Top-nσ: Not All Logits Are You Need

A logit-space sampling method for reasoning-oriented generation

Based on Chenxia Tang, Jianchun Liu, Hongli Xu, Liusheng Huang
University of Science and Technology of China •
arXiv:2411.07641v1 (Nov 2024)

Core claim: sampling can beat greedy decoding on reasoning tasks if we first separate informative logits from noisy logits.

How an LLM Calculates Logits

Steps performed before top-nσ

- Break down a user prompt into a list of tokens
- Calculate the hidden states by running the tokens through the transformer blocks
- Use the unembedding formula to map the final hidden state into logit values for all tokens in the vocabulary

$$\text{Logits} = h \bullet W_u + b$$

Context (tokens so far) › Model forward pass › **Logits for every vocabulary token** › Softmax with temperature T › Choose next token

How an LLM Chooses the Next Token

Steps performed after top-nσ

- Softmax converts logits into probabilities that sum to 1.
- Temperature rescales logits before Softmax: low T sharpens the distribution; high T flattens it.
- Decoding then selects one token from this distribution and appends it to the context.

$$p_i = \frac{\exp(l_i/T)}{\sum_j \exp(l_j/T)}$$

Common Decoding Methods & Temperatures Effect

Common decoding rules:

- **Greedy:** pick the single highest probability token.
- **Top-k:** keep only the k highest probability tokens, renormalize, then sample.
- **Top-p (nucleus):** keep the smallest set whose cumulative probability is at least p, renormalize, then sample.

Temperature's effect on distribution:

- Lower T makes the leader more dominant.
- Higher T spreads mass toward lower ranked tokens.
- That increases diversity, but can also let weak tokens survive truncation.

Main Limitations of Current Sampling Methods

Post Softmax truncation:

- Compute probabilities with temperature first.
- Then keep tokens via top-k or top-p.
- As temperature rises, the tail gets flatter, so more low-value tokens can become eligible.

Why the authors move to logit space:

Raw Logits: Strong separation between top logits and the noisy bulk may already be present.

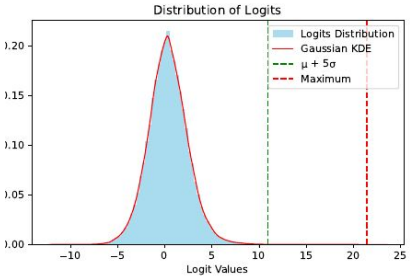
Softmax + T: Flattening can lift weak tokens enough to cross top-k / top-p cutoffs.

Papers Idea: Filter in logit space first, then use temperature only within the surviving set.

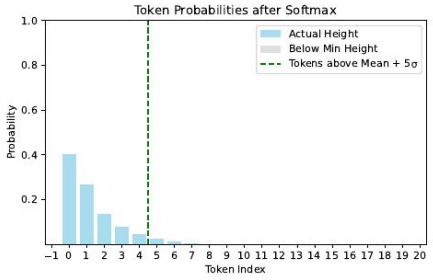
This is the central motivation of the paper.

Key empirical observation in logit space

The paper claims most logits form a Gaussian-like noisy bulk, while a few large logits carry most useful mass



(a) Distribution of logits



(b) Descendingly sorted Probabilities. Only the top 2(tokens are shown.

What Figure 1 suggests:

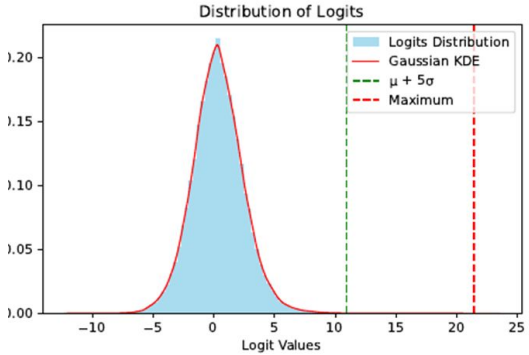
- The background logits look approximately Gaussian.
- The top-probability tokens live in a right-tail region far above the mean.
- The maximum logit can be roughly 10σ above the mean on the shown example.
- After softmax, that tiny tail dominates the actual probability mass.

Interpretation: the model already separates “mostly irrelevant” from “worth considering” before softmax.

Source: Tang et al. (2024), Fig. 1, p. 2

Figure 1(a): logit distribution before softmax

Clarify noisy region, informative region, and the main insight.



