SOL: MEASURING GAPS BETWEEN TEXT DISTRIBUTIONS BY DOUBLE SLICED WASSERSTEIN METRICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Evaluating text generation requires measuring how well the generated distribution matches the data distribution. For autoregressive models, this is done by the perplexity. Diffusion and flow-based language models can only provide a likelihood bound, whose tightness differs between model families. Sample-based substitutes such as generative perplexity with entropy do not consider the distribution fit. We propose SOL, a distance between text distributions. Each sequence is represented by the empirical measure of its hidden states under a fixed transformer and the distributions of these measures are compared by the double sliced Wasserstein distance. We prove that SOL is a metric if the transformer is injective. Experiments show that SOL detects distributional failures, recovers expected model trends, and provides stable sample-based estimates. We put forward SOL to fill the gap in the current evaluation protocol used for non auto-regressive models. As a first step we use SOL to re-evaluate a variety of models trained on OpenWebText.

1 INTRODUCTION

Large language models based on autoregressive decoding are the prevailing approach to text generation. Recently, diffusion and flow-based models, which generate the whole sequence iteratively by reversing a noising process, have become a serious alternative. Discrete diffusion models corrupt the tokens by masking or by uniform replacement (Austin et al., 2021; Lou et al., 2024; Sahoo et al., 2024; 2025). Continuous models embed the tokens into an embedding space and define the noise process there (Li et al., 2022; Dieleman et al., 2022; Gulrajani & Hashimoto, 2023; Hu et al., 2026; Chen et al., 2026; Chemseddine et al., 2026). A natural question is how well the generated text distribution p_{gen} approximates the data distribution p_{data}.

For autoregressive models, the standard answer is the perplexity of held-out data, which is computed from the exact likelihood p_θ. For diffusion models, the likelihood is intractable. Instead, an upper bound of the perplexity is reported, based on an evidence lower bound (ELBO). This causes two problems. First, an ELBO is often not available. For continuous models, the bound holds only in expectation over the noise, requires the change-of-variables formula along the flow and has only been derived for selected models (Gulrajani & Hashimoto, 2023; Chen et al., 2026). Moreover, most recent models use self-conditioning, where the network receives its own previous prediction as input. This input arises along the sampling trajectory and cannot be recovered for a given data sequence. Second, even when an ELBO exists, it is not comparable across model families. Sahoo et al. (2026) observe that perplexity ranks models correctly within one family of diffusion models but not between families. The reason is that the gap between ELBO and likelihood is a property of the noise process and its parameterization, not of the model alone. For masked diffusion, the bound is tight (Sahoo et al., 2024; Shi et al., 2025), whereas for uniform diffusion it is looser and not even optimized by the training objective (Gourevitch et al., 2026). A better perplexity may thus reflect a tighter bound rather than a better model.

Since every model can be sampled, the generative perplexity (Gen. PPL) together with the token entropy has become the common substitute (Zheng et al., 2025; Franca & Tong, 2026; Hu et al., 2026; Chen et al., 2026; Chemseddine et al., 2026). Here, Gen. PPL is the perplexity of the generated texts under a fixed reference model such as GPT-2 Large, and the entropy is computed from token frequencies within sequences. Since both are averaged over sequences, they do not reward variety in generated outputs, see Figure 4 for an example of mode collapse. Moreover, Gen. PPL measures agreement with the reference model and not with p_{data}. Hence, it rewards imitation of the reference model instead of actual data. In short, neither quantity measures the distance between p_{gen} and p_{data}.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

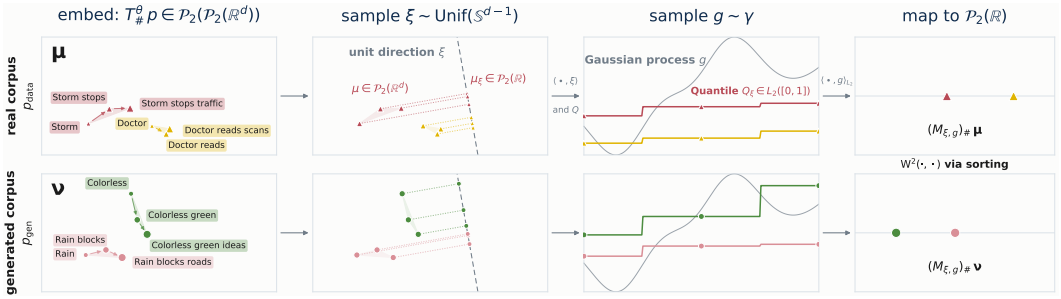


Figure 1: Visualizing a slice of SOL for fixed slicing directions ξ, g with a causal transformer. As the context window grows, a sentence traces out a point cloud $T^{\theta}(\mathbf{x}) \in \mathcal{P}_2(\mathbb{R}^d)$, and a corpus becomes a collection of such clouds. Slicing the clouds along a vector ξ, reading off quantile functions and slicing them along a function g, lands the corpus in $\mathcal{P}_2(\mathbb{R})$. Semantically close sentences (red dots) end up close on the final line, while the nonsensical text (green) sits to the left. We obtain our SOL (DSW) by integrating over ξ and g.

