SN Visibility for the Saturated RNN: Full Pattern Inventory Including Recurrence

Claude MJC

2026-02-07

Abstract

We extend the SN visibility infrastructure to include all three weight layers of a saturated RNN (0.079 bpc on 1024 bytes): input-to-hidden (W_{xh}), hidden-to-hidden (W_{hh}), and hidden-to-output (W_{hy}). The previous export (20260201) covered only W_{xh} and W_{hy} (302 patterns on the 5.69 bpc enwik9 model); the recurrent patterns in W_{hh} were completely unexplored. We export 2,999 total RNN patterns and 1,915 n-gram UM patterns in the same SN format, enabling direct visual comparison. Strength calibration quantifies the bottleneck effect: the ratio of RNN chain strength to UM pattern strength decreases monotonically from 0.73 (bigrams) to 0.31 (order 11), showing how much information is lost when n-gram statistics are embedded into 128 hidden neurons. This is the intermediate step toward reverse isomorphism—we now see what patterns exist in both representations, and the remaining problem is the embedding.

1 Introduction

The pattern-chains paper (20260206) showed that a saturated RNN on 1024 bytes has a flat chain depth distribution and that a direct n-gram UM matches it at order 11. MJC's commentary identifies the next step: *SN visibility*—make all patterns the RNN has learned visible in spectral notation, then compare with UM patterns found by direct counting.

The existing SN infrastructure (20260201 archive) exports only feedforward patterns: 49 input→hidden and 253 hidden→output, totaling 302 for the 5.69 bpc model on full enwik9. The recurrent layer (W_{hh}, $128 \times 128 = 16{,}384$ weights) contains the model's temporal memory, but these patterns were never exported. This paper fills that gap for the saturated model.

We also translate the n-gram UM into the same SN format. For bigrams, the translation is direct: input event → output event with strength $\lfloor \log_2(\text{count}) + 0.5 \rfloor$. For longer n-grams, we introduce context events that parallel the RNN's hidden layer: where the RNN uses h_j to carry state, the UM uses "ctx:ab" events encoding the conditioning string.

With both representations in SN, we can compare them side by side in the viewer and calibrate the strength relationship.

2 Full Pattern Inventory

2.1 Export Method

We load the saturated model (`sat_model.bin`, 329 KB) and export every weight with $\lfloor 2|w| + 0.5 \rfloor \geq 1$ as an SN pattern. The sign convention for W_{hh}: entry $W_{hh}[j][k]$ means "from neuron k at time t to neuron j at time $t+1$." Under doubled-E, source is always h_k^+ (positive activations propagate); positive weight gives $h_k^+ \to h_j^+$; negative weight gives $h_k^+ \to h_j^-$.

2.2 Pattern Counts

Layer	Total weights	Patterns ($s \geq 1$)	Max strength
W_{xh} (input→hidden)	32,768	522	4
W_{hh} (hidden→hidden)	16,384	1,006	3
W_{hy} (hidden→output)	32,768	1,471	8
Total	81,920	2,999	

Table 1: Pattern counts per layer for the saturated model. The 5.69 bpc enwik9 model had 302 feedforward patterns (49 + 253). The saturated model has 10× more patterns overall, reflecting memorization of the 1024-byte dataset. The W_{hh} layer contributes 1,006 patterns—a third of the total.

2.3 Strength Distribution

Strength	W_{xh}	W_{hh}	W_{hy}
1	435	964	1,249
2	71	38	169
3	14	4	35
4	2	—	11
5	—	—	5
7	—	—	1
8	—	—	1

Table 2: Strength distribution histogram. W_{hh} patterns overwhelmingly have strength 1 (96% of 1,006), consistent with the moderate weight-count correlations ($r = 0.46$) reported in saturation.pdf. W_{hy} has the strongest patterns (up to 8), reflecting sharp output predictions for specific contexts.

The maximum possible strength from 1024 bytes is $\log_2(1024) = 10$. The strongest observed pattern (strength 8, in W_{hy}) approaches this limit, indicating near-certainty for some output predictions. The W_{hh} maximum of 3 means no single recurrent connection carries more than $\approx 2^3 = 8\times$ evidence—the memory is distributed across multiple neurons and chains.

3 Recurrent Pattern Structure

The W_{hh} patterns form a directed graph on 128 neurons (256 doubled-E events). Each pattern $h_k^+ \to h_j^+$ (or $h_k^+ \to h_j^-$) represents one step of temporal state propagation.

3.1 Hub Neurons

The average neuron has ∼8 connections (fan-in + fan-out), but the distribution is highly uneven: the top 10 hubs account for a disproportionate share of recurrent routing.

