GENERATIVE MODELLING FOR CASCADING EXTREMES VIA AUTOREGRESSIVE TRANSFORMER

Johnny Myung Won Lee, Miguel de Carvalho, Jordan Richards, Sotirios Sabanis

School of Mathematics, University of Edinburgh

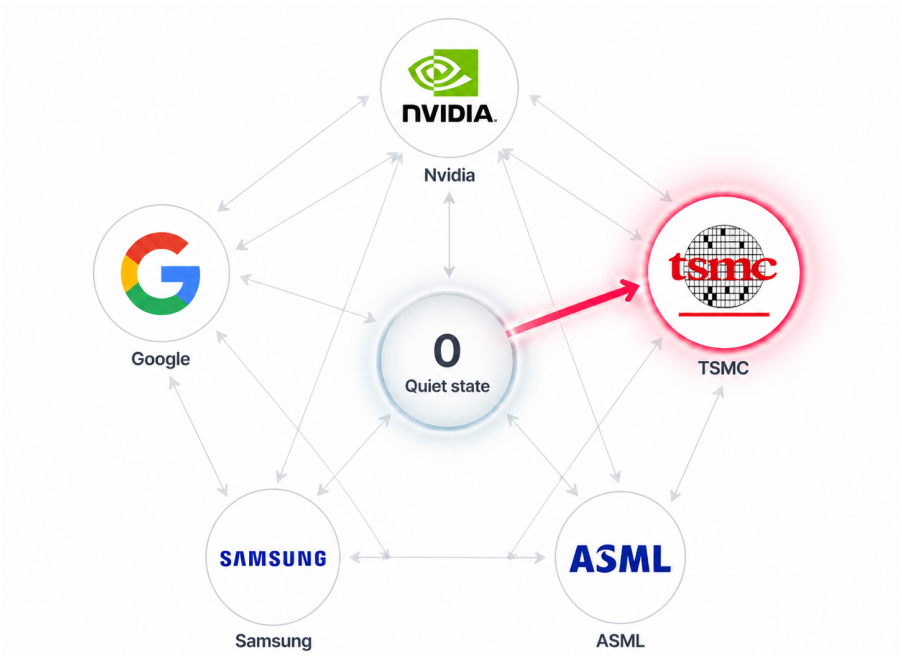
12th June 2026

THE UNIVERSITY *of* EDINBURGH
School of Mathematics

una europa

Cascading Extremes

in the context of finance



Motivation: Cascading Extreme Events

market crash, supply-chain disruption, natural disasters, power-grid failures.

Motivation: Cascading Extreme Events

market crash, supply-chain disruption, natural disasters, power-grid failures.

Extreme events often appear as **sequences of dependent shocks**, rather than isolated observations.

Shock in one component $\Longrightarrow$ increased risk of future extremes elsewhere.

Motivation: Cascading Extreme Events

market crash, supply-chain disruption, natural disasters, power-grid failures.

Extreme events often appear as **sequences of dependent shocks**, rather than isolated observations.

Shock in one component $\Longrightarrow$ increased risk of future extremes elsewhere.

Key question. Given the recent history of extremes, can we predict the next likely extreme event?

$$P\left(A_t \mid A_1, \ldots, A_{t-1}\right).$$

This motivates a generative, history-dependent model for cascading extremes (de Carvalho et al., 2026).

Cascading Extremes

in the context of transformer

- Let $\{\boldsymbol{Y}_t\}_{t=1}^{N}$ be a d-dimensional multivariate stochastic process with $\boldsymbol{Y}_t = (Y_{t,1}, \ldots, Y_{t,d})^{\mathrm{T}} \in \mathbb{R}^d$ at time $t = 1, \ldots, N$.
- Given a large threshold vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_d)^{\mathrm{T}} \in \mathbb{R}^d$, the system is considered to be in an extreme state if a variable $Y_{t,j}$ exceeds their respective threshold, μ_j at time t.
- We define the extreme indicator as $\boldsymbol{I}_t \in \{0, 1\}^d$ where $I_{t,j} = \mathbb{I}(\boldsymbol{Y}_{t,j} > \mu_j)$ for $j = 1, \ldots, d$.

Cascading Extremes

in the context of transformer

- Let $\{\boldsymbol{Y}_t\}_{t=1}^{N}$ be a d-dimensional multivariate stochastic process with $\boldsymbol{Y}_t = (Y_{t,1}, \ldots, Y_{t,d})^{\mathrm{T}} \in \mathbb{R}^d$ at time $t = 1, \ldots, N$.
- Given a large threshold vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_d)^{\mathrm{T}} \in \mathbb{R}^d$, the system is considered to be in an extreme state if a variable $Y_{t,j}$ exceeds their respective threshold, μ_j at time t.
- We define the extreme indicator as $\boldsymbol{I}_t \in \{0, 1\}^d$ where $I_{t,j} = \mathbb{I}(\boldsymbol{Y}_{t,j} > \mu_j)$ for $j = 1, \ldots, d$.

