Smoothie: Smoothing Diffusion on Token Embeddings for Text Generation

Semantic geometry, embedding regression, and sampling trade-offs

Paper authors: Alexander Shabalin,
Viacheslav Meshchaninov, Dmitry Vetrov

Research group reading | 14 October 2026
arXiv:2505.18853v2, revised 15 May 2026

Use token distances to define corruption,
and predict embeddings to reverse it.

Can corruption preserve useful token relationships?

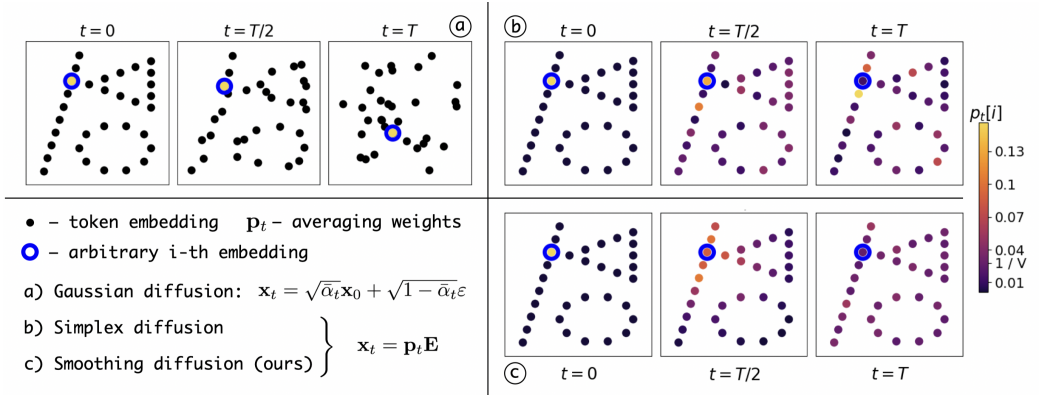
The design problem

Text needs both a meaningful notion of similarity and a clear route back to discrete tokens.

Representation	What carries information?	Main issue or proposed benefit
Embedding	A continuous vector in $\mathbb{R}^d$	Semantic geometry; requires token decoding
Simplex	One score per vocabulary item	Clear token coordinates; all wrong tokens start equally far away
Smoothie	Distances to every token	Token scores with semantic similarity

Paper, Secs. 1–3, PDF pp. 1–3. Earlier discrete methods also explored semantic corruption.

Smoothing spreads weight through embedding geometry



- Panel (c): nearby embeddings receive weight before distant ones.
- Colors are vocabulary weights; the blue circle marks the original token.

Paper, Fig. 1, PDF p. 2. Original schematic, not an experimental scatter plot.

A token becomes a vector of vocabulary distances

For a target sequence of length m and fixed embeddings $\mathbf{E} \in \mathbb{R}^{V \times d}$:

$$[\mathbf{D}_0]_{ij} = -\frac{1}{2}\|\mathbf{E}_{w_i^y} - \mathbf{E}_j\|^2, \qquad \mathbf{D}_0 \in \mathbb{R}^{m \times V}.$$

- The original token has score 0; nearby tokens have less negative scores.
- Each coordinate still names a discrete vocabulary item.
- Clean distance scores decode by nearest embedding:

$$\arg\max_j[-\tfrac{1}{2}\|\hat{e} - \mathbf{E}_j\|^2] = \arg\min_j \|\hat{e} - \mathbf{E}_j\|^2.$$

Where the semantic assumption enters

Euclidean proximity in the fixed embedding space should reflect useful token similarity.

Paper, Sec. 4.1, Eq. (2), PDF p. 3. Ties are possible for identical embeddings.

