{2\pi w_j (a+b-c)}{p} \right)$$

[Nanda et al.]
logits

When $c = a+b \bmod p$ cosines constructively interfere & $z(abc)$ is large.

When $c \neq a+b \bmod p$ cosines destructively interfere & $z(abc)$ is small.

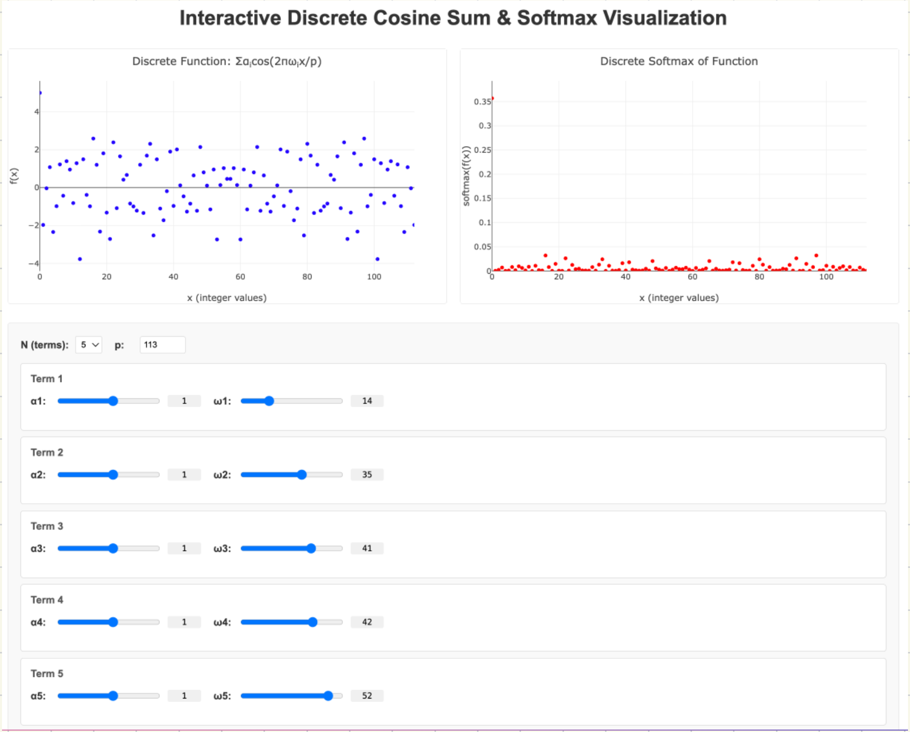
Probabilities:

We have $$P(w) = \frac{e^{z(w)}}{\sum_{w \in \mathcal{X}^L} e^{z(w)}}$$ and:

$$z(a) = z(ab) = 0, \quad z(abc) = \frac{1}{p^2} \sum_{j=1}^{N} \cos\left(\frac{2\pi w_j (a+b-c)}{p}\right)$$

So $P(a) = \frac{1}{p}$, $P(ab) = \frac{1}{p^2}$ and

$$P(abc) = \frac{\exp\left(\frac{1}{p^2} \sum_{j=1}^{N} \cos\left(\frac{2\pi w_j (a+b-c)}{p}\right)\right)}{\sum_{a,b,c} \exp\left(\frac{1}{p^2} \sum_{j=1}^{N} \cos\left(\frac{2\pi w_j (a+b-c)}{p}\right)\right)}$$



Interactive Discrete Cosine Sum & Softmax Visualization
Discrete Function: Σαᵢcos(2πωᵢx/p)
f(x)
x (integer values)
Discrete Softmax of Function
softmax(f(x))
x (integer values)
N (terms): 5
p: 113
Term 1
α1: 1
ω1: 14
Term 2
α2: 1
ω2: 35
Term 3
α3: 1
ω3: 41
Term 4
α4: 1
ω4: 42
Term 5
α5: 1
ω5: 52

Predictive vectors of sRRMod$_p$:

Recall the EHMM predictive vector: $\langle \eta^{(w)}| = \langle \eta | H^{(w)}$