(a) Distribution of logits

re 1: Distribution of logits and descendingly so

Informative Region:

- The sparse right tail contains a few tokens far above the mean.
- Those are the plausible next-token candidates the model seems to “care about.”
- Main insight: useful candidates may already be separated from noise in raw logit space.

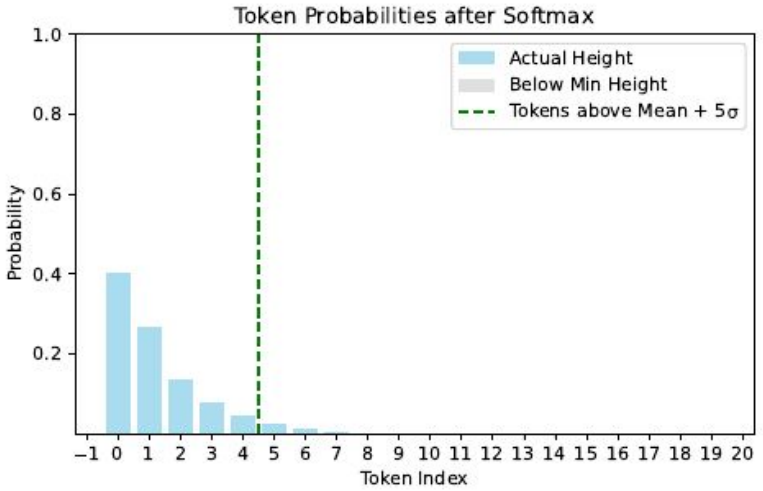
Noisy Region:

- The broad central hump around the mean represents the bulk of vocabulary tokens.
- These tokens are treated as mostly uninformative for this step: many have similar, middling logits.
- Because there are so many of them, any method that becomes too permissive can accidentally admit some of them.

Source: Tang et al. (2024), Fig. 1, p. 2

Probability Distribution after Softmax

Why the same step looks so different once logits are exponentiated and normalized.



What changes after Softmax?

- Exponentiation magnifies logit gaps: a small right-tail advantage can become a huge probability advantage.
- So the top few tokens dominate the normalized distribution; the long tail contributes almost nothing individually.
- This strengthens the paper's motivation: the informative set may be tiny, and letting many weak tokens enter it is unnecessary.

Source: Tang et al. (2024), Fig. 1, p. 2

The proposed method: top-nσ

Rule:

$$\text{keep token } i \text{ if } l'_i \geq M - n\sigma$$

- Scale logits by temperature first.
- Compute the maximum and standard deviation.
- Filter out tokens farther than nσ below the max.
- Apply softmax only to the surviving tokens, then sample.

How it differs:

- top-k keeps a fixed number of tokens.
- top-p keeps enough tokens to hit a probability mass target.
- min-p keeps tokens above a probability floor relative to the maximum.
- top-nσ keeps tokens based on their logit distance from the max.

Claimed advantages:

- Filters in logit space before low quality tokens gain mass through softmax.
- Uses cheap statistics: max and standard deviation.
- Separates candidate filtering from within set exploration.

From probabilities to logits

Nucleus-style selection can be written as a logit thresholding problem

Common decoding practice:

$$p_i = \frac{e^{l_i}}{s}, \quad \text{where } s = \sum_{j=1}^{V} e^{l_j}, 1 \le i \le V$$

$$\mathcal{N} = \{i \mid p_i \ge p^{(t)}\} = \{i \mid l_i \ge t\}$$

- A probability cutoff implicitly defines a matching logit cutoff.
- So we can study top-p-like behavior through the distribution of logits rather than normalized probabilities.

Why this matters:

- If logits follow a recognizable distribution f(x), the paper derives the nucleus mass analytically from integrals over e^x f(x).
- This creates a bridge from observed logit geometry to practical decoding rules.
- It also lets the authors reinterpret methods like top-p and min-p as threshold rules in logit space.

Source: Tang et al. (2024), Sec. 2.1

Interpreting the two regions

The paper gives a conceptual account of where “noise” comes from and why the informative region may behave differently

Noisy region (Gaussian-like bulk):

- The authors treat the bulk of the vocabulary as effectively noisy because these tokens have negligible utility but non-zero logits.
- Proposed sources: training data noise, regularization, and the “noise of silence” caused by softmax assigning finite mass to irrelevant tokens.
- At high temperature, this bulk becomes more dangerous because it is easier for noise tokens to enter the sampling pool.