Forward corruption shrinks the signal at fixed noise

$$\mathbf{D}_t = \frac{\mathbf{D}_0}{\sigma_t^2} + \delta\boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, I), \quad \delta = 1.$$

$$\mathbf{p}_t = \mathrm{softmax}(\mathbf{D}_t), \qquad h_t = \mathbf{p}_t \mathbf{E} \in \mathbb{R}^{m \times d}.$$

- Larger bandwidth σ_t weakens distance differences.
- Without logit noise, weights are normalized Gaussian kernels:

$$p_{t,ij} \;\propto\; \exp\left(-\frac{\|\mathbf{E}_{w_i^y} - \mathbf{E}_j\|^2}{2\sigma_t^2}\right).$$

- The Transformer consumes weighted embeddings, not V-dimensional hidden states.
- Given $\mathbf{D}_0$, noisy states at different times are sampled independently.

Paper, Sec. 4.1, Eq. (3), PDF pp. 3–4; implementation, p. 5.

Embedding MSE recovers the optimal distance weights

The practical training objective predicts d-dimensional embeddings:

$$L_{\mathbf{E}} = \mathbb{E}\big[\|\mathbf{E}_{\mathbf{w}^y} - f_\theta(\mathbf{p}_t, t)\|^2\big].$$

Let U be a clean embedding and Z the observed noisy input and condition.

$$\mu = \mathbb{E}[U \mid Z],$$
$$\mathbb{E}[-\tfrac{1}{2}\|U - \mathbf{E}_j\|^2 \mid Z] = -\tfrac{1}{2}\|\mu - \mathbf{E}_j\|^2 - \tfrac{1}{2}\operatorname{tr}\operatorname{Cov}(U \mid Z).$$

Why this matters

The covariance term is the same for every candidate j. Softmax removes this row-wise shift.

The identity concerns optimal regressors with squared Euclidean distances.

Paper, Theorem 4.1 and Eq. (8), PDF p. 4; proof in Appendix C, p. 14. Conditional notation made explicit.

Sampling rebuilds a noisy state from each prediction

Start with $\mathbf{D}_T \sim \mathcal{N}(0, \tilde{\delta}^2 I)$. At each step:

$$\mathbf{p}_t = \mathrm{softmax}(\mathbf{D}_t), \qquad \hat{e}_t = f_\theta(\mathbf{p}_t, t),$$

$$\boxed{\mathbf{D}_{t-1} = \frac{\mathbf{D}_0(\hat{e}_t)}{\sigma_{t-1}^2} + \tilde{\delta}\varepsilon.}$$

- No explicit residual term carries the current noisy latent into the update.
- The prediction still depends on $\mathbf{D}_t$ through $\mathbf{p}_t$ and on self-conditioning.
- $\tilde{\delta}$ controls generation randomness independently of training noise.
- Algorithm 2 takes vocabulary argmax of the final scores; padding is removed.

Paper, Sec. 4.2, Eq. (9), Algorithm 2, PDF p. 4. Eq. (9) includes noise in the final score update.

The model processes soft embeddings with a Transformer

Noisy scores $\mathbf{D}_t$ $\longrightarrow$ softmax$(\mathbf{D}_t)\mathbf{E}$ $\longrightarrow$ Predicted embeddings

Denoiser

- 12 Transformer layers, width 768, 12 attention heads.
- U-Net style additive skips between early and late layers.
- Learned timestep embeddings in each block.

Conditioning and tokens

- Conditional tasks add a 6-layer source encoder and cross-attention.
- Fixed, normalized BERT embeddings.
- Predict padding to support different output lengths.

Approximately 100M-scale models; denoisers are trained per task, with pretrained embeddings.

Paper, Secs. 4.5–5, PDF p. 5; Appendix J, Table 12, p. 17.

A bandwidth schedule and self-conditioning shape denoising

$$\sigma_t = (\sigma_{\max} - \sigma_{\min})\frac{2}{\pi}\arctan\left(\frac{1}{d_s}\sqrt{\frac{t}{T - t + \epsilon}}\right) + \sigma_{\min}.$$

- $\sigma_{\min} = 1.5$, $\sigma_{\max} = 200$; schedule shape $d_s \in \{5, 7\}$.
- Smaller d_s means stronger corruption at the same relative time.