For image generation, the dominant evaluation method is FID (Heusel et al., 2017). It maps samples of both distributions into the feature space of a fixed network and compares the two feature distributions there. We would like to have a similar score for text distributions which

(1) requires only samples from p_{gen} and p_{data},
(2) is a distance, i.e., it is zero if and only if the two distributions agree,
(3) reflects both the quality of the generated texts and their coverage of p_{data},
(4) has an estimator which is stable in the number of samples, and
(5) is cheap to compute.

The most popular alternative to Gen. PPL is MAUVE (Pillutla et al., 2021; 2023). It embeds each text by a latent representation, quantizes both samples by K-means clustering and compares the resulting histograms through a KL-divergence frontier. However, MAUVE is not a distance, its value depends on the number of samples and clusters, and the choice of representation is problematic: the classically used last hidden state of a causal text transformer is insensitive to changes at the beginning of the sequence (He et al., 2023), while averaging over hidden states may map distinct sequences to similar vectors (Hara et al., 2026). Instead, we keep the whole set of hidden states and represent *each* sequence by their empirical measure in $\mathcal{P}_2(\mathbb{R}^d)$. The distribution of *all* sequences then becomes a probability measure on $\mathcal{P}_2(\mathbb{R}^d)$, which can be compared by a hierarchical Wasserstein distance (Piening et al., 2026; Bonet et al., 2025; Piening & Beinert, 2026b). Since this nested transport problem is expensive, we use the double sliced Wasserstein distance (Piening & Beinert, 2026b): first the hidden states are projected onto random directions, then the resulting one-dimensional measures are represented by their quantile functions and sliced again in $L_2([0,1])$. This sorting-based procedure gives *sliced optimal transport for language* (SOL), which we visualize in Figure 1. We give a detailed overview of related work in Appendix B.2. A Python implementation of SOL is provided in the supplementary material.

Contributions.

1. We propose SOL, a sample-based distance between text distributions, by representing each text as an empirical distribution of contextual token states and comparing the resulting distributions with the double sliced Wasserstein distance. Under an injectivity assumption, SOL is a metric on empirical text distributions (Proposition 4.2).
2. We validate SOL under structural perturbations, mode collapse, sampling changes, and human judgments. SOL detects failures missed by Gen. PPL and token entropy, yields stable finite-sample estimates (Sections 5.2), and correlates better with the evaluated human rankings than MAUVE (Table 2).
3. We use SOL to evaluate a broad range of autoregressive, discrete and continuous diffusion language models on OpenWebText (Table 22).

2 Preliminaries

Let $\mathcal{V}$ be a finite vocabulary of tokens, and let $\mathcal{V}^{\leq L} := \bigcup_{l=1}^{L} \mathcal{V}^l$ be the set of token sequences of length $\leq L$. By $\mathcal{P}(\mathcal{V}^{\leq L})$, we denote the set of probability measures on $\mathcal{V}^{\leq L}$. Given a sequence

of tokens $\mathbf{x} = (x_1, \dots, x_{|\mathbf{x}|})$ of length $|\mathbf{x}|$, we write $\mathbf{x}_{\leq l} = (x_1, \dots, x_l)$ for its partial sequence consisting of the first $l \leq |\mathbf{x}|$ tokens. Since all evaluation methods between text distributions in this paper use transformers in a certain way, we give a brief description for fixing the notation.

Causal transformers. A *causal decoder-only transformer* $\mathbf{H}^\theta$ computes a hidden state at each position, which depends only on the preceding states, resulting in

$$\mathbf{H}^\theta : \mathcal{V}^{\leq L} \to \mathbb{R}^{d \times L}, \quad \mathbf{x} \mapsto (h^\theta_{\mathbf{x}\leq 1}, ..., h^\theta_{\mathbf{x}\leq L}). \tag{1}$$

From the output layer, we get the probabilities of the next token recursively by $p_\theta(x_l \mid \mathbf{x}_{\leq l-1}) := \mathrm{softmax}(W_\theta h^\theta_{\mathbf{x}\leq l-1} + b_\theta)$. A transformer is used as an autoregressive generative model by factorizing the probability of a sequence $\mathbf{x} = (x_1, \dots, x_L)$ as

$$p_\theta(\mathbf{x}) = \prod_{l=1}^{L} p_\theta(x_l \mid \mathbf{x}_{\leq l-1}), \tag{2}$$

and generating a new sequence is done by iteratively evaluating the model p_θ and sampling from it.

