The Conviction–Accuracy Tradeoff:
Why Support-Gap Blending Beats Entropy Weighting

Universal Model Project

March 2026

Abstract

Gap-weighted blending ($w_k = 2^{s_1 - s_2}$ per order k) beats entropy-weighted blending ($w_k = \exp(-H_k)$) by 0.093–0.174 bpc on order-6 and order-8 n-gram UM models, despite entropy being a strictly better *selector* of the oracle-best order (62% vs 32% accuracy at order 6; 63% vs 23% at order 8). We call this the conviction–accuracy tradeoff: gap-blend wins not by picking the right order more often, but by assigning exponentially more weight when it is right. This note presents the oracle-correlation evidence and explains the mechanism.

1 Setup

At each position t we have per-order distributions $p_k(\cdot)$ for $k = 0, 1, \ldots, K$. Define:

- **Support gap**: $g_k = s_1^{(k)} - s_2^{(k)}$, the difference between the top two log-probability supports in order k.
- **Entropy**: $H_k = -\sum_b p_k(b) \log_2 p_k(b)$.
- **Oracle-best order**: $k^* = \arg\max_k p_k(b_t)$, the order that assigns highest probability to the actual next byte b_t.

Gap-selection picks $\arg\max_k g_k$. **Entropy-selection** picks $\arg\min_k H_k$. Accuracy = fraction of positions where the selection matches k^*.

2 Results

2.1 Order 6, 1M bytes

Metric	Gap	Entropy
Selection accuracy	32.3%	62.0%
Mean oracle surprise (selected)	3.21 bpc	4.80 bpc

Per-order breakdown:

Order	Gap picks	Ent picks	Gap acc	Ent acc
0	103,250	306	9.9%	23.2%
1	147,917	13,530	11.2%	34.2%
2	198,756	77,508	24.3%	56.8%
3	201,466	160,348	32.9%	63.7%
4	173,193	245,989	48.2%	69.0%
5	108,537	252,999	55.4%	64.8%
6	66,685	249,124	56.6%	54.2%

2.2 Order 8, 1M bytes

Metric	Gap	Entropy
Selection accuracy	23.4%	62.6%
Mean oracle surprise (selected)	3.71 bpc	5.42 bpc

Per-order breakdown:

Order	Gap picks	Ent picks	Gap acc	Ent acc
0	408,297	6,454	6.1%	8.1%
1	90,993	12,974	9.7%	35.2%
2	93,339	61,447	28.6%	63.5%
3	122,310	140,995	35.2%	66.4%
4	119,209	214,884	47.8%	71.0%
5	79,668	205,558	48.9%	67.7%
6	43,703	146,569	40.7%	60.3%
7	25,155	109,668	35.8%	53.9%
8	17,130	101,255	46.7%	48.4%

3 The Conviction Mechanism

The paradox: entropy selects the oracle-best order nearly twice as often as gap, yet gap-blend beats ent-blend by 0.093 bpc (order 6) and 0.085 bpc (order 8). The resolution lies in *how much* weight each method assigns when it is right versus wrong.

3.1 Gap concentrates weight exponentially

Gap weights are $w_k = 2^{g_k}$. When one order has a decisive gap ($g = 8$), its weight is $2^8 = 256$, dominating all other orders. When no order is confident ($g \approx 0$ for all), weights are roughly uniform—the blend degrades gracefully to a flat average.

Entropy weights are $w_k = \exp(-H_k)$. Even a "concentrated" distribution with $H = 1$ bit gives $w = 0.37$, while a diffuse one with $H = 4$ gives $w = 0.018$. The ratio is only 20:1. Entropy cannot express the extreme conviction ratios that gap achieves.

3.2 Gap's failure mode is benign

When gap picks order 0 (41% of the time at order 8), the unigram distribution has high support everywhere. Its gap is large because the top-2 supports are well-separated (the unigram has a strong mode at space/e/t/a). But the damage of over-weighting unigram is bounded: its predictions are never catastrophically wrong.

When entropy picks the wrong order, the damage can be large: a low-entropy distribution that is *confidently wrong* about the next byte incurs a large surprise penalty.

3.3 Why the tradeoff matters for blending

For *selection* (winner-take-all), accuracy is all that matters. For *blending* (weighted combination), what matters is:

$$\text{bpc} = \sum_t \sum_k \frac{w_k}{\sum_j w_j} \left(-\log_2 p_k(b_t)\right)$$

High-conviction correct weights reduce bpc quadratically (the correct order dominates the sum), while high-conviction incorrect weights increase bpc linearly (the incorrect order is diluted by the correct ones if any have moderate weight). This asymmetry favors the high-conviction strategy: occasional extreme wins outweigh frequent moderate losses.