Self-conditioning

Training: with probability $1/2$, use zero; otherwise use a detached preliminary prediction at the same noisy input.

Generation: use zero first, then the previous denoising step's prediction.

Paper, Secs. 4.3–4.4, Eq. (10), PDF pp. 4–5. Here d_s distinguishes the schedule parameter from embedding width; the paper uses d for both.

Evaluation covers five tasks and two views of quality

Dataset	Task	Main metrics
ROCStories	Unconditional stories	MAUVE ↑, PPL ↓, diversity ↑
XSum	Summarization	BERTScore, ROUGE ↑
Quasar-T	Question generation	BERTScore, BLEU, R-L ↑
QQP	Paraphrase generation	BERTScore, BLEU, R-L ↑
ParaDetox	Detoxification	BLEU, J-Score ↑

- ROCStories MAUVE: five generation seeds, 1,000 samples each, then average.
- Generative PPL evaluates output text; it is not Smoothie's held-out likelihood.
- Internal variants share architecture; schedules differ, and Simplex uses cross-entropy rather than MSE.

Paper, Sec. 5, PDF pp. 5–7. External baseline rows combine prior reports and reproductions.

Computing over the vocabulary adds sampling cost

Method	Time / s ↓	Peak memory / MB ↓
Embedding	1.642	593.3
Simplex	2.897	678.4
Smoothie	2.897	593.3

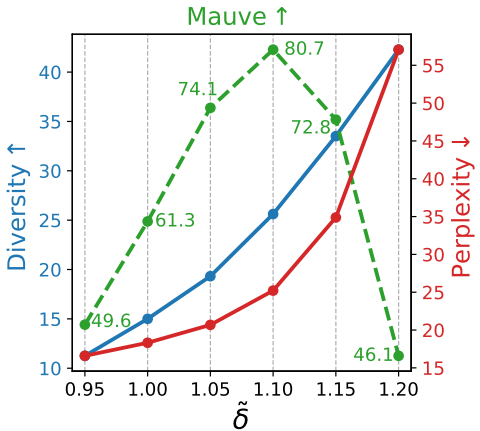
- Measurement: 100 generation steps, batch 32, target length 80.
- Computing distances to all embeddings costs $O(BmVd)$ per step.
- The paper gives Embedding 1.75× more steps to approximately match generation cost.

Interpretation

Low-dimensional network outputs do not eliminate vocabulary-wide work.

Paper, Table 2, PDF p. 6; generation-speed protocol, pp. 6–7. The measured time ratio is about 1.76.

Sampling noise exposes a quality–diversity trade-off



- Blue: diversity, left axis.
- Red: PPL, right axis.
- Green labels: MAUVE values.
- Training noise stays at $\delta = 1$.

ROCStories: best plotted MAUVE at $\tilde{\delta} = 1.1$.

Conditional tasks: use $\tilde{\delta} = 0.1$ after a separate search.

Paper, Fig. 2, Sec. 5.1, PDF pp. 5–6; conditional noise search, Table 8, p. 15.

ROCStories shows the largest distribution-quality gain

Method	MAUVE ↑	PPL ↓	Div ↑
GPT-2 (AR reference)	78.9	20.5	25.2
MDLM	63.9	58.1	35.1
TEncDM	76.2	29.1	29.5
Embedding	41.5	28.3	26.1
Simplex	15.2	25.3	12.4
Smoothie	**80.7**	**24.9**	25.1

- Versus TEncDM: MAUVE +4.5 points, with lower PPL and lower diversity.
- Simplex has a competitive PPL but a much worse MAUVE.
- GPT-2 still has the lowest PPL in this selection.

Paper, Table 4, PDF p. 8. Selected rows; numbers use the paper's reporting scale.