Bidirectional transformers. Alternatively, a *bidirectional transformer* $\mathbf{H}^{\theta,\mathrm{bi}}$ computes hidden states at each position using the complete input sequence. In particular, for masked diffusion models, the input is a corrupted sequence $\mathbf{x}^t$ at diffusion time t without time conditioning, resulting in

$$\mathbf{H}^{\theta,\mathrm{bi}} : \overline{\mathcal{V}}^L \to \mathbb{R}^{d \times L}, \quad \mathbf{x}^t \mapsto \big(h^{\theta,\mathrm{bi}}_{\mathbf{x}^t,1}, \dots, h^{\theta,\mathrm{bi}}_{\mathbf{x}^t,L}\big),$$

where $\mathcal{V} \subset \overline{\mathcal{V}}$ may contain additional corruption tokens such as `[MASK]`.

3 Current Evaluation Methods for Language Models

We are interested how well a generative text distribution $p_{\text{gen}} \in \mathcal{P}(\mathcal{V}^{\leq L})$ approximates the data distribution $p_{\text{data}} \in \mathcal{P}(\mathcal{V}^{\leq L})$. We recall two groups of quantitative evaluation ways of assessing generated texts: perplexity-based scores and latent token comparisons in the Euclidean space.

3.1 Perplexity-Based Methods

Perplexity. For an autoregressive *learned* model p_θ in (2), the *perplexity*

$$\mathrm{PPL}(p_\theta, p_{\mathrm{data}}) = \exp\Big(- \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}\Big[\log p_\theta(\mathbf{x})\Big] / \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}[|\mathbf{x}|]\Big)$$

is the exponential of the token-length normalized cross-entropy between the data distribution and the model distribution. It measures how well the model assigns probability to samples from the data distribution p_{data}. Lower perplexity indicates that the model assigns higher likelihood to the observed data.

Evidence Lower Bound Perplexity. For many diffusion and latent-variable models, computing the exact log-likelihood $\log p_\theta$ is intractable. Instead, we can evaluate - if available - an *evidence lower bound* (ELBO) on the log-likelihood, $\log p_\theta(\mathbf{x}) \geq \mathrm{ELBO}_\theta(\mathbf{x})$. This gives rise to the ELBO-*based perplexity*

$$\mathrm{PPL}_{\mathrm{ELBO}}(p_\theta, p_{\mathrm{data}}) = \exp\Big(- \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}\Big[\mathrm{ELBO}_\theta(\mathbf{x})\Big] / \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}[|\mathbf{x}|]\Big),$$

which upper-bounds the perplexity obtained from the exact model likelihood. However, comparisons based on the ELBO should be interpreted with care, since its value depends not only on the model likelihood, but also on the tightness of the variational bound. As an example, the ELBOs of masked and uniform diffusion language models seem to differ Sahoo et al. (2026). Moreover, for many generative language models no valid ELBOs are available.

Generative Perplexity. An alternative is therefore to evaluate generated samples under a *fixed* reference model p_θ. Specifically, we choose a *pretrained* autoregressive causal transformer p_θ as the reference model and define the *generative perplexity* (Gen. PPL) as

$$\mathrm{GenPPL}(p_{\mathrm{gen}}, p_\theta) := \exp\Big(- \frac{1}{\mathbb{E}_{\mathbf{x}\sim p_{\mathrm{gen}}}[|\mathbf{x}|]} \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{gen}}}\Big[\sum_{l=1}^{|\mathbf{x}|} \log p_\theta(x_l \mid \mathbf{x}_{\leq l-1})\Big]\Big).$$

Ignoring the sequence-length normalization, the corresponding cross-entropy decomposes as $\mathbb{E}_{\mathbf{x}\sim p_{\mathrm{gen}}}[-\log p_\theta(\mathbf{x})] = \mathrm{H}(p_{\mathrm{gen}}) + \mathrm{KL}(p_{\mathrm{gen}} \| p_\theta)$ with the *Kullback-Leibler divergence* KL and the *distribution entropy* H. Thus a generator can lower its Gen. PPL by lowering its entropy instead of moving closer to the reference model.

