Infinite-Width Limit of a Single Attention Layer: Analysis via Tensor Programs

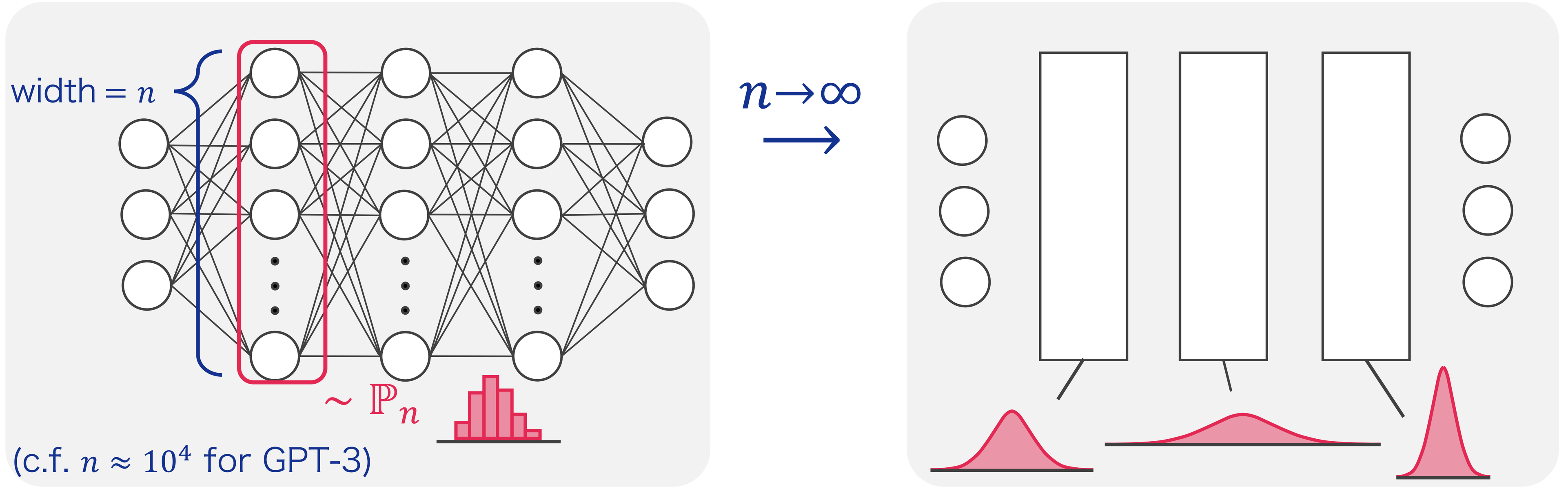
Mana Sakai[1,3], Ryo Karakida[2,3], Masaaki Imaizumi[1,3] ([1]The University of Tokyo [2]AIST [3]RIKEN AIP)

Paper Code

Message: The infinite-width limit of an attention layer is described by a non-Gaussian distribution

Background: Infinite-Width Limit of NNs

- **Goal:** Understand how information propagates in wide NNs
- **Approach:** Take the infinite-width limit and observe information propagation; check whether it explodes, vanishes, or stays stable



- For standard NN architectures, $\mathbb{P}_n \xrightarrow{n\to\infty}$ Gaussian (+correction term)
- **Notable frameworks:** NNGP (Lee+ 2017; Matthews+ 2018), TP (Yang 2019)

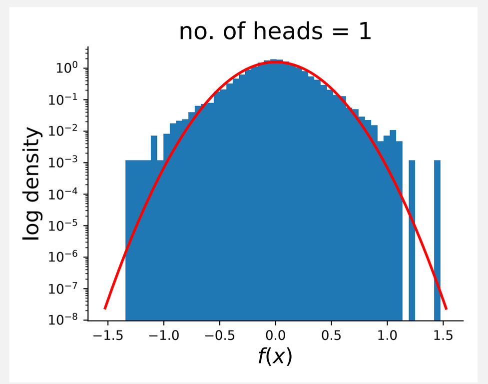
Attention Cannot Be Approximated by a Gaussian

- Attention output $\mathrm{Attn}: \mathbb{R}^{s\times n} \to \mathbb{R}^{s\times n}$ is

$$\mathrm{Attn}(X) = \frac{1}{\sqrt{H}}\sum_{a=1}^{H} \mathrm{SoftMax}\left(\frac{1}{\sqrt{n}}(XW^{Q,a})(XW^{K,a})^\top\right)(XW^{V,a})W^{O,a}$$

- Attention outputs are empirically known to be non-Gaussian
- **Prior work employs tailored assumptions** (e.g., infinite heads, $1/n$-scaling of attention scores) **to make NNGP and TP applicable**