XSum improves ROUGE, while BERTScore favors TEncDM

Method	BS ↑	R-1 ↑	R-2 ↑	R-L ↑
FLAN-T5 (AR)	72.7	34.6	12.9	27.2
TEncDM	**69.9**	31.9	10.7	25.3
Embedding	68.2	32.1	10.1	24.6
Simplex	63.8	29.6	8.5	23.0
Smoothie	68.8	**33.7**	**11.1**	**26.0**

What the result supports

Better lexical coverage among the compared diffusion models.

- R-L: +1.4 versus internal Embedding, +0.7 versus TEncDM.
- BERTScore: 1.1 points below TEncDM; FLAN-T5 is stronger on all listed metrics.

Paper, Table 3 (XSum), PDF p. 7. Bold highlights the best diffusion value in each column.

Quasar-T gains depend on the metric

Method	BS ↑	BLEU ↑	R-L ↑
BART (AR)	66.2	17.4	38.8
SSD-LM	62.8	14.1	**38.5**
TESS (random initialization)	60.8	19.0	36.1
Embedding	62.0	18.9	35.2
Simplex	63.0	19.3	36.9
Smoothie	**63.1**	**19.9**	36.5

- Best diffusion BS and BLEU, but not the best R-L.
- Internal Simplex already improves on TESS: architecture also matters.
- Against BART, BLEU is higher while BS and R-L are lower.

Paper, Table 3 (Quasar-T), PDF p. 7. Selected rows.

QQP and detoxification show complementary gains

QQP: reference matching

Method	BS	BLEU	R-L
BART	85.7	30.4	61.4
Embedding	83.4	31.3	59.4
Simplex	80.6	26.8	54.9
Smoothie	83.9	30.8	60.9

Higher BS and R-L than Embedding, but lower BLEU.

ParaDetox: style conversion

Method	BLEU	J-Score
GPT-2	67.7	60.4
Embedding	67.6	49.1
Simplex	65.1	47.7
Smoothie	69.2	51.7

Stronger diffusion scores; GPT-2 retains a higher J-Score.

All metrics shown increase with better performance.

Paper, Table 4, PDF p. 8. Main-table QQP values differ from some ablation/default-setting entries; see backup.

More denoising steps do not consistently improve quality

Steps	XSum BS	Quasar-T BS	QQP BS	ParaDetox J
25	67.7	63.1	**83.9**	51.1
50	68.5	63.1	83.8	51.4
100	68.7	63.1	83.7	**51.7**
200	**68.8**	63.1	83.6	51.0
500	68.4	63.1	83.5	50.8

- XSum benefits up to 200 steps; QQP peaks at 25 in this sweep.
- Quasar-T BS is unchanged at the reported precision.
- Authors suggest self-conditioning mismatch as one explanation for late degradation.

Paper, Sec. 5.3, Table 5, PDF pp. 7–8. Higher is better for every metric shown.

Rebuilding from predictions gives modest solver gains

Solver	XSum R-L	Quasar-T R-L	QQP R-L	ParaDetox J
DDPM	25.6	35.5	60.7	51.5
DDIM	24.6	31.8	57.9	50.2
Eq. (9)	**26.0**	**36.5**	**60.9**	**51.7**

- The proposed update wins on all metrics in this comparison; margins over DDPM are small.
- The DDPM/DDIM comparison introduces time-dependent forward variance:

$$\mathrm{Var}(\mathbf{D}_t \mid \mathbf{D}_0) = \left(1 - \sigma_t^{-4}\right) \delta^2 I.$$

- Reduced drift is the authors' explanation, not a proved mechanism.

Paper, Sec. 5.4, Eq. (11), Table 6, PDF p. 8. Selected metric columns, all higher is better.