Definition 1 (Extremal Vocabulary and Token)

The extremal vocabulary, $\mathcal{V}$ as the finite set

$$\mathcal{V} = \{0, 1, \ldots, d\},$$

where 0 represents the quiet event (no exceedances). Let $A_t \in \mathcal{V}$ denote extremal token at time t:

$$A_t = \sum_{j=1}^{d} j \cdot I_{t,j}.$$

Cascading Extremes

in the context of transformer

Definition 2 (Extremal Sentence)

We define an extremal sentence (representing a single, isolated cascading sequence) as a finite sub-sequence of non-zero tokens bounded by the quiet state. Then the ψ-th cascading sequence $\mathcal{S}^{(\psi)}$ of length N_ψ is formally defined as:

$$\mathcal{S}^{(\psi)}_{t:t+N_\psi} = \{A_t, \ldots, A_{t+N_\psi}\},$$

where $A_\tau > 0$ for $\tau \in [t, t + N_\psi - 1]$ are the active extremal token and $A_{t+N_\psi} = 0$ is the end-of-sentence (EoS) token.

Cascading Extremes

in the context of transformer

Definition 2 (Extremal Sentence)

We define an extremal sentence (representing a single, isolated cascading sequence) as a finite sub-sequence of non-zero tokens bounded by the quiet state. Then the ψ-th cascading sequence $\mathcal{S}^{(\psi)}$ of length N_ψ is formally defined as:

$$\mathcal{S}^{(\psi)}_{t:t+N_\psi} = \{A_t, \ldots, A_{t+N_\psi}\},$$

where $A_\tau > 0$ for $\tau \in [t, t+N_\psi - 1]$ are the active extremal token and $A_{t+N_\psi} = 0$ is the end-of-sentence (EoS) token.

Example 1 (Token Process (Aldous, 2013))

$$\mathbf{0} \to \mathbf{0} \to \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(1)}} \to \mathbf{0} \to \mathbf{2} \to \mathbf{0} \to \underbrace{\mathbf{1} \to \mathbf{3} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(2)}} \to \cdots$$

Autoregressive Modelling

- The joint conditional distribution of an extremal sentence has an autoregressive structure,

$$\pi(\mathcal{S}) = \pi(A_1, \ldots, A_N) = \prod_{\tau=1}^{N} \pi(A_\tau \mid A_1, \ldots, A_{\tau-1}).$$

- To model the autoregressive structure of the sequence, we employ decoder-only autoregressive transformer (Vaswani et al., 2017)

Autoregressive Transformer

Decoder-only Transformer setup

- Consider an embedded input sequence $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\top \in \mathbb{R}^{N \times D}$.
- The fundamental goal of the Transformer is to build a network that transforms $\boldsymbol{X}$:

$$\widetilde{\boldsymbol{X}} = \text{TransformerLayer}[\boldsymbol{X}]$$

- The cornerstone of this transformation is Attention (Bahdanau et al., 2015; Vaswani et al., 2017).

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Token Embedding & Positional Encoding

Preparation

- First, we represent the discrete extremal token A_t as a one-hot encoded vector $\boldsymbol{v}_t \in \{0,1\}^{|\mathcal{V}|}$.
- Then, we project these extremal tokens into a continuous, high-dimensional vector space:

$$\boldsymbol{e}_t = \boldsymbol{v}_t \boldsymbol{W}^{(e)}, \quad \text{where } \boldsymbol{W}^{(e)} \in \mathbb{R}^{|\mathcal{V}| \times D}.$$

Token Embedding & Positional Encoding

Preparation

- First, we represent the discrete extremal token A_t as a one-hot encoded vector $\boldsymbol{v}_t \in \{0,1\}^{|\mathcal{V}|}$.
- Then, we project these extremal tokens into a continuous, high-dimensional vector space:

$$\boldsymbol{e}_t = \boldsymbol{v}_t \boldsymbol{W}^{(e)}, \quad \text{where } \boldsymbol{W}^{(e)} \in \mathbb{R}^{|\mathcal{V}| \times D}.$$

- Cascading extremes are strictly ordinal requires preservation of this structural dependency.