Neuron	Fan-in	Fan-out	Total
h100	36	36	72
h91	25	39	64
h69	27	28	55
h53	25	29	54
h54	22	29	51
h113	19	32	51
h118	30	19	49
h4	21	27	48
h107	20	27	47
h16	22	25	47

Table 3: Top 10 hub neurons by total connections (fan-in + fan-out) in the W_{hh} graph. h100 is the dominant hub with 72 connections (56% of all neurons). Compare with the chain analysis (pattern-chains.pdf) where h15, h8, h112 were the top chain participants—the hub structure and chain routing highlight different aspects of the same recurrence.

3.2 Self-Connections

The diagonal of W_{hh} encodes each neuron's temporal persistence. Twenty neurons have self-connections with $|w| \geq 0.25$:

Neuron	Weight	Role	Neuron	Weight	Role
h9	−1.252	flip	h57	+0.685	persist
h16	+0.670	persist	h73	+0.631	persist
h21	−0.524	flip	h113	+0.516	persist
h118	−0.503	flip	h62	+0.431	persist
h4	−0.313	flip	h27	+0.472	persist

Table 4: Strongest self-connections. "Persist" neurons maintain their sign across timesteps; "flip" neurons tend to oscillate. h9 has the strongest self-connection ($w = -1.252$, strength 3), a strong oscillator. Compare h35: in the 5.69 bpc model its self-connection was -0.184 (word-length decay); here it is $+0.288$ (weak persistence), reflecting the different data regime (XML vs English text).

3.3 Relation to Chain Depth

The flat chain depth distribution from pattern-chains.pdf (depths 0–8 each ∼10–14% of chains) can now be understood structurally. The W_{hh} graph is densely connected: the median neuron has 8 connections, and no neuron is isolated (every neuron participates in at least one recurrent pattern). This density ensures that chains of any available depth can find strong links at each step. A sparser W_{hh} graph would show depth-dependent drop-off as chains hit dead ends.

4 N-gram UM Patterns in SN

4.1 Translation

For each n-gram of length ℓ with count $c \geq 2$:

- **Bigrams** ($\ell = 2$): Directly “The input is ‘a’.” → “The output is ‘b’.” with strength $\lfloor \log_2(c) + 0.5 \rfloor$.
- $\ell \geq 3$: We introduce context events of the form “ctx:s” where s is the conditioning string. The context event parallels the RNN’s hidden state: where the RNN uses h_j to carry temporal information, the UM uses an explicit context event encoding the conditioning string. For a trigram abc: “ctx:a” → “The output is ‘c’.” with strength $\lfloor \log_2(c) + 0.5 \rfloor$. Context-building patterns connect shorter contexts to longer ones.

4.2 Pattern Counts

Order	Prediction patterns	Context events
2 (bigram)	124	—
3	131	
4	125	
5	119	
6	106	
7	98	
8	91	
9	84	
10	78	
11	73	
Total predictions	1,029	886 context events
+ context-building	+886	
Grand total	1,915	

Table 5: UM pattern counts by n-gram order. Only n-grams with count ≥ 2 are included. The 886 context events parallel the RNN’s 256 hidden events; where the RNN has a fixed-size state, the UM has as many context events as there are distinct conditioning strings in the data.

4.3 Side-by-Side Example

Consider the 4-gram “`wiki`” (count = 6 in the 1024-byte dataset):

UM representation (from n-gram counting):

```
"The input is ’w’." -> "ctx:w"      strength=5 (from w-count)
"ctx:w"             -> "ctx:wi"     strength=4 (from wi-count)
"ctx:wi"            -> "ctx:wik"    strength=4 (from wik-count)
"ctx:wik"           -> out(’i’)     strength=3 (floor(log2(6)+0.5))
```

RNN chain (traced through hidden activations):

```
in(’w’) --[Wx,s=1]--> h_j+ --[Wh,s=1]--> h_k+
        --[Wh,s=1]--> h_m+ --[Wy,s=?]--> out(’i’)
Chain strength = min(links) = 1.25
```

The UM strength (3) exceeds the RNN chain strength (1.25), ratio 0.48—the W_{hh} bottleneck compresses the evidence. The UM representation is transparent: each pattern maps directly to a counted n-gram. The RNN representation is opaque until decoded via the isomorphism.