For this reason, Gen. PPL is usually reported together with the *token entropy* $H_{\text{tok}}(p_{\text{gen}}) := \mathbb{E}_{\mathbf{x}\sim p_{\text{gen}}}\big[H(\hat{p}_{\mathbf{x}})\big]$, where $\hat{p}_{\mathbf{x}}$ denotes the empirical distribution of the tokens in $\mathbf{x}$ (Hu et al., 2026; Chen et al., 2026; Chemseddine et al., 2026). Both quantities are expectations of a score of a single sequence, the log-likelihood under the reference model and the entropy within the sequence. Neither of them looks at the distribution of the sequences, and p_{data} does not enter at all. In Figure 4, we replace a growing fraction of a text sample by copies of a single sequence, the token entropy remains unchanged and Gen. PPL even improves. Table 1 shows that Gen. PPL also favors the reference model itself, as GPT-2 Large scores better than the more powerful GPT-2 XL under a GPT-2 Large reference.

GPT-2	Gen. PPL (↓)
Small	12.5
Medium	11.4
Large	7.5
XL	8.5

Table 1: Gen. PPL across GPT-2 sizes for unconditional generation prefers GPT-2 Large.

3.2 Euclidean Latent Space Methods

Standard methods for comparing language distributions, such as MAUVE (Pillutla et al., 2021) and BERT-FID (Alihosseini et al., 2019), rely on a single latent token representation obtained from the final hidden layer of a transformer. Alternatively, the hidden states across positions can be aggregated by mean pooling. We briefly introduce both representations before turning to MAUVE.

3.2.1 Text Representation in the Euclidean Space

Last hidden-token representation. The *last hidden-token map* of a causal transformer in (1) reads as

$$\mathbf{h}^\theta : \mathcal{V}^{\le L} \to \mathbb{R}^d, \quad \mathbf{x} \mapsto h^\theta_{\mathbf{x}\le|\mathbf{x}|}, \tag{3}$$

where, in contrast to the whole sequence generator $\mathbf{H}^\theta$, we use $\mathbf{h}^\theta$ for the last-token representation. It allows to push the text distribution $p_{\text{data}} \in \mathcal{P}(\mathcal{V}^{\le L})$ to a distribution in Euclidean space $\mathbf{h}^\theta_\# p_{\text{data}} \in \mathcal{P}(\mathbb{R}^d)$. Under the assumption that language models are invertible, which is validated in the next remark, this distribution uniquely characterizes the language distribution.

Remark 3.1 (Language models are almost surely invertible). In (Nikolaou et al., 2026), it was proved that the last-token map is almost surely invertible. Empirically, the authors found no collisions across several autoregressive language models and large prompt collections. They further showed that, under certain assumptions, causal transformers are injective after gradient-based training.

Mean-pooling. In contrast to the above representation, *mean-pooling* takes into account the tokens at all positions:

$$\overline{\mathbf{H}}^\theta : V^{\le L} \to \mathbb{R}^d, \quad \mathbf{x} \mapsto \frac{1}{|\mathbf{x}|}\sum_{l=1}^{|\mathbf{x}|} h^\theta_{\mathbf{x}\le l}. \tag{4}$$

The associated distribution $\overline{\mathbf{H}}^\theta_\# p_{\text{data}} \in \mathcal{P}(\mathbb{R}^d)$ allows for standard methods such as MAUVE or FBD (BERT-FID). Mean pooling retains only the first moment of the token-cloud distribution and discards higher-order structure, so token clouds with the same mean but different spread or shape become indistinguishable, see Hara et al. (2026) and Appendix A.3. Moreover, it exhibits poor scaling behavior, i.e., discontinuities, under token pollution for MAUVE, see Appendix E.2.1.

3.2.2 MAUVE

MAUVE (Pillutla et al., 2021; 2023) is a sample-based score for comparing generated and reference text distributions. It embeds each text by a single latent representation, jointly quantizes both samples using k-means, and represents them by the resulting empirical cluster frequencies. MAUVE then summarizes the trade-off between $\mathrm{KL}(\tilde{p}_{\text{gen}}\|\tilde{r}_\lambda)$ and $\mathrm{KL}(\tilde{p}_{\text{data}}\|\tilde{r}_\lambda)$, where $\tilde{r}_\lambda = \lambda\tilde{p}_{\text{data}} +$

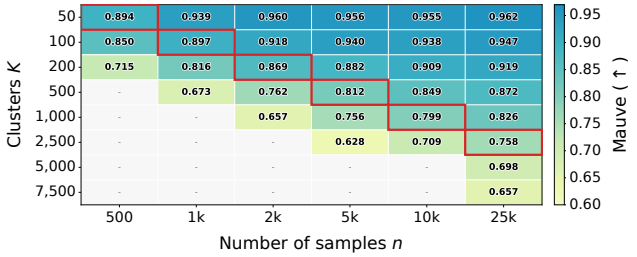


Figure 2: For a fixed diffusion language model and dataset, the reported MAUVE score depends on the number of evaluation samples n and clusters K. In particular, for the standard choice $K = n/10$ (red boxes), even after 5k samples it is not stable. Generation details are given in Appendix E.2.2.