Example 2 (Permutation Invariance)

$$\underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{Cascading direction of } \mathcal{S}^{(1)}} \quad \not\equiv \quad \underbrace{\mathbf{4} \to \mathbf{1} \to \mathbf{5} \to \mathbf{0}}_{\text{Cascading direction of } \mathcal{S}^{(2)}}$$

Token Embedding & Positional Encoding

Preparation

- First, we represent the discrete extremal token A_t as a one-hot encoded vector $\boldsymbol{v}_t \in \{0,1\}^{|\mathcal{V}|}$.
- Then, we project these extremal tokens into a continuous, high-dimensional vector space:

$$\boldsymbol{e}_t = \boldsymbol{v}_t \boldsymbol{W}^{(e)}, \quad \text{where } \boldsymbol{W}^{(e)} \in \mathbb{R}^{|\mathcal{V}| \times D}.$$

- Cascading extremes are strictly ordinal requires preservation of this structural dependency.

Example 2 (Permutation Invariance)

$$\underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{Cascading direction of } \mathcal{S}^{(1)}} \quad \not\equiv \quad \underbrace{\mathbf{4} \to \mathbf{1} \to \mathbf{5} \to \mathbf{0}}_{\text{Cascading direction of } \mathcal{S}^{(2)}}$$

- *We inject sinusoidal positional encodings $\boldsymbol{p}_t \in \mathbb{R}^D$ (Vaswani et al., 2017):*

$$\boldsymbol{x}_t = \boldsymbol{e}_t + \boldsymbol{p}_t,$$

where the j-th dimension of $\boldsymbol{p}_t$ for $j \in [0, D-1]$ is given by:

$$p_{t,j} = \begin{cases} \sin\left(t/10000^{j/D}\right), & \text{if } j \text{ is even;} \\ \cos\left(t/10000^{(j-1)/D}\right), & \text{if } j \text{ is odd.} \end{cases}$$

Autoregressive Transformer

Masked Multi-head Self-Attention (Michel et al., 2019)

Definition 3 (Self-Attention Head)

Let the input sequence $\boldsymbol{X}$ *be projected into*

$$\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^{(q)} \in \mathbb{R}^{N \times D_q}, \quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^{(k)} \in \mathbb{R}^{N \times D_k}, \quad \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^{(v)} \in \mathbb{R}^{N \times D_v}.$$

The matrices $\{\boldsymbol{W}^{(q)}, \boldsymbol{W}^{(k)}\} \in \mathbb{R}^{D \times D_k}$ *and* $\boldsymbol{W}^{(v)} \in \mathbb{R}^{D \times D_v}$ *are learnable projection weights. An attention head is then defined as:*

$$\boldsymbol{H}_h = \text{Attn}(\boldsymbol{Q}_h, \boldsymbol{K}_h, \boldsymbol{V}_h, \boldsymbol{M}) = \text{Softmax}\left(\frac{\boldsymbol{Q}_h \boldsymbol{K}_h^{\mathrm{T}}}{\sqrt{D/H}} + \boldsymbol{M}\right)\boldsymbol{V}_h, \quad \text{for } h = 1, \ldots, H.$$

$$\text{MHSA}(\boldsymbol{X}) = \boldsymbol{H}\boldsymbol{W}^{(o)} = \text{Concat}[\boldsymbol{H}_1, \ldots, \boldsymbol{H}_H]\boldsymbol{W}^{(o)}, \quad \textit{where } \boldsymbol{W}^{(o)} \in \mathbb{R}^{D \times D}.$$

Autoregressive Transformer

Casual Self-attention

- The masked attention (causal mask matrix), $\boldsymbol{M} \in \mathbb{R}^{N \times N}$ is defined as,

$$M_{i,j} = \begin{cases} 0, & \text{if, } i \geq j \text{ (past and present tokens are allowed)} \\ -\infty, & \text{if, } i < j \text{ (future tokens are masked)} \end{cases}$$

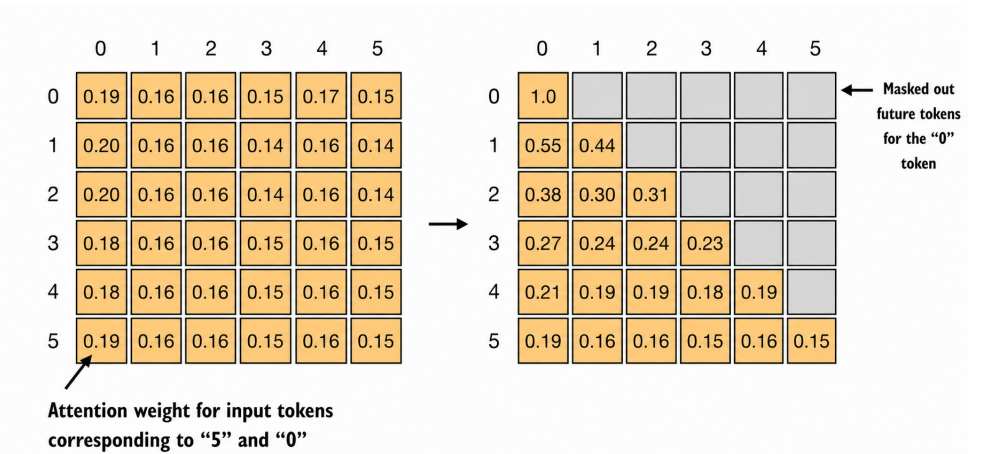


Figure. A concrete masked attention matrix and future-token entries are removed by the causal mask.