Informative region (small right tail):

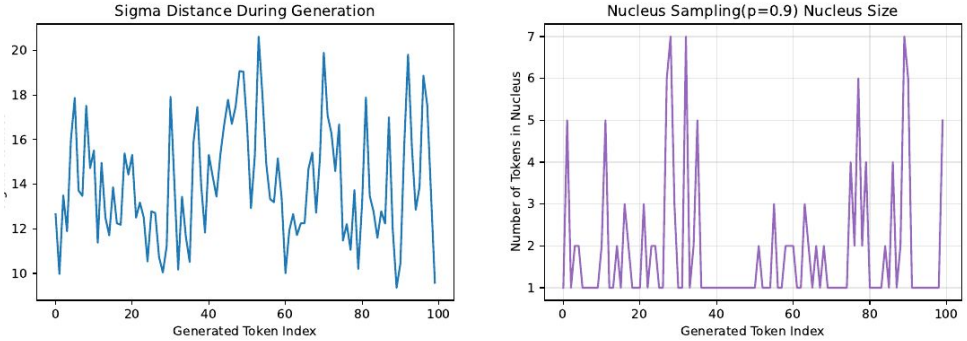
- A small number of tokens account for most probability mass and reflect what the model actually “knows” at that step.
- The authors argue min-p’s success hints that this informative region behaves approximately like a uniform band in logit space.
- That observation motivates using a distance below the maximum logit, rather than a probability threshold, to define the candidate set.

Takeaway: preserve the small useful band near the maximum logit and suppress the noisy bulk.

Source: Tang et al. (2024), Sec. 2.2

How do they choose the boundary?

The paper rejects classic outlier detection and instead measures distance downward from the maximum logit



Relationship between σ-distance and nucleus size when temperature T = 1. High σ-distances = small nucleus sizes (mostly 1), lower σ-distances = larger nucleus sizes (remaining above 10).

What Figure 1 suggests:

- Measured σ-distance = (M - μ)/σ is often well above 10 during generation.
- Empirically, higher σ-distance corresponds to smaller useful nuclei, often just 1 token.
- So informative tokens should not be modeled as Gaussian outliers from the mean.
- Instead, start at the maximum M and keep tokens within nσ below it.

Proposed boundary: keep tokens with $l_i \geq M - n\sigma$

The top-nσ algorithm

A simple inference time filter that works directly on scaled logits

Algorithm (one step):

$$\text{Given scaled logits } l'_i = l_i/T, \quad M = \max_i l'_i, \quad \sigma = \text{std}(l')$$

$$m_i = \mathbf{1}[l'_i \geq M - n\sigma]$$

$$p = \text{softmax}(l')$$

- Compute max and standard deviation.
- Mask out all tokens below M - nσ.
- Sample only inside the surviving set.

Why the authors like it:

- No extra softmax manipulation before filtering.
- No sorting step, unlike top-k or top-p style truncation.
- Efficient GPU primitives: max and std are cheap.
- Easy to add to existing inference frameworks.

Source: Tang et al. (2024), Algorithm 1

What makes top-nσ different?

How top-nσ differs from the other commonly used sampling methods

Other algorithms

- **Greedy:** The highest probability
- **Top-k:** Only the k highest probabilities
- **Top-p:** Smallest probability set adding to at least p
- **Min-p:** Keep only tokens at least p% as likely as the highest probability

Top-nσ:

- Finds the highest probability token
- Calculates the standard deviation of the logits
- Masks out all tokens n standard deviations below the maximum
- Renormalizes the remaining tokens
- Randomly chooses a token from the remaining token set

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$$

Theory: probability mass and the role of n

The paper analyzes two limiting cases to argue that top-nσ preserves informative tokens while suppressing noise

Gaussian boundary case:

- When logits look Gaussian, the retained mass can be expressed through an error function term.
- As σ approaches 0, the retained mass from the noisy region goes to 0, which is exactly what we want.
- Intuition: the method naturally avoids swallowing the Gaussian bulk when there is little meaningful spread.

Uniform boundary case:

- If the informative region is roughly uniform over a band below M, the paper derives a lower bound on retained mass.
- For n = 1 and a typical σ = 1.9, the lower bound is about 0.85.
- This is the authors' argument that top-nσ keeps most useful tokens while excluding low-value ones.

Hyperparameter guidance from the paper

n should be positive, below $2\sqrt{3} \approx 3.46$ in theory, and empirically ≥ 0.5 with 1.0 as the recommended default.

Source: Tang et al. (2024), Sec. 3.1 and 4.1

Theory: temperature invariance

This is the most distinctive theoretical property claimed by the paper

Result:

For any temperature T > 0, the selected set is unchanged because both M and σ scale by 1/T.

$$l_i' \geq M' - n\sigma' \iff \frac{l_i}{T} \geq \frac{M}{T} - \frac{n\sigma}{T} \iff l_i \geq M - n\sigma$$

So temperature only changes how we explore inside the nucleus, not which tokens are considered valid.

Why that is useful:

- Top-p and min-p can expand the candidate set as temperature rises, which admits more noise tokens.
- Top-k is temperature-invariant too, but k is static and does not adapt to context.
- Top-nσ separates two knobs: n sets the valid region; temperature only shapes exploration within it.
- This makes the method appealing for repeated sampling and test-time scaling.

Source: Tang et al. (2024), Sec. 3.2

Experimental setup

Benchmarks span from school math to graduate level scientific QA

Model and framework

LLaMA-3-8B-Instruct with vLLM inference framework.

Datasets

Dataset	Task type	Difficulty	Single-pass metric	Multi-pass metric
AQuA	Math word problems	Reasoning	Exact Match	Maj@20
MATH	Competition-style math	Hard reasoning	Exact Match	Maj@20
GSM8K	Grade-school math	Moderate	Exact Match	Maj@20
GPQA	Graduate science MCQ	Very hard	Exact Match	Maj@20

Baselines: plain sampling, top-p = 0.9, top-k = 20, min-p = 0.1, top-nσ = 1.0; hyperparameters are fixed across temperatures.

Example Questions

The type of questions likely to be found in each data set

AQuA

Basic, MCQ

A car can drive 150 miles in 5 hours, how far can it go in 7 hours?

A) 450
B) 210
C) 300
D) 180

GSM8K

Basic, FRQ

A bus has 14 passengers. At its next stop, it lets off 3 and picks up 5, how many passengers are on the bus?

MATH

Advanced, FRQ

Problem: The equation $x^2 + 2x = i$ has two complex solutions. Determine the product of their real parts.
Solution: Complete the square by adding 1 to each side. Then $(x+1)^2 = 1 + i = e^{\frac{1\pi}{4}}\sqrt{2}$, so $x + 1 = \pm e^{\frac{1\pi}{8}}\sqrt[4]{2}$. The desired product is then

$$\left(-1 + \cos\left(\tfrac{\pi}{8}\right)\sqrt[4]{2}\right)\left(-1 - \cos\left(\tfrac{\pi}{8}\right)\sqrt[4]{2}\right) =$$

$$1 - \cos^2\left(\tfrac{\pi}{8}\right)\sqrt{2} = 1 - \frac{\left(1+\cos\left(\frac{\pi}{4}\right)\right)}{2}\sqrt{2} = \boxed{\frac{1-\sqrt{2}}{2}}.$$

GPQA

Expert, MCQ

Question: A Fe pellet of 0.056g is first dissolved in 10mL of hydrobromic acid HBr (0.1M). The resulting solution is then titrated by KMnO4 (0.02M). How many equivalence points are there?
(A) Two points, 25ml and 35ml
(B) One point, 25mL
(C) One point, 10ml
(D) Two points, 25ml and 30ml

Source: AQuA dataset, GSM8K dataset, MATH dataset, GPQA dataset

Evaluation Metrics

How the two different evaluation metrics work

Single-Pass: Exact Match (EM)

- Generate a single response
- Receive a score of 1 if the answer is exactly correct and a score of 0 otherwise
- Average score over the entire dataset

Provides the answer to:
How good is a singular response to each question?

Repeated Sampling: Maj@20

- Sample 20 responses for each problem
- Take the majority answer over all 20 responses
- Recieve a score of 1 if the majority is exactly correct and a score of 0 otherwise
- Average score over the entire dataset

Provides the answer to:
How reliably can the model answer questions correctly?

Single-pass results

Top-nσ is the only stochastic method reported to consistently beat greedy decoding across all four datasets