→ We derive the limit distribution of attention without relying on such assumptions



Blue: Attention output (log scale)
Red line: log-Gaussian
(Extracted from Hron+ 2020, Figure 1)

Result: Non-Gaussian Limit of Attention

ith row vector of the attention output $\mathrm{Attn}(X) \in \mathbb{R}^{s\times n}$ is

$$y^i = \frac{1}{\sqrt{H}}\sum_{a=1}^{H}\sum_{j=1}^{s} \mathrm{SoftMax}_j\left(p_{i,1}^{(a)},\dots,p_{i,s}^{(a)}\right) W^{O,a}W^{V,a}x^i, \qquad p_{i,j}^{(a)} = \frac{1}{\sqrt{n}}\left(W^{Q,a}x^i\right)^\top\left(W^{K,a}x^j\right) \in \mathbb{R}$$

Limit Distribution of an Attention Output

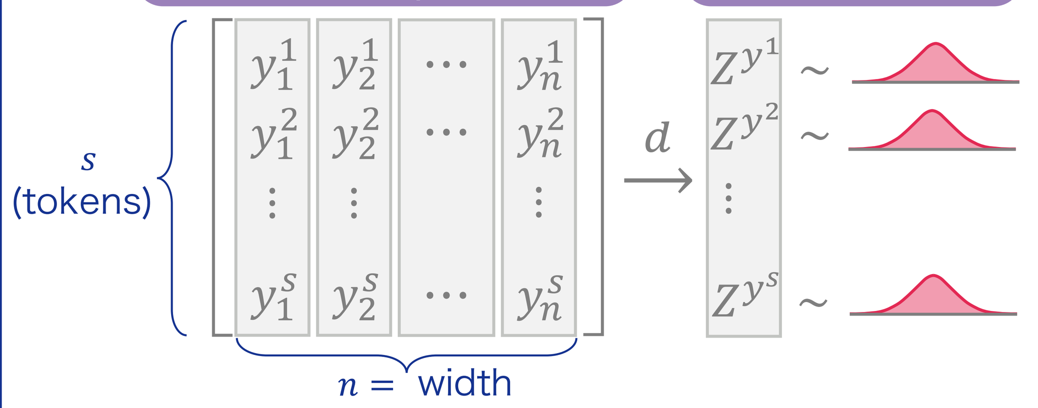
Suppose the network does not contain an attention layer prior to $\{y^i\}_{i\in[s]}$. Then,

$$(y^1_\alpha,\dots,y^s_\alpha) \xrightarrow{d} \left(Z^{y^1},\dots,Z^{y^s}\right) \quad (n\to\infty),$$

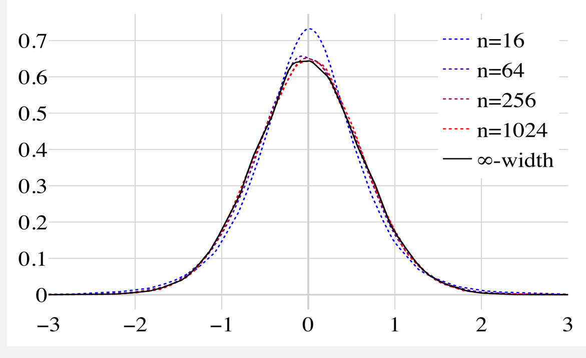
$$Z^{y^i} = \frac{1}{\sqrt{H}}\sum_{a=1}^{H}\sum_{j=1}^{s} \mathrm{SoftMax}_j\left(p_{i,1}^{\circ(a)},\dots,p_{i,s}^{\circ(a)}\right) Z^{\tilde{v}^{a,j}}$$

where

- $\{Z^{\tilde{v}^{a,j}}\}_{j\in[s],a\in[H]}$ is independent of $\{p_{i,j}^{\circ(a)}\}_{i,j\in[s],a\in[H]}$
- $\{Z^{\tilde{v}^{a,j}}\}_{j\in[s],a\in[H]} \sim N(0,\kappa)$
- $\{p_{i,j}^{\circ(a)}\}_{i,j\in[s],a\in[H]} \sim N(0,\kappa')$



Simulation



Dotted lines: The empirical distribution of the attention output

Black solid line: The limit distribution

Implications

- $\left(Z^{y^1},\dots,Z^{y^s}\right)$ is Gaussian conditional on $\{p_{i,j}^{\circ(a)}\}$

→ It follows a conditionally Gaussian (**hierarchical Gaussian**) structure

→ Marginally, it is inherently **non-Gaussian**

- A new framework distinct from existing Gaussian-based approaches is essential

Architectures with Gaussian-based limit distribution: MLP, CNN, LSTM, RNN, etc.