OpenWebText uses a hybrid solver and shows smaller gains

Method	MAUVE ↑	PPL ↓	Div ↑
GIDD	28.6	228.3	58.8
Embedding	55.7	51.9	21.2
Smoothie, $\tilde{\delta} = 0.96$	58.9	**48.2**	19.8
Smoothie, $\tilde{\delta} = 1.02$	**59.9**	62.2	23.4

- Target length 512; Smoothie uses 100 steps, Embedding 175.
- Smoothie: first 40% DDPM, then Eq. (9); Embedding: DDPM throughout.
- Best MAUVE gain over Embedding is 4.2 points, with a PPL/diversity trade-off.

Paper, Appendix D, Table 7, PDF pp. 14–15. This is an explicit exception to the main-text solver description.

The embedding geometry is an important design choice

Embedding source	MAUVE ↑	PPL ↓	Div ↑
BERT	80.7	24.9	25.1
GPT-2	64.4	23.1	25.0
GloVe	36.8	36.4	24.6

- On ROCStories, BERT embeddings yield much better MAUVE.
- GPT-2 embeddings lower PPL without matching distribution quality.
- This also changes token vocabularies: GPT-2 is about 50k; GloVe BPE is 10k.

Design consequence

The corruption geometry is inherited from a pretrained representation.

Paper, Appendix F, Table 9, PDF p. 15. GloVe embeddings were trained by the authors on Wikipedia.

Three lessons from Smoothie

1. **Design the corruption space.**
   Distances preserve vocabulary coordinates while giving nearby tokens related scores.

2. **Use the geometry to simplify learning.**
   Squared-distance regression reduces to embedding MSE up to a softmax-invariant shift.

3. **Tune sampling to the task.**
   Noise, iteration count, and solver choice affect the quality–diversity balance.

Main costs: fixed embedding geometry and vocabulary-wide computation at each step.

Shabalin, Meshchaninov, and Vetrov. Smoothie: Smoothing Diffusion on Token Embeddings for Text Generation. arXiv:2505.18853v2, 2026. Main talk ends here; following slides are backup.

Backup: a three-token example of kernel smoothing

Teaching example: squared distances to the original token are $0, 4, 36$.
Ignore the stochastic logit term and normalize $\exp[-r^2/(2\sigma^2)]$.

Bandwidth σ	Original	Nearby	Distant
1.5	0.7085	0.2913	0.0002
3	0.5165	0.4136	0.0699
6	0.3918	0.3706	0.2376

- Small bandwidth mixes the original token with its neighbor first.
- With actual noise, the large-bandwidth limit is $\mathrm{softmax}(\delta\varepsilon)$, not a fixed uniform vector.

Teaching calculation based on Eq. (3), PDF p. 3; not a paper experiment or real-token measurement.

Backup: the simplified objective and its indexing

The variational bound contains Gaussian KL terms for $t \to t-1$:

$$\mathrm{KL}\left(q(\mathbf{D}_{t-1} \mid \mathbf{D}_0) \,\|\, p_\theta(\mathbf{D}_{t-1} \mid \mathbf{D}_t)\right).$$

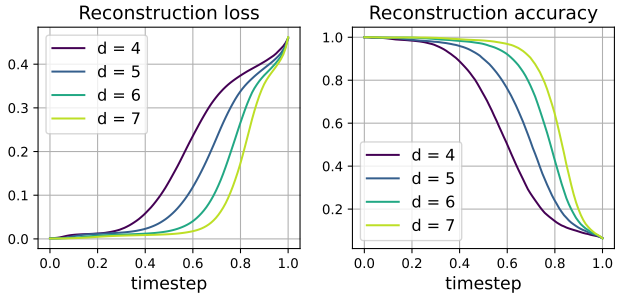
With Eq. (3), its model-dependent part for $t > 1$ is proportional to

$$\frac{1}{2\tilde{\delta}^2} \left\| \frac{\mathbf{D}_0}{\sigma_{t-1}^2} - \mu_\theta(\mathbf{p}_t, t) \right\|^2.$$

- Predict clean distances and drop the time weighting to obtain $L_{\mathbf{D}}$.
- The paper's Eqs. (6) and (23) use σ_t here; Eq. (9) uses σ_{t-1}.
- Algorithm 1 also prints $\mathbf{D}_t$ in the clean-input position; Eq. (3) requires $\mathbf{D}_0$.