Autoregressive Transformer

Sub-layer 1: Masked Multi-head Self-Attention Layer (Michel et al., 2019)

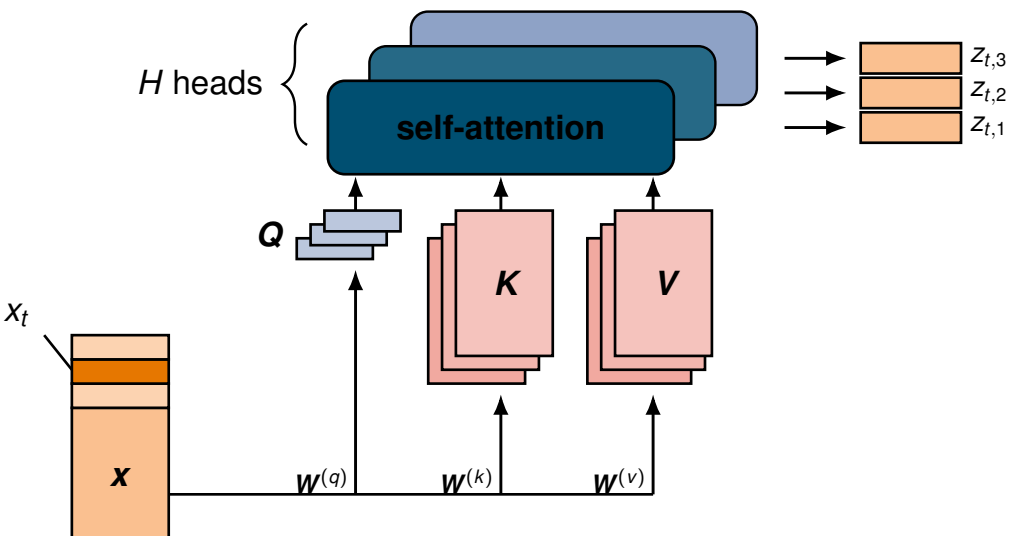


Figure. Schematic of multi-head self-attention applied to an embedded cascading sequence.

Autoregressive Transformer

Sub-layer 2: Transformer Layer

- We perform residual connections (He et al., 2016) and layer normalisation (Ba et al., 2016) to the output of MHA and MLP.

$$\boldsymbol{Z} = \text{LayerNorm}[\text{MHSA}(\boldsymbol{X}) + \boldsymbol{X}],$$
$$\widetilde{\boldsymbol{X}} = \text{LayerNorm}[\text{MLP}(\boldsymbol{Z}) + \boldsymbol{Z}].$$

Autoregressive Transformer

Sub-layer 2: Transformer Layer

- We perform residual connections (He et al., 2016) and layer normalisation (Ba et al., 2016) to the output of MHA and MLP.

$$\boldsymbol{Z} = \text{LayerNorm}[\text{MHSA}(\boldsymbol{X}) + \boldsymbol{X}],$$
$$\widetilde{\boldsymbol{X}} = \text{LayerNorm}[\text{MLP}(\boldsymbol{Z}) + \boldsymbol{Z}].$$

- Then, the temperature adjusted transition probability vector $\boldsymbol{P}_{t+1} \in [0,1]^{|\mathcal{V}|}$ is formally defined as:

$$\boldsymbol{P}_{t+1} = \text{Softmax}\left(\frac{\widetilde{\boldsymbol{X}}\boldsymbol{W}^{(p)}}{T}\right),$$

where T is the temperature parameter controlling the randomness of the generated token.

Numerical Experiments

p-th Order Markov Chain

- We simulate using a p-th order Markov chain over the vocabulary $\mathcal{V} = \{0, 1, 2, 3, 4, 5\}$.
- Let $h_t = (A_{t-p}, \ldots, A_{t-1})$ denote the recent p-step history of the d-dimension stochastic process.

$$A_t \mid h_t = h \sim \text{Categorical}\Big(\pi(h)\Big)$$

Numerical Experiments

p-th Order Markov Chain

- We simulate using a p-th order Markov chain over the vocabulary $\mathcal{V} = \{0, 1, 2, 3, 4, 5\}$.
- Let $h_t = (A_{t-p}, \ldots, A_{t-1})$ denote the recent p-step history of the d-dimension stochastic process.

$$A_t \mid h_t = h \sim \text{Categorical}\Big(\pi(h)\Big)$$

- The conditional transition distribution is defined as a combination of a static base transition probability π_{base} and a history-dependent transition probability $\pi_{\text{rules}}(h_t)$:

$$P(A_t \mid h_t) = (1 - \alpha)\pi_{\text{base}}(A_t) + \alpha\pi_{\text{rules}}(A_t \mid h_t)$$

- Here, $\alpha \in [0, 1]$ represents the transition strength parameter.