⇔

Architectures with non-Gaussian-based limit distribution: attention

General Result Using Tensor Programs

NETSOR Program

Consider a feed-forward neural network as a finite set of random vectors $h^1,\dots,h^J \in \mathbb{R}^n$, which is inductively generated as follows:

- $\mathcal{V}_0 \subset \{h^1,\dots,h^J\}$: fixed set of initial vectors (input layer)
- Each $h^k \notin \mathcal{V}_0$ is generated either by:
  - **MatMul**: matrix multiplication $h^k = Wh^j,\ W\in\mathbb{R}^{n\times n}$
  - **Nonlin**: coordinatewise nonlinearity $h^k = \phi(h^{j_1},\dots,h^{j_m})$, $\quad h^k_\alpha = \phi\left(h^{j_1}_\alpha,\dots,h^{j_m}_\alpha\right)\ (\alpha\in[n])$

Setup & Assumptions

- $r, m \in \mathbb{N}$ satisfy $m \geq 2r$
- Consider a NETSOR program, and suppose all nonlinearities used in Nonlin are pseudo-Lipschitz
- $g^1,\dots,g^m \in \mathbb{R}^n$ are vectors in NETSOR generated by MatMul
- A subset $\{g^{i,j}\}_{i\in[r],j\in[2]} \subset \{g^1,\dots,g^m\}$ is defined by $g^{i,j} = W^{i,j}x^{i,j}$, $\quad x^{i,j} = \phi^{i,j}(g^1,\dots,g^m)$
- Each $\phi^{i,j}$ is bounded and pseudo-Lipschitz
- The weight matrices $W^{i,j}\in\mathbb{R}^{n\times n}$ satisfy:

  1) $\{W^{i,j}\}_{i\in[r],j\in[2]}$ is not used for any $g \in \{g^1,\dots,g^m\}\setminus\{g^{i,j}\}_{i\in[r],j\in[2]}$

  2) $W^{i,j}$ may be the same matrix as $W^{i',j'}$ unless $i=i', j\neq j'$
- Define the scalar dot-products by $p_i = n^{-1/2}\left(g^{i,1}\right)^\top g^{i,2} \quad (i\in[r])$

Theorem (informal)

Let $h^1,\dots,h^k\in\mathbb{R}^n$ be vectors whose elements are given by

$$h^j_\alpha = \varphi^j(g^1_\alpha,\dots,g^m_\alpha,p_1,\dots,p_r) \quad (\alpha\in[n], j\in[k]),$$

where each φ^j is pseudo-Lipschitz. Then, for any bounded and pseudo-Lipschitz function $\psi:\mathbb{R}^k\to\mathbb{R}$, we have

$$\frac{1}{n}\sum_{\alpha=1}^{n}\psi(h^1_\alpha,\dots,h^k_\alpha) \xrightarrow{d} \mathbb{E}\left[\psi\left(Z^{h^1},\dots,Z^{h^k}\right) \,\middle|\, p^\circ_1,\dots,p^\circ_r\right] \quad (n\to\infty),$$

where

- $Z^{h^j} = \varphi^j\left(Z^{g^1},\dots,Z^{g^m},p^\circ_1,\dots,p^\circ_r\right) \quad (j\in[k])$.
- $(Z^{g^1},\dots,Z^{g^m})$ is statistically independent of $(p^\circ_1,\dots,p^\circ_r)$.
- $(Z^{g^1},\dots,Z^{g^m})$ is defined as in Yang (2019).
- $(p^\circ_1,\dots,p^\circ_r)$ is Gaussian with $\mathbb{E}(p^\circ_i)=0$ and $\mathrm{Cov}(p^\circ_k,p^\circ_i) = E\left[Z^{g^{i,1}}Z^{g^{i,2}}Z^{g^{k,1}}Z^{g^{k,2}}\right]$, where $\{Z^{g^{i,j}}\}_{i\in[r],j\in[2]}$ is defined as in Yang (2019).

As a corollary, we have $\quad \left(h^1_\alpha,\dots,h^k_\alpha\right) \xrightarrow{d} \left(Z^{h^1},\dots,Z^{h^k}\right) \quad (n\to\infty)$.

References

- Lee et al. (2017). Deep Neural Networks as Gaussian Processes.
- Matthews et al. (2018). Gaussian Process Behaviour in Wide Deep Neural Networks.
- Yang (2019). Tensor Programs I: Wide Feedforward or Recurrent Neural Networks of Any Architecture are Gaussian Processes.
- Hron et al. (2020). Infinite attention: NNGP and NTK for deep attention networks.