Paper, Sec. 4.2 and Algorithm 1, PDF p. 4; Appendix B, pp. 13–14. Index-consistent expression shown above.

Backup: the schedule shifts where reconstruction becomes hard



ROCStories MAUVE for schedule $d = 4, 5, 6, 7$: **74.2, 80.7, 76.5, 68.9**.
Each setting selects its own best $\tilde{\delta}$.

- Left: reconstruction loss; right: non-padding token accuracy.
- Smaller d removes usable information earlier.

Paper, Fig. 3, PDF p. 16; Table 10, p. 15. The caption mentions training with $d = 5$ while the legend varies d; separate training per curve is not clarified.

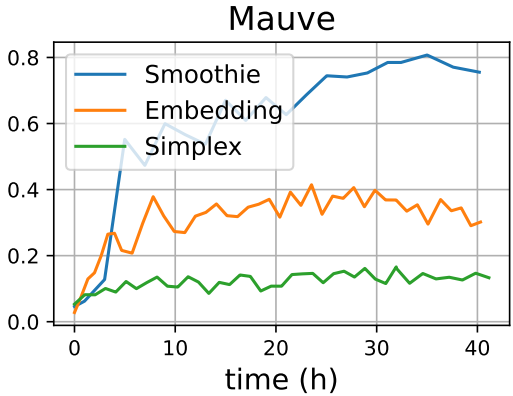
Backup: self-conditioning helps Smoothie unevenly

Task	Metric	With SC	Without SC
XSum	BERTScore	68.8	67.5
XSum	ROUGE-L	26.0	25.4
Quasar-T	BERTScore	63.0	61.9
Quasar-T	BLEU	19.0	19.0
QQP	BERTScore	83.9	83.2
QQP	BLEU	30.8	29.4

- Larger gains on XSum BS and QQP BLEU; Quasar-T BLEU is unchanged.
- Embedding diffusion on Quasar-T improves when SC is removed in this ablation.

Paper, Appendix G, Table 11, PDF p. 16. Use within-table pairs; its Quasar-T values differ from the main table.

Backup: the MAUVE advantage persists over much of training



- X-axis: training hours.
- Y-axis: MAUVE on a 0–1 scale.
- Evaluation uses 350 steps for Embedding and 200 for the other variants.
- After the initial phase, Smoothie leads through most of the plotted interval.

Paper, Appendix I, Fig. 4, PDF p. 16. The main tables report MAUVE on a 0–100 scale.

Backup: default configurations and reported variations

	ROCStories	XSum	Quasar-T	QQP	ParaDetox
Target length	80	64	50	50	40
Train steps	1M	225k	150k	50k	150k
Sample steps	200	200	100	100	100
$\tilde{\delta}$	1.1	0.1	0.1	0.1	0.1
Schedule d	5	5	7	5	7

- AdamW, learning rate 2×10^{-4}, 5k warmup, EMA 0.9999.
- QQP main-table BS is 83.9; the 100-step sweep reports 83.7, despite the listed default of 100 steps.
- The SC ablation's Quasar-T row also differs from the main table. The paper does not reconcile these entries.

Paper, Table 12, PDF p. 17; compare Tables 4–5, p. 8, and Table 11, p. 16.

Backup: semantic degradation is observed through a decoder

Noise level	Qualitative change in the example
Low	Related substitutions: the number of children changes; one US state replaces another.
Intermediate	Semantic relations and sentence structure become unreliable.
High	Frequent tokens and repetitions dominate the reconstructed text.

How to interpret the example

The authors sample noisy forward latents, run the model, and decode its predictions. These are reconstructions, not direct token samples from the forward weights.

Paper, Appendix L, Table 13, PDF p. 18. Paraphrased qualitative description.