Dataset	Method	Temperature				
		0.0	1.0	1.5	2.0	3.0
GPQA	Sample	**32.03**	30.47	14.84	7.03	0.00
	Top-p	–	**30.86**	20.31	8.98	0.00
	Top-k	–	29.69	25.00	19.14	7.42
	Min-p	–	27.73	31.25	26.95	16.02
	Top-$n\sigma$	–	27.34	**32.42**	**27.73**	**25.00**
GSM8K	Sample	**81.25**	76.95	21.48	0.00	0.00
	Top-p	–	78.52	66.02	0.00	0.00
	Top-k	–	75.78	62.11	21.88	2.34
	Min-p	–	**80.47**	76.56	66.41	14.84
	Top-$n\sigma$	–	78.52	**82.03**	**79.30**	**74.61**
AQuA	Sample	**36.61**	–	–	–	–
	Top-p	–	39.76	–	–	–
	Top-k	–	39.76	30.71	21.65	–
	Min-p	–	37.80	37.01	33.07	–
	Top-$n\sigma$	–	**41.73**	**40.94**	**40.16**	–
MATH	Sample	**19.92**	–	–	–	–
	Top-p	–	16.41	–	–	–
	Top-k	–	14.06	10.55	3.91	–
	Min-p	–	15.63	14.45	10.94	–
	Top-$n\sigma$	–	**20.31**	**16.02**	**14.06**	–

What the table actually supports:

- AQuA: top-nσ is best among listed methods at T = 1.0, 1.5, and 2.0.
- MATH: top-nσ at T = 1.0 beats greedy (20.31 vs 19.92), but performance falls at higher T.
- GSM8K and GPQA: the best top-nσ point is at T = 1.5, not T = 1.0.

Most important empirical pattern:

- At high temperatures, standard sampling and top-p collapse much faster.
- top-nσ remains much more robust: e.g. at T = 3.0 it still reports 74.61 on GSM8K and 25.00 on GPQA.

Source: Tang et al. (2024), Table 2

Repeated sampling / Maj@20 results

The method still supports exploration, which is important for test-time scaling

Dataset	Method	Temperature			
		1.0	1.5	2.0	3.0
GSM8K	Sample	**90.63**	75.00	0.00	0.00
	Top-p	89.06	89.45	0.00	0.00
	Top-k	89.45	**91.41**	62.89	2.73
	Min-p	89.84	90.63	89.84	53.13
	Top-nσ	**90.63**	**91.41**	**91.80**	**90.23**
GPQA	Sample	30.47	27.34	12.89	0.00
	Top-p	30.08	27.34	12.89	0.00
	Top-k	**32.03**	31.64	26.17	24.61
	Min-p	30.47	**33.20**	31.25	**30.47**
	Top-nσ	31.64	**33.20**	**32.42**	**30.47**
AQuA	Sample	-	-	-	-
	Top-p	44.88	-	-	-
	Top-k	48.03	48.03	40.16	-
	Min-p	44.09	51.18	47.64	-
	Top-nσ	47.64	46.06	49.61	-
MATH	Sample	-	-	-	-
	Top-p	32.03	-	-	-
	Top-k	31.25	20.70	12.50	-
	Min-p	30.86	28.91	23.83	-
	Top-nσ	32.03	35.16	33.98	-

Main readout:

- All stochastic methods improve with repeated sampling on several datasets.
- top-nσ remains competitive and often excellent, especially as temperature rises.

Examples:

- GSM8K: top-nσ reaches 91.80 at T = 2.0 and still 90.23 at T = 3.0.
- MATH: top-nσ improves from 32.03 at T = 1.0 to 35.16 at T = 1.5.

Interpretation:

- The method does not sacrifice exploration. Instead, it tries to keep exploration inside a validated token set.

Source: Tang et al. (2024), Table 3

Critical discussion

Why this paper is interesting and where to be cautious

Strengths:

- Simple, actionable idea with a clear implementation path.
- Strong conceptual story: separate candidate validation from exploration.
- The temperature-invariance theorem cleanly explains the observed robustness.
- Empirical results are intriguing because they challenge the “greedy is best for reasoning” norm.

Caveats / open questions:

- The paper is a short preprint and explicitly says the version is incomplete.
- Most theoretical arguments rely on stylized Gaussian/uniform approximations rather than a full generative model of logits.
- Evaluation uses one model family; cross-model generality is not yet established.
- The improvements are not uniformly dominant in every table cell, so the biggest contribution may be robustness rather than absolute SOTA.

Research directions: validate on larger models, inspect token-level calibration, and connect this inference-time view back to training objectives that suppress noise of silence.

Source: Tang et al. (2024), conclusion + authors’ own caveat

Takeaways

1. Raw logits appear to contain a useful geometric separation between a noisy bulk and an informative tail.

2. top-nσ keeps tokens within n standard deviations below the maximum logit, rather than thresholding on probabilities.

3. Because the selected set is temperature invariant, temperature can be used for exploration without constantly expanding the candidate pool.

4. On the reported benchmarks, this simple rule is competitive with or better than standard sampling and can even beat greedy decoding for reasoning.

Source: Tang et al. (2024)

Thank You!