Reusing parts of previous calculations:

$$\langle \eta^{(a)}| = \langle \eta | H^{(a)} = \left[0 \quad \frac{1}{p} \bigoplus_{j=1}^{N} \langle \nu_{a}^{(w_j)}| \quad \underset{\sim}{0} \right]$$

$$\langle \eta^{(ab)}| = \langle \eta | H^{(a)} H^{(b)} = \left[0 \quad \underset{\sim}{0} \quad \frac{1}{p^2} \bigoplus_{j=1}^{N} \langle \nu_{a+b \bmod p}^{(w_j)}| \right]$$

$$\langle \eta^{(abc)}| = \langle \eta | H^{(a)} H^{(b)} H^{(c)} = \left[\frac{1}{p^2} \sum_{j=1}^{N} \langle \nu_{a+b \bmod p}^{(w_j)} | \nu_{c}^{(w_j)} \rangle \quad \underset{\sim}{0} \quad \underset{\sim}{0} \right]$$

Projecting out the zero entries, the set of predictive vectors is given by:

$\left[\cos\left(\frac{2\pi w i}{p}\right) \quad \sin\left(\frac{2\pi w i}{p}\right) \right]$

vertices of a p-gon

$$V_p^{(w)} = \left\{ \langle \nu_i^{(w)}| \;\middle|\; i = 0, 1, \ldots, p-1 \right\}$$

Summary:

Process	Output	Predictive vectors
$RRMod_p$	$P(w) = \langle \eta \| T^{(w)} \| 1 \rangle$	$(p-1)$-simplex
$sRRMod_p$	$Z(w) = \langle \eta \| H^{(w)} \| 1 \rangle$	p-gon

What do models represent?

Results

Recipe:

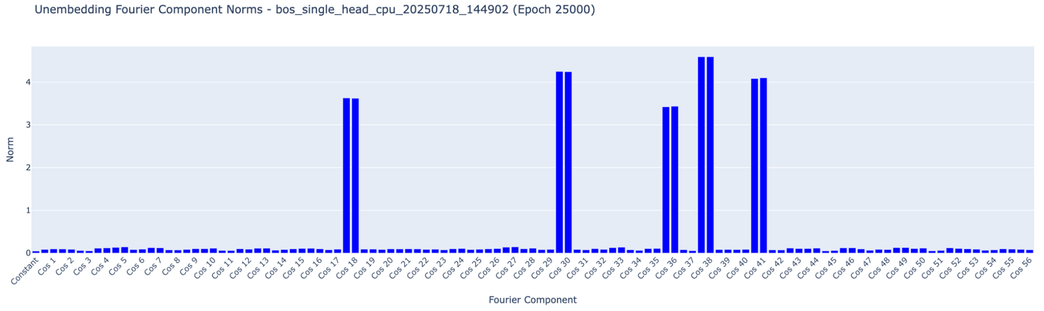
1) Train a one-layer one-head transformer to "grok" modular addition
2) Fit the logits of the transformer to:

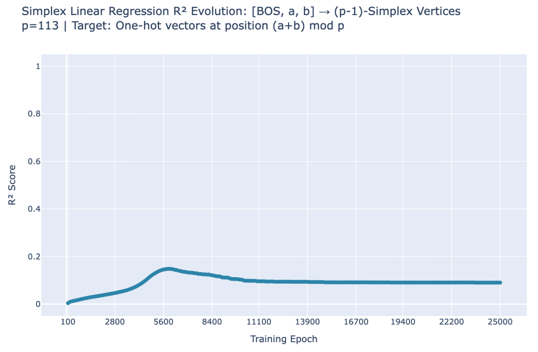
$$\sum_{i=1}^{N} \cos\left(\frac{2\pi w_i (a+b-c)}{p}\right)$$

to determine $\underset{\sim}{w} = (w_1, w_2, \ldots, w_N)$ of sRRModp

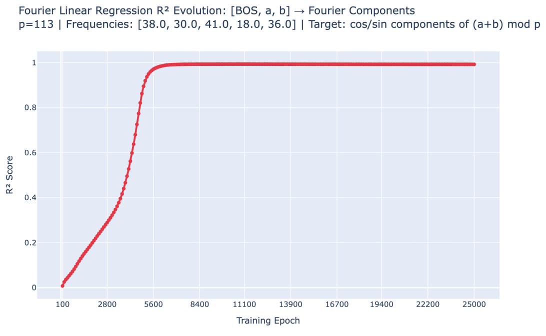
3) Analyse model activations with:
   * linear regression
   * PCA

Find frequencies: [Nanda et al.]