5 Strength Calibration

5.1 Method

For each n-gram of length $\ell = 2, \dots, 11$ with count $c \geq 2$:

1. UM strength: $s_{\text{UM}} = \log_2(c)$
2. Find the best matching RNN chain (trace the n-gram through actual hidden activations)
3. RNN chain strength: $s_{\text{RNN}} = \min_i(2|w_i|)$ over all links
4. Compute ratio: $s_{\text{RNN}}/s_{\text{UM}}$

5.2 Results

Length	n-grams	Matched	Avg s_{UM}	Avg s_{RNN}	Avg ratio
2	124	124 (100%)	2.27	1.37	0.73
3	131	131 (100%)	2.03	1.10	0.67
4	125	125 (100%)	2.02	0.88	0.55
5	119	118 (99%)	2.01	0.83	0.51
6	106	104 (98%)	2.09	0.76	0.45
7	98	98 (100%)	2.12	0.70	0.41
8	91	91 (100%)	2.14	0.67	0.40
9	84	82 (98%)	2.18	0.63	0.35
10	78	77 (99%)	2.22	0.59	0.32
11	73	72 (99%)	2.24	0.57	0.31
Total: 1,029		1,022 (99.3%)		Overall avg ratio: 0.49	

Table 6: Strength calibration: UM vs RNN chain strength by n-gram length. The ratio decreases monotonically from 0.73 (bigrams) to 0.31 (order 11), quantifying the bottleneck effect. 99.3% of n-grams with count ≥ 2 have traceable RNN chains.

5.3 The Bottleneck Effect

The calibration reveals a clear pattern:

1. **Bigrams** ($\ell = 2$): RNN preserves 73% of UM strength. Only two links (input→hidden, hidden→output), so the bottleneck is just the weight quantization in the hidden layer.
2. **Mid-range** ($\ell = 4$–6): Ratio drops to 0.45–0.55. Each additional recurrent step introduces another potential weak link, and the min-over-links chain strength amplifies this.
3. **Long-range** ($\ell = 9$–11): Ratio reaches 0.31–0.35. The chain passes through 7–9 W_{hh} links, each of which has maximum strength 3 but typical strength 1. The min over many weak links converges toward the weakest.

The monotonic decrease confirms that the hidden layer is a *bottleneck* for longer patterns: the same 128 neurons must serve all chain depths, and each additional depth multiplies the chance of a weak link.

5.4 Connection to $Q = \lambda$

The ratio $s_{\text{RNN}}/s_{\text{UM}}$ measures how much of the UM's optimal compression is preserved through the hidden embedding. In the quotient framework, the UM achieves luck λ for each pattern; the RNN achieves λ^r where r is the ratio.

At overall average ratio 0.49, the RNN preserves about half the log-evidence of each pattern. In the limit of infinite hidden size, $r \to 1$ and the RNN becomes the UM. For our 128-neuron model on 1024 bytes, the bottleneck costs ∼0.5× the available evidence per pattern—but the RNN still achieves 0.079 bpc because it has enough capacity to represent the patterns that matter most.

6 Implications

6.1 Toward Reverse Isomorphism

We now have:

1. Complete visibility into what the RNN has learned (2,999 patterns across all layers)
2. The same patterns found independently by the UM (1,029 n-gram prediction patterns, 886 context-building patterns)
3. A quantitative measure of the bottleneck (ratio 0.73 at bigram level down to 0.31 at order 11)

The remaining problem for reverse isomorphism is the *embedding*: given n-gram UM patterns with known strengths, find W_{xh}, W_{hh}, W_{hy} that realize those patterns. The SVD approach (20260131_4) is one method; the calibration data informs what strength ratios to target. The key constraint: the W_{hh} graph must be dense enough to sustain chains at all depths while preserving at least 30% of the UM strength at the longest chains.

6.2 Scaling Implications

On 1024 bytes, the bottleneck is modest—128 neurons suffice for ∼order-11 patterns with 99.3% coverage. As dataset size grows, two pressures emerge:

- More distinct patterns to embed (vocabulary grows sublinearly but patterns grow combinatorially)
- Longer-range patterns needed (sentence structure, discourse)

The UM can always be extended by counting; the question is whether RNN capacity scales to embed the patterns. The 0.31 ratio at order 11 suggests that even on this small dataset, the hidden layer is near its capacity for long-range patterns.

7 Reproducibility

Repository. `https://github.com/inimino/hutter` (commit: `8ce5e5f`).

Model. Elman RNN, 256 → 128 → 256, tanh activation. Trained with SGD, cosine LR decay (0.01 → 0), gradient clipping at 5, BPTT length 50, 4,000 epochs on first 1024 bytes of enwik9. Checkpoint: `sat_model.bin` (329 KB), symlinked from 20260206 archive. Downloadable: `https://cmpr.ai/hutter/archive/20260206/sat_model.bin`.

Data. First 1024 bytes of enwik9 (`http://mattmahoney.net/dc/enwik9.zip`).

Source files. `sat_sn_full.c` (full SN export), `sat_um_sn.c` (n-gram UM to SN), `sat_calibrate.c` (strength calibration), `reproduce.sh` (all-in-one).

To reproduce:

```
git clone https://github.com/inimino/hutter.git
cd hutter
git checkout [TBD]
bash reproduce.sh
```