Numerical Experiments

Transition Dynamics

- The base transition probability $\pi_{\text{base}}(\cdot)$ is defined as,

Nodes	Quiet	NVIDIA	GOOGLE	TSMC	SAMSUNG	ASML
Tokens	0	1	2	3	4	5
Probability	0.7	0.08	0.02	0.1	0.05	0.05

Numerical Experiments

Transition Dynamics

- The base transition probability $\pi_{\text{base}}(\cdot)$ is defined as,

Nodes	Quiet	NVIDIA	GOOGLE	TSMC	SAMSUNG	ASML
Tokens	0	1	2	3	4	5
Probability	0.7	0.08	0.02	0.1	0.05	0.05

- The rule-based transition $\pi_{\text{rules}}(\cdot)$ dynamically shifts probability mass based on some behaviours:
  1. **Reversion to Quiet:** If $h_t = (0, \ldots, 0)$, probability mass shifts toward the quiet state $A_t = 0$.
  2. **First-Order Cascades:** Specific rule on heavily influence immediate successors (e.g., $A_{t-1} = 3$ significantly increases the likelihood of $A_t \in \{1, 4\}$).
  3. **Higher-Order Interactions:** The system explicitly models complex trajectories. For example:

$$(A_{t-3} = 5, A_{t-2} = 3, A_{t-1} = 4) \implies \text{Higher probability of } A_t = 5$$

$$\Rightarrow P(A_t = 5) + \delta, \quad \text{where } \delta \sim \text{Beta}(2, 6),$$

Numerical Results

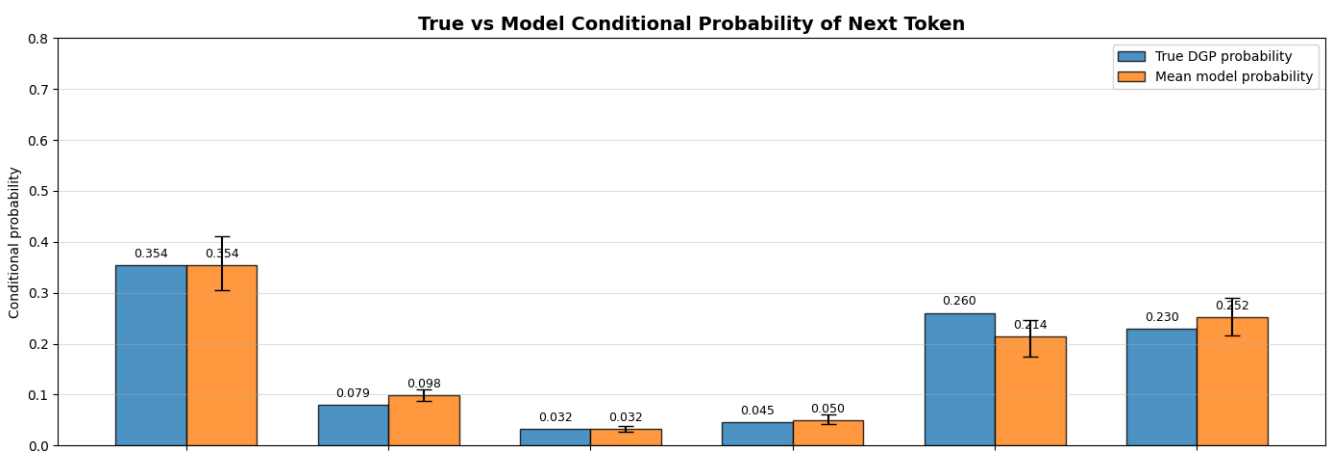
Conditional Next Token Probability

- We simulate 50 000 sequences of 3rd-order Markov chain
- $H = 4, D = 32, L = 2$

Numerical Results

Conditional Next Token Probability

- We simulate 50 000 sequences of 3rd-order Markov chain
- $H = 4, D = 32, L = 2$
- Using a random window of the testing sequence $h_t = (1, 5, \ldots, 2, 0, 3, 5, 1)$ with $T = 1$.