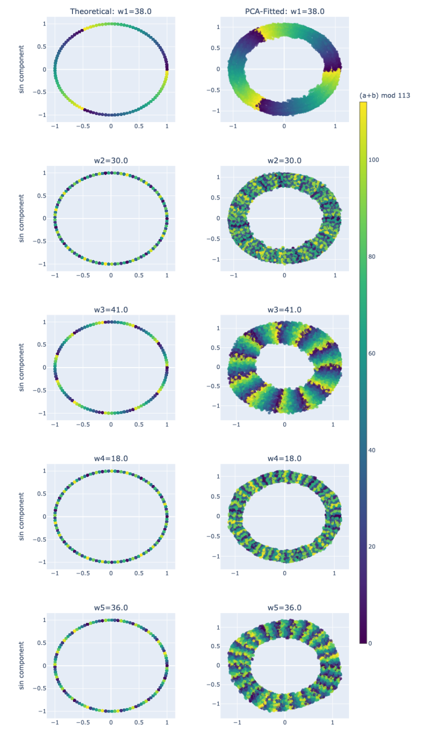




Simplex Linear Regression R² Evolution: [BOS, a, b] → (p-1)-Simplex Vertices
p=113 | Target: One-hot vectors at position (a+b) mod p
R² Score
Training Epoch

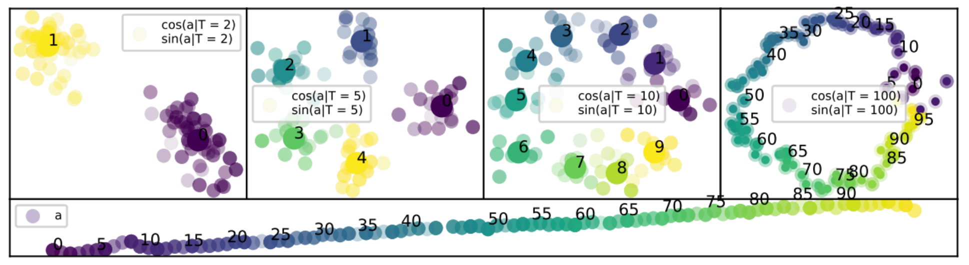


Fourier Linear Regression R² Evolution: [BOS, a, b] → Fourier Components
p=113 | Frequencies: [38.0, 30.0, 41.0, 18.0, 36.0] | Target: cos/sin components of (a+b) mod p
R² Score
Training Epoch

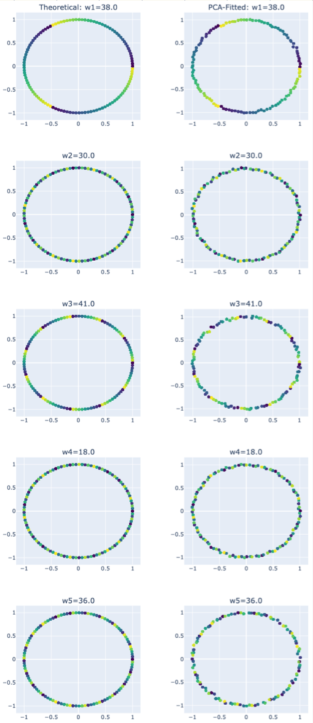


Theoretical: w1=38.0
PCA-Fitted: w1=38.0
w2=30.0
w3=41.0
w4=18.0
w5=36.0
sin component
(a+b) mod 113

These results seem like toy versions of the [Kantamneni et al.] results:

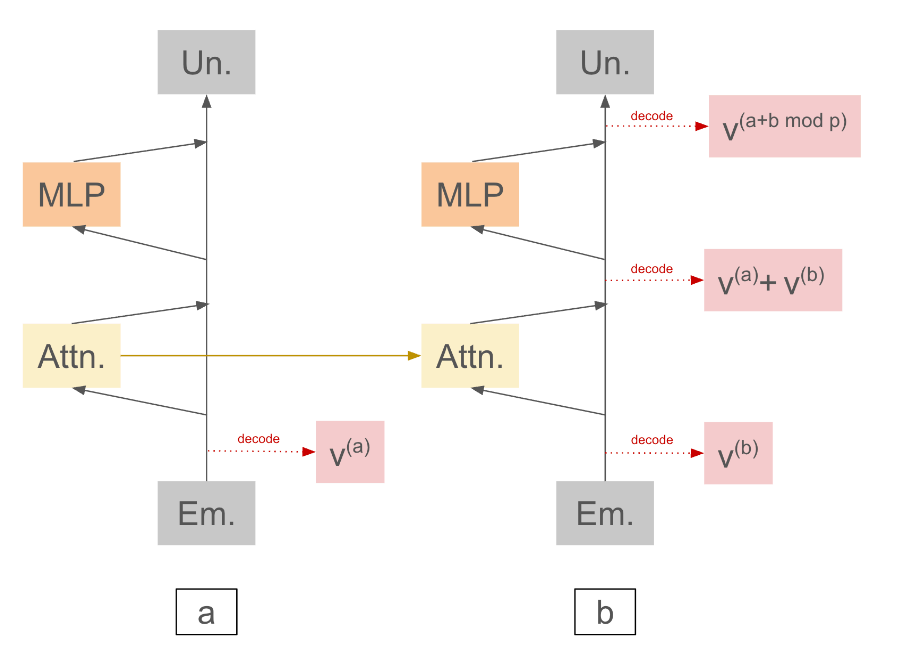


Bonus!

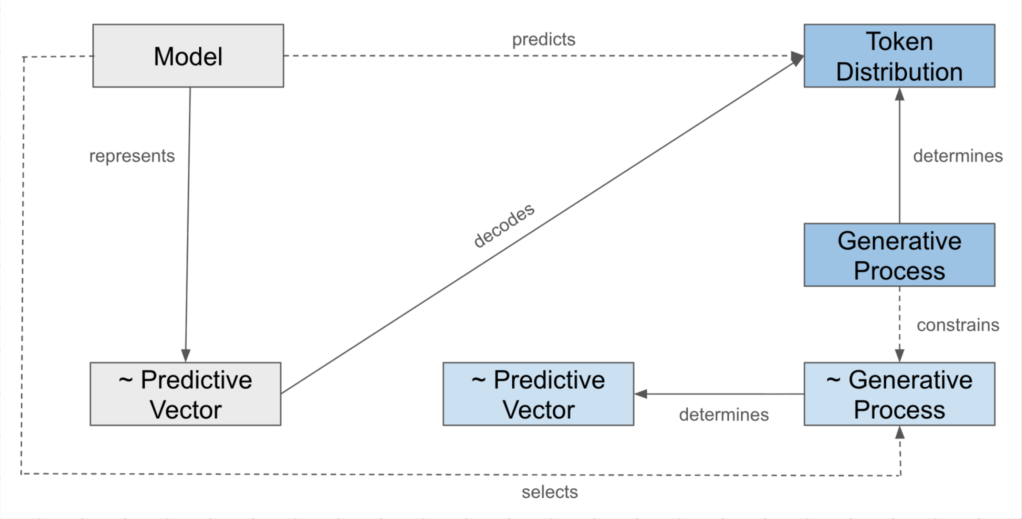


... actually represents more fourier components

see [Yip et al.] for details


Un.
MLP
Attn.
decode
v(a)
Em.
a
Un.
decode
v(a+b mod p)
MLP
decode
v(a)+ v(b)
Attn.
decode
v(b)
Em.
b

Outlook


Model
predicts
Token Distribution
represents
determines
decodes
Generative Process
constrains
~ Predictive Vector
~ Predictive Vector
~ Generative Process
determines
selects

Questions:

* What if we don't initialise in a synchronised state?

* If we directly train models to predict EHMM processes, are the predictive vectors decodable from activations?

* Is there a EHMM corresponding to familiar HMMs, e.g., is there an EHMM for Mess3?

Thanks for Listening!