$(1-\lambda)\tilde{p}_{\text{gen}}$, by the area under their so-called divergence frontier, see Appendix B.1 for a detailed algorithmic description.

The method has two drawbacks relevant to our goal. First, MAUVE is sensitive to the finite-sample quantization and can be unstable with respect to both the number of samples and clusters, see Figure 2. Second, its behavior depends strongly on the chosen text representation. In particular, MAUVE based on the last hidden state of a causal transformer is insensitive to perturbations at the beginning or middle of a sequence (He et al., 2023), which we reproduce in Figure 3. Mean pooling alleviates this particular problem but introduces new ones, see Appendix E.2.1.

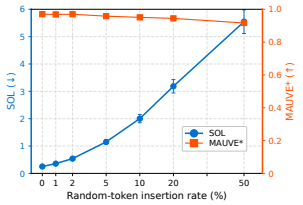


Figure 3: Replacing random tokens in the sequence beginning is not detected by MAUVE, but by SOL.

4 Evaluation of Language Models by Wasserstein Distances

In this section, we represent each sequence by the empirical measure of all its hidden states, so a text distribution becomes a measure on $\mathcal{P}_2(\mathbb{R}^d)$. Comparing such distributions by optimal transport leads to a numerically intractable Wasserstein-on-Wasserstein problem (Piening & Beinert, 2026b). We therefore recall sliced optimal transport as a tractable substitute and then define SOL in Section 4.3, showing that it satisfies the requirements (1)–(5) from the introduction.

4.1 Lifting a Sequence to a Probability Measure

The distribution of human-texts $p_{\text{data}} \in \mathcal{P}(\mathcal{V}^{\leq L})$ is a distribution over token sequences. While we could embed it into a meta-distribution via the canonical lift $\mathcal{V}^{\leq L} \to \mathcal{P}(\mathcal{V}^{\leq L})$, $\mathbf{x} \mapsto \delta_{\mathbf{x}}$, the set $\mathcal{V}^{\leq L}$ lacks a semantically meaningful distance. Instead, we propose to use the learned causal transformer $\mathbf{H}^\theta$, and introduce the associated *token-cloud distribution* by

$$T^\theta : \mathcal{V}^{\leq L} \to \mathcal{P}_2(\mathbb{R}^d), \qquad \mathbf{x} \mapsto \frac{1}{|\mathbf{x}|} \sum_{l=1}^{|\mathbf{x}|} \delta_{h^\theta_{\mathbf{x}\leq l}}. \tag{5}$$

This allows us to push forward $p_{\text{data}} \in \mathcal{P}(\mathcal{V}^{\leq L})$ to $T^\theta_\# p_{\text{data}} \in \mathcal{P}_2(\mathcal{P}_2(\mathbb{R}^d))$. This representation has two advantages. First, it compares sequences $\mathbf{x}_1$ and $\mathbf{x}_2$ of different lengths, $|\mathbf{x}_1| \neq |\mathbf{x}_2|$. Second, it allows reordered sentences such as '*cat on a mat*' and '*mat under a cat*' to remain close without imposing an artificial ordering. As shown in Section 4.3, the map remains injective due to the injectivity of the last hidden-state representation, where token representations encode their context. For a bidirectional encoder, we replace $h^\theta_{\mathbf{x}\leq l}$ by $h^{\theta,\text{bi}}_{\mathbf{x},l}$.

4.2 Double-Sliced Wasserstein Distance

In the following, we denote the Borel probability measures on a metric space (X, dist) by $\mathcal{P}(X)$, and the probability measures with finite second moments by $\mathcal{P}_2(X)$. Here we will not consider discrete spaces as $\mathcal{V}^{\leq L}$, but X will be either the Euclidean space $\mathbb{R}^d$ or the infinite dimensional space of (equivalence classes of) quadratic integrable functions $L_2([0,1])$.