4 Sample Efficiency

The conviction mechanism is not only a high-data effect. At order 6:

Policy	100K	1M
KN-interp	2.741	2.418
Ent-blend	2.799	2.337
Gap-blend	2.559	2.244

At 100K, entropy-weighted blending is still worse than KN, while gap-blend already wins. By 1M, entropy-blend catches up and overtakes KN, but gap-blend remains best. This matters because it shows the support gap is not merely a late-data refinement. It is already a useful confidence signal when counts are still sparse.

One way to read this is: entropy needs enough data for distributional sharpness to become trustworthy, while gap can exploit early winner-margin information without waiting for the whole distribution to stabilize.

5 Implications

1. **Gap-blend is the right default** for UM n-gram models. It beats KN by 0.174 bpc (order 6) and ent-blend by 0.093 bpc, while being a native tropical operation (2^g is just a shift).

2. **The combination problem has a tropical solution.** Max-min (flat accumulation) costs 2.9 bpc. But the tropical semiring can express confidence-weighted combination via the gap measure. The "price of tropical" drops from 2.9 to <0.1 bpc with the right combination rule.

3. **Conviction beats accuracy** in distribution blending. This is a general principle: when combining multiple probabilistic predictors, it is better to weight by *how committed* each predictor is (support gap) than by *how concentrated* its distribution is (entropy). Commitment to a specific outcome is a better proxy for "I know the answer" than general sharpness.

4. **The oracle accuracy asymmetry grows with order.** At order 6, gap selects correctly 32% vs entropy's 62%. At order 8, it drops to 23% vs 63%. Yet gap-blend's advantage *also grows* with order (from 0.093 to 0.085 bpc, roughly stable). As more orders compete, gap's extreme weighting becomes more valuable because it cuts through the noise.

6 Conviction Depth: Per-Position Evidence

The oracle correlation shows entropy selects better. The conviction depth experiment directly measures the per-position surprise difference between gap-blend and ent-blend.

6.1 Order 6, 1M bytes

	Gap wins	Ent wins	Ties
Positions	362,026 (36.2%)	632,969 (63.3%)	5,004
Mean advantage	1.053 bpc	0.456 bpc	—
Total bits saved	381,379	288,623	—

Net: gap-blend ahead by 92,756 bits = 0.093 bpc.

Gap wins fewer positions but wins **2.3× larger** on average. The surprise-difference histogram is strongly right-skewed: ent-blend wins cluster in $[-0.5, 0)$ bpc (small wins) while gap-blend wins have a long tail out to +7.5 bpc.

6.2 Order 8, 1M bytes

	Gap wins	Ent wins	Ties
Positions	314,743 (31.5%)	678,995 (67.9%)	6,261
Mean advantage	1.484 bpc	0.563 bpc	—
Total bits saved	467,061	382,386	—

Net: gap-blend ahead by 84,675 bits = 0.085 bpc.

At order 8, gap's mean winning advantage grows to **1.484 bpc** (up from 1.053 at order 6), while ent's stays moderate at 0.563. Gap dominantly selects order 0 for 46.5% of its winning positions.

6.3 The asymmetry principle

For blending, what matters is not how often you pick the right predictor, but the *product* of frequency and magnitude. Let f_g, f_e be gap/ent win frequencies and m_g, m_e be mean advantages. The net is $f_g \cdot m_g - f_e \cdot m_e$:

- Order 6: $0.362 \times 1.053 - 0.633 \times 0.456 = 0.381 - 0.289 = +0.092$
- Order 8: $0.315 \times 1.484 - 0.679 \times 0.563 = 0.467 - 0.382 = +0.085$

Both match the observed bpc difference, confirming the mechanism is pure frequency×magnitude asymmetry.

7 References

two complementary evidence notes are:

- *The Conviction–Accuracy Tradeoff: Why Support-Gap Blending Beats Entropy Weighting*, which shows why gap can beat entropy despite worse oracle-order selection;
- *Conviction Depth: Fewer Wins, Bigger Wins*, which shows how that advantage is distributed across positions.

8 Reproduce

```
./umr oracle-corr enwik9 6 1000000
./umr oracle-corr enwik9 8 1000000
./umr ngram-ablation enwik9 6 1000000
./umr ngram-ablation enwik9 8 1000000
```

References

[1] Graf. *Conviction Depth: Fewer Wins, Bigger Wins.* Hutter archive, 12 March 2026.

[2] Vanguard, Graf, and Vector. *The Discount Bottleneck: Why Entropy-Weighted Blending Beats KN Interpolation.* Hutter archive, 12 March 2026.