Numerical Results

$S_t = (1, 5, \ldots, 2, 0, 3, 5, 1)$

Token (j)	Asset	$P_{\text{true}}(A_t = j \mid h_t)$	$P_{\text{model}}(A_t = j \mid h_t)$	Error
0	Quiet (None)	0.3535	0.3536 (0.3056, 0.4104)	−0.0001
1	NVIDIA	0.0794	0.0982 (0.0876, 0.1109)	−0.0188
2	GOOGLE	0.0322	0.0325 (0.0273, 0.0388)	−0.0003
3	TSMC	0.0451	0.0499 (0.0414, 0.0608)	−0.0049
4	SAMSUNG	0.2602	0.2137 (0.1744, 0.2470)	0.0465
5	ASML	0.2297	0.2521 (0.2170, 0.2902)	−0.0224

Table. Comparison of the Monte Carlo mean of the conditional probabilities of next token against the true.

Amortisation

Goal. Adapt to many possible cascading mechanisms with different order.
Why useful? At test time, the model can predict under a new unseen DGP without re-estimating a full p-order transition table.

Amortisation

Goal. Adapt to many possible cascading mechanisms with different order.
Why useful? At test time, the model can predict under a new unseen DGP without re-estimating a full p-order transition table.

$$p^{(b)} \sim \text{DiscreteUniform}(1,3) \quad \text{for } 1,\dots,B.$$

For each batch b of size 64, we generate a sequence from

$$A_t^{(b)} \mid h_t^{(b)} \sim \text{Categorical}\Big(\pi^{(b)}(h_t^{(b)})\Big), \qquad h_t^{(b)} = (A_{t-p^{(b)}}^{(b)},\dots,A_{t-1}^{(b)}).$$

Amortisation

Goal. Adapt to many possible cascading mechanisms with different order.
Why useful? At test time, the model can predict under a new unseen DGP without re-estimating a full p-order transition table.

$$p^{(b)} \sim \text{DiscreteUniform}(1,3) \quad \text{for } 1,\dots,B.$$

For each batch b of size 64, we generate a sequence from

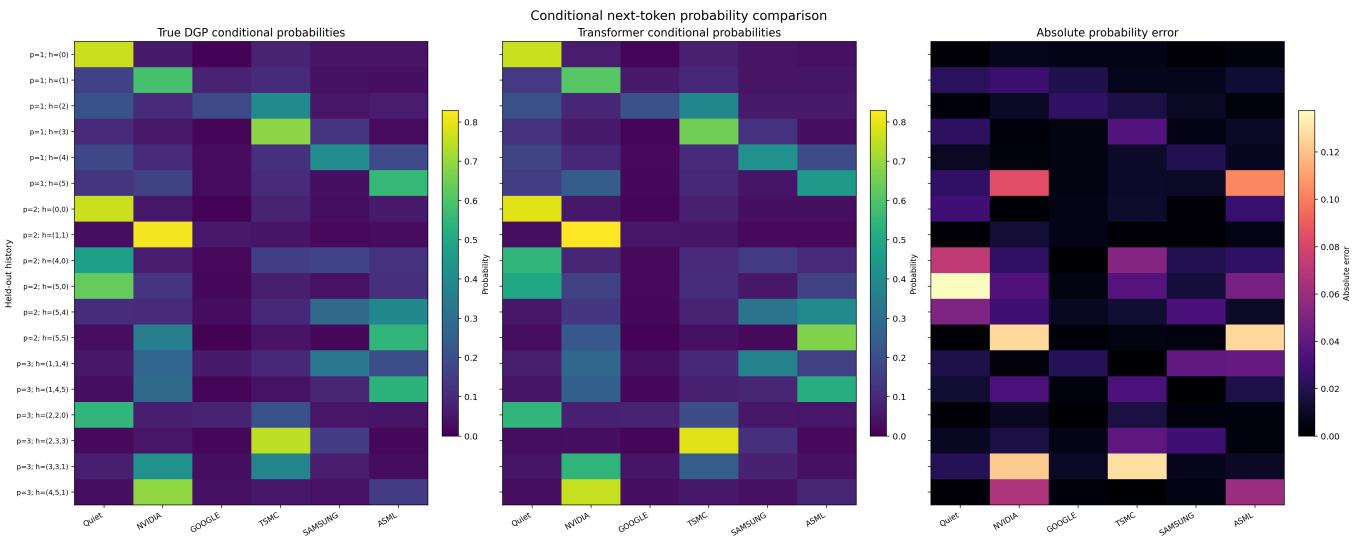
$$A_t^{(b)} \mid h_t^{(b)} \sim \text{Categorical}\Big(\pi^{(b)}(h_t^{(b)})\Big), \qquad h_t^{(b)} = (A_{t-p^{(b)}}^{(b)},\dots,A_{t-1}^{(b)}).$$

Amortised training procedure. The DGP is refreshed every 2 epochs, while the same model parameters θ ($\approx 100K$) continue to be updated.

$$\mathcal{L}_{\text{batch}}(\theta) = -\frac{1}{N}\sum_{b=1}^{B}\sum_{t}\ell_{\theta}\left(A_t^{(b)} \mid h_t^{(b)}\right).$$

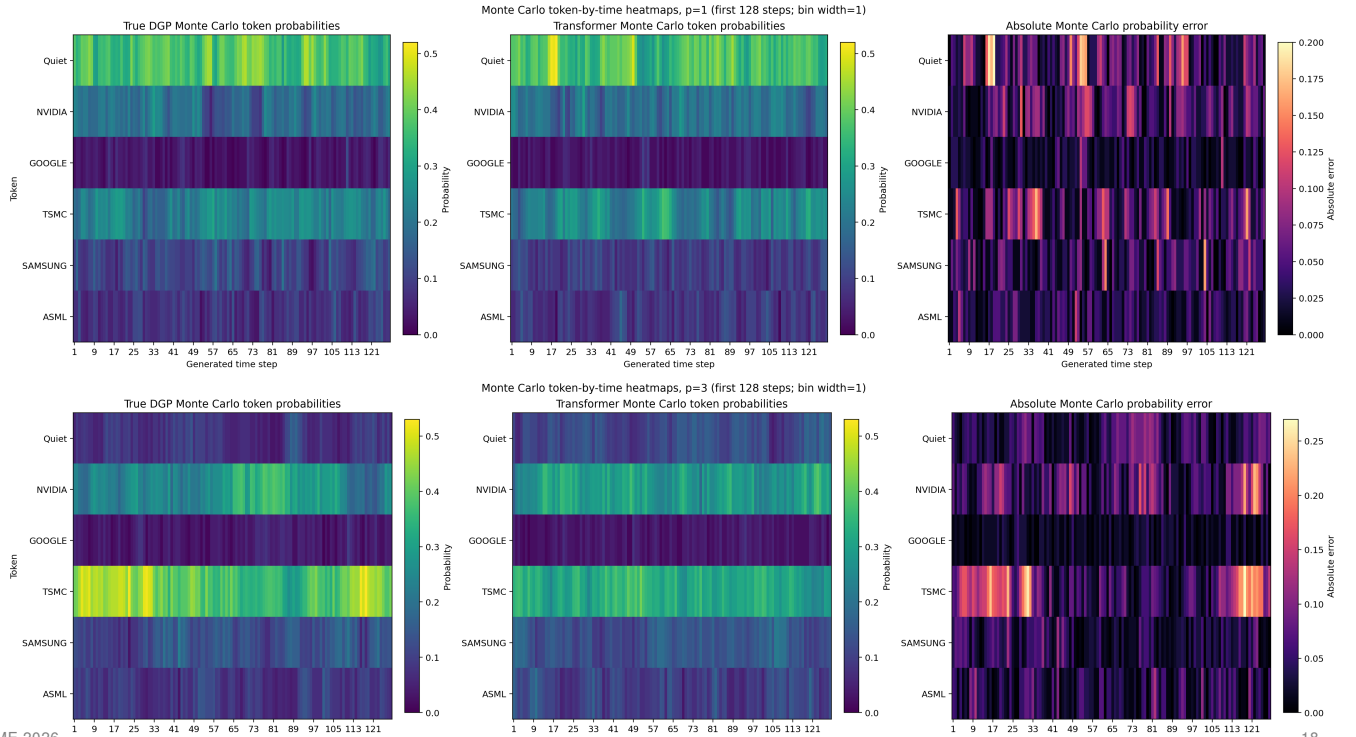
Amortisation

Conditional Next Token Probabilities



Amortisation

Conditional probability of each token across 200 Monte Carlo replicates.



Future direction

Application Intraday stock returns from entire stock market ($d \geq 500$)

Future direction

Application Intraday stock returns from entire stock market ($d \geq 500$)

Framework Move beyond current assumptions.

1. **Multiple cascading events** at t (Pandemic process (Aldous, 2013))

$$\mathbf{0} \to \mathbf{0} \to \left\{ \begin{array}{c} \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(1)}} \\ \underbrace{\mathbf{2} \to \mathbf{3} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(2)}} \end{array} \right. \to \mathbf{0} \to \mathbf{5} \to \mathbf{0} \to \left\{ \begin{array}{c} \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(3)}} \\ \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{1} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(3)}} \\ \underbrace{\mathbf{1} \to \mathbf{3} \to \mathbf{2} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(4)}} \end{array} \right. \to \cdots$$

2. **Delayed cascading events**

$$\mathbf{0} \to \underbrace{\mathbf{1} \to \mathbf{0} \to \mathbf{0} \to \mathbf{0} \to \mathbf{4} \to \mathbf{0} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(1)}} \to \mathbf{0} \to \mathbf{0} \to \underbrace{\mathbf{2} \to \mathbf{1} \to \mathbf{1} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(2)}} \to \cdots$$

Future direction

Application Intraday stock returns from entire stock market ($d \geq 500$)

Framework Move beyond current assumptions.

1. **Multiple cascading events** at t (Pandemic process (Aldous, 2013))

$$\mathbf{0} \to \mathbf{0} \to \left\{ \begin{array}{c} \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(1)}} \\ \underbrace{\mathbf{2} \to \mathbf{3} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(2)}} \end{array} \right. \to \mathbf{0} \to \mathbf{5} \to \mathbf{0} \to \left\{ \begin{array}{c} \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(3)}} \\ \underbrace{\mathbf{1} \to \mathbf{4} \to \mathbf{1} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(3)}} \\ \underbrace{\mathbf{1} \to \mathbf{3} \to \mathbf{2} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(4)}} \end{array} \right. \to \cdots$$

2. **Delayed cascading events**

$$\mathbf{0} \to \underbrace{\mathbf{1} \to \mathbf{0} \to \mathbf{0} \to \mathbf{0} \to \mathbf{4} \to \mathbf{0} \to \mathbf{5} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(1)}} \to \mathbf{0} \to \mathbf{0} \to \underbrace{\mathbf{2} \to \mathbf{1} \to \mathbf{1} \to \mathbf{0}}_{\text{cascading sequence } \mathcal{S}^{(2)}} \to \cdots$$

Inclusion of multiple inputs (e.g. magnitude of variables, temporal dependence)

References

- Aldous, D. (2013).**Interacting particle systems as stochastic social dynamics.** *Bernoulli*, *19*(4), 1122–1149.
- Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016).**Layer normalization.** *arXiv preprint arXiv:1607.06450*.
- Bahdanau, D., Cho, K., & Bengio, Y. (2015).**Neural machine translation by jointly learning to align and translate.** *International Conference on Learning Representations*.
- de Carvalho, M., Ferrer, C., & Vallejos, R. (2026).**A Kolmogorov–Arnold neural model for cascading extremes.** *Extremes*, 1–26.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016).**Deep residual learning for image recognition.** *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
- Michel, P., Levy, O., & Neubig, G. (2019).**Are sixteen heads really better than one?** *Advances in Neural Information Processing Systems*, *32*.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017).**Attention is all you need.** *Advances in Neural Information Processing Systems*, *30*.

Full Transformer Architecture

Algorithm Decoder-Only Transformer (Forward Pass)

Input:

Sequence of input token indices $\boldsymbol{X} \in \mathbb{R}^{N \times D}$

Attention weights for L layers and H heads: $(\boldsymbol{W}^{(q)}, \boldsymbol{W}^{(k)}, \boldsymbol{W}^{(v)})$.

Multi-layer Perceptrons Parameters for L layers: $(\boldsymbol{W}_1, \boldsymbol{W}_2)$ and $(\boldsymbol{b}_1, \boldsymbol{b}_2)$

Final output weight matrix $\boldsymbol{W}^{(p)} \in \mathbb{R}^{D \times |\mathcal{V}|}$

Output: Transition probability matrix $\boldsymbol{P} \in [0,1]^{N \times |\mathcal{V}|}$

$\boldsymbol{X}^{(0)} \leftarrow \text{TokenEmbed}(\boldsymbol{X}) + \text{PositionalEmbed}(\boldsymbol{X})$

for $l = 1, \ldots, L$ **do**

  $\boldsymbol{X}^{(l)'} \leftarrow \text{LayerNorm}\left[\boldsymbol{X}^{(l-1)}\right]$

  $\boldsymbol{Z}^{(l)} \leftarrow \text{LayerNorm}\left[\boldsymbol{X}^{(l)'} + \text{MHSA}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}, \boldsymbol{M})\right]$

  $\boldsymbol{X}^{(l)} \leftarrow \boldsymbol{Z}^{(l)} + \max\left(0,\ \boldsymbol{Z}^{(l)}\boldsymbol{W}_1^{(l)} + \boldsymbol{b}_1^{(l)}\right)\boldsymbol{W}_2^{(l)} + \boldsymbol{b}_2^{(l)}$

$\widetilde{\boldsymbol{X}} \leftarrow \text{LayerNorm}\left[\boldsymbol{X}^{(L)}\right]$

$\boldsymbol{P} \leftarrow \text{Softmax}(\widetilde{\boldsymbol{X}}\boldsymbol{W}^{(p)})$

return $\boldsymbol{P}$