Wasserstein distance. The *Wasserstein-2 distance* W on $\mathcal{P}_2(X)$ is defined by

$$\mathrm{W}^2(\mu,\nu) := \inf_{\pi \in \Pi(\mu,\nu)} \int_{X\times X} \text{dist}^2(x,y)\,\mathrm{d}\pi(x,y), \quad \forall \mu,\nu \in \mathcal{P}_2(X)$$

where $\Pi(\mu,\nu) \subset \mathcal{P}(X \times X)$ denotes set of measure with marginals μ and ν. For measures on the real line $\mu \in \mathcal{P}_2(\mathbb{R})$, using the isometric embedding

$$Q : \mathcal{P}_2(\mathbb{R}) \to L_2([0,1]), \qquad \mu \mapsto Q_\mu, \tag{6}$$

which maps a measure to its quantile function, the Wasserstein distance becomes

$$\mathrm{W}^2(\mu,\nu) = \int_0^1 |Q_\mu(s) - Q_\nu(s)|^2\,\mathrm{d}s, \qquad \mu,\nu \in \mathcal{P}_2(\mathbb{R}). \tag{7}$$

In one dimension, for empirical measures $\mu = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$, $x_i \in \mathbb{R}$, the computation of W reduces to the sorting of real numbers, which can be in $O(n \log n)$ operations.

Sliced Wasserstein distance. The idea behind sliced Wasserstein distances arises from the simple access to the problem in 1D. It is based on a *slicing operator* and its push-forward

$$S_\xi \colon \mathbb{R}^d \to \mathbb{R}, \quad x \mapsto \langle \xi, x \rangle, \qquad \mathbf{e}_\xi \colon \mathcal{P}_2(\mathbb{R}^d) \to \mathcal{P}_2(\mathbb{R}), \qquad \mu \mapsto S_{\xi,\#}\,\mu, \tag{8}$$

where $\xi \in \mathbb{S}^{d-1} := \{x \in \mathbb{R}^d \mid \|x\| = 1\}$. The *sliced Wasserstein-2 distance* reads as

$$\mathrm{SW}^2(\mu, \nu) := \int_{\mathbb{S}^{d-1}} \mathrm{W}^2(\mathbf{e}_\xi\,\mu, \mathbf{e}_\xi\,\nu)\,\mathrm{d}\mathcal{U}(\mathbb{S}^{d-1})(\xi),$$

where we integrate with respect to the uniform distribution $\mathcal{U}(\mathbb{S}^{d-1})$ on $\mathbb{S}^{d-1}$ and the Wasserstein distance on $\mathbb{R}$ is given by (7). Similar to the Wasserstein distance, SW metrizes the weak convergence, see (Bonnotte, 2013; Nadjahi et al., 2020). The spherical integral is usually approximated by Monte Carlo schemes, see (Bonneel et al., 2015; Nguyen et al., 2024; Hertrich et al., 2025).

Double-Sliced Wasserstein distance. Having lifted the data distribution from $p_{\text{data}} \in \mathcal{P}(\mathcal{V}^{\leq L})$ to $T^\theta_\# p_{\text{data}} \in \mathcal{P}_2(\mathcal{P}_2(\mathbb{R}^d))$, we follow (Piening & Beinert, 2026b) to construct the double-sliced Wasserstein distance for meta-measures $\boldsymbol{\mu} \in \mathcal{P}_2(\mathcal{P}_2(\mathbb{R}^d))$. Since $\mathcal{P}_2(\mathbb{R}^d)$ is not a Euclidean space, but, together with the Wasserstein distance W, an infinite-dimensional metric space, we have to define an appropriate slicing procedure.

First, using the slicing operation S_ξ from (8), we map measures in $\mathcal{P}(\mathbb{R}^d)$ to measures in $\mathcal{P}(\mathbb{R})$ using its push-forward operator $\mathbf{e}_\xi : \mathcal{P}_2(\mathbb{R}^d) \to \mathcal{P}_2(\mathbb{R})$.

Second, we apply the isometry $Q : \mathcal{P}_2(\mathbb{R}) \to L_2([0,1])$ in (6) to map

$$Q \circ \mathbf{e}_\xi : \mathcal{P}_2(\mathbb{R}^d) \to L_2([0,1]), \quad (Q \circ \mathbf{e}_\xi)_\# : \mathcal{P}_2(\mathcal{P}_2(\mathbb{R}^d)) \to \mathcal{P}_2(L_2([0,1])).$$

Third, we define a second slicing on $\mathcal{P}_2(L_2[0,1])$. Since this is a probability space of measures on the infinite dimensional Hilbert space $L_2([0,1])$, such slicing can no longer rely on the uniform distributions on $\mathbb{S}^{d-1}$. Instead, we slice functions on $L_2([0,1])$ with functions $g \in L_2([0,1])$ by

$$\mathcal{S}_g : L_2([0,1]) \to \mathbb{R}, \qquad f \mapsto \langle f, g \rangle_{L_2},$$
$$\mathcal{S}_{g,\#} : \mathcal{P}_2(L_2([0,1])) \to \mathcal{P}_2(\mathbb{R}), \qquad \eta \mapsto \mathcal{S}_{g,\#}\eta.$$

Then, instead of the uniform measure on the sphere, we choose a Gaussian reference measure $\gamma \in \mathcal{P}_2(L_2([0,1]))$, see Bogachev (1998), and define the *sliced Wasserstein distance* on $\mathcal{P}_2(L_2([0,1]))$ by

$$\mathrm{SW}^2(\eta, \zeta \mid \gamma) := \int_{L_2} \mathrm{W}^2(\mathcal{S}_{g,\#}\eta, \mathcal{S}_{g,\#}\zeta)\,\mathrm{d}\gamma(g).$$

To realize this sliced distance, i.e., the integral, we use the fact that there exists a one-to-one correspondence between Gaussian measures on $L_2([0,1])$ and Gaussian processes, see (Rajput & Cambanis, 1972, Thm 2). For details see (Piening & Beinert, 2026b).

Fourth, to combine the three steps, we introduce, for $\xi \in \mathbb{S}^{d-1}$ and $g \in L_2([0,1])$, the operator

$$M_{\xi,g} : \mathcal{P}_2(\mathbb{R}^d) \to \mathbb{R}, \qquad \mu \mapsto \left\langle Q_{\mathbf{e}_\xi \mu}, g \right\rangle_{L_2},$$
$$(M_{\xi,g})_\# : \mathcal{P}_2(\mathcal{P}_2(\mathbb{R}^d)) \to \mathcal{P}_2(\mathbb{R}), \qquad \boldsymbol{\mu} \mapsto (M_{\xi,g})_\# \boldsymbol{\mu}$$

Now we can define the *double sliced Wasserstein distance* on meta-measures by

$$\begin{aligned}\mathrm{DSW}^2(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma) :&= \int_{\mathbb{S}^{d-1}} \mathrm{SW}^2((Q \circ \mathbf{e}_\xi)_\# \boldsymbol{\mu}, (Q \circ \mathbf{e}_\xi)_\# \boldsymbol{\nu} \mid \gamma)\,\mathrm{d}\mathcal{U}(\mathbb{S}^{d-1})(\xi) \\ &= \mathbb{E}_{\xi \sim \mathcal{U}(\mathbb{S}^{d-1}), g \sim \gamma}\left[\mathrm{W}^2((M_{\xi,g})_\# \boldsymbol{\mu}, (M_{\xi,g})_\# \boldsymbol{\nu})\right].\end{aligned}$$

For empirical measures over empirical measures, DSW is a metric, see Piening & Beinert (2026b).

Proposition 4.1. *Let the assumptions of Propositions A.3 and A.5 hold, and assume in addition that the samples are independent of the slices. Then*

$$\mathbb{E}\left|\mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}_n, \boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma)\right| \leq \frac{\sqrt{8}\,R\sqrt{m_2(\gamma)}}{n^{1/4}} + 2R\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right)^{1/4}. \tag{9}$$

The proof of the sample complexity is given in Appendix A.2.

4.3 SOL - Sliced Optimal Transport for Language

We propose to apply the *double sliced Wasserstein distance for text comparison* after applying a reference causal transformer $\mathbf{H}^\theta$ and the resulting token-cloud distributions (5) of the data distribution p_{data} and the generated distribution p_{gen}, resulting in

$$\text{SOL}_\theta(p_{\text{data}}, p_{\text{gen}}) := \sqrt{d}\,\text{DSW}(T^\theta_\# p_{\text{data}}, T^\theta_\# p_{\text{gen}} \mid \gamma).$$

SOL is reweighted by the dimension of the hidden dimension. Based on the injectivity of the last hidden-token representation in Remark 3.1, we obtain the following proposition, where the proof is given in Appendix A.1.

Proposition 4.2. *Assume that* $\mathbf{h}^\theta$ *is injective and* $\gamma \in P_2(L_2(0,1))$ *has full support. For empirical distributions* $p, q \in \mathcal{P}(\mathcal{V}^{\leq L})$ *with uniform weights, we have* $\text{SOL}_\theta(p, q) = 0$ *if and only if* $p = q$.

5 Numerical Experiments

As we are interested in establishing a metric for small scale models and especially diffusion language models, we are doing our evaluation on a smaller scale. To establish the validity of SOL as a valid language metric, we first compare the quality against known language model properties, such as generation length, decoding algorithm, and model sizes, similar to Pillutla et al. (2021). Afterward, we show the numerical stability and favorable sampling complexity. In the end, we are evaluating our method on known failure cases for the other methods especially of MAUVE and do an evaluation of existing continuous and discrete language models on OpenWebText. For a detail comparison with MAUVE see Section E.2.3.

Setup. We are mostly interested in unconditional text generation and text continuations. As generative models can only approximate the distributions p_{data} that they have seen in training, for the reference distribution, a holdout set of the data distribution is used. The preprocessing of the reference distribution follows the one during training. The SOL metric is approximated with a finite number of inner (L_g) and outer slices (L_ξ) and a Gaussian process to sample g, see Appendix F.1. By default, we use $L_\xi = L_g = 1024$ and $n = 5{,}000$ samples. For Dream, we increase $L_\xi = L_g$ to 8192 to reduce estimator variance in its higher-dimensional hidden space ($d = 3584$, compared with $d = 1280$ for GPT-2 Large).

Embedding Models We evaluate different embedding backbones in T^θ for (5). Unless stated otherwise, we use GPT-2 Large as backbone embedding, matching the standard MAUVE setup and enabling direct comparison. We also evaluate more recent models such as OLMo2-1B (OLMo et al., 2025) and Qwen3-0.6B Base (Yang et al., 2025). Moreover, we consider Dream (Ye et al., 2025) which uses a bidirectional transformer. For further comparisons with contrastive embedding models see Appendix D.10

5.1 Generative Perplexity and Entropy are Blind to Mode Collapse

Generative perplexity (Gen. PPL) and average token entropy can fail to detect mode collapse. We split OpenWebText into two disjoint sets of 3000 sequences of 1024 tokens, using one as reference and progressively replacing samples in the other with copies of a single sequence. Throughout this collapse, token entropy remains nearly constant (5.437 ± 0.001), while Gen. PPL improves from 14.1 to 8.9. Thus, even when all samples are identical, these metrics suggest improved quality without reduced diversity. In contrast, SOL rises sharply (Figure 4).

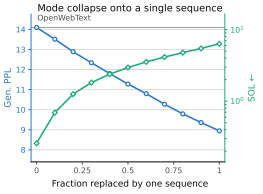


Figure 4: Mode collapse is invisible to Gen.PPL, while SOL rises sharply, see Subsection 5.1.

5.2 Does the SOL behave as expected?

To validate SOL, we validate if the method behaves as excepted in tests, where we know the outcome. First, we *increase the generation length* in the GPT-2 models and verify if the generated text degenerates accordingly. This holds true as can be seen Figure 5 left. Further, we can clearly verify that the model sizes are correctly ranked from most powerful (XL) to least powerful (small).

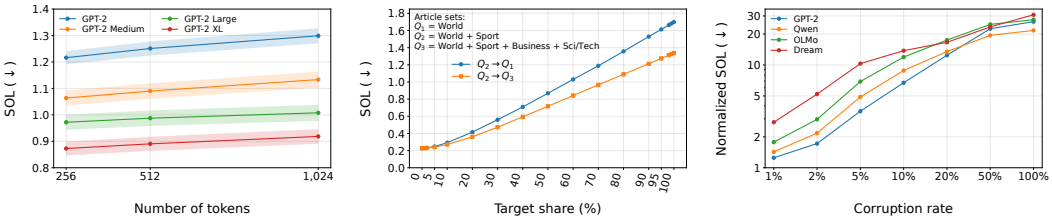


Figure 5: Left: *increasing the sequence length* diminishes the generation quality as expected. Middle: behavior under *mode-collapse*, i.e. going from Q_2 to Q_1 (blue), and if we introduce *additional modes*, i.e. going from Q_2 to Q_3 (orange). Right: SOL under *word shuffling* for different embeddings.

	MAUVE			SOL		
Criterion ↑	GPT-2 Large	OLMo2 1B	Dream 7B	GPT-2 Large	OLMo2 1B	Dream 7B
Human-like	0.857	0.881	0.881	0.881	0.929	**0.952**
Interesting	0.714	0.762	0.762	0.762	**0.786**	0.762
Sensible	0.762	0.786	0.786	0.786	0.881	**0.905**

Table 2: Human judgement comparison at sequence length 1024 for different embedding models. Worst-case Spearman rank correlation $\rho_{\min}$ between metric-induced model rankings and the Bradley–Terry (BT) human ranking. Results are shown for MAUVE and SOL for a full setup see Section C.4.

Second, based on (Bronnec et al., 2024), we evaluate SOL under both *mode collapse* and *support expansion*, using the AG news dataset (Zhang et al., 2015). Starting from a mixture of WORLD and SPORTS articles (Q_2), we progressively remove SPORTS articles (Q_1) or add articles from two new topics (Q_3). Both changes move the topic distribution away from the original. Therefore, we expect that SOL relative to the original mixture increases in both cases, as confirmed in Figure 5 (middle).

Third, we evaluate what happens if we *randomly shuffle* increasingly many tokens or *randomly pollute* the text with a fixed token. Increasing the percentage of shuffling or pollution continuously increases SOL, see Figure 5 right, resp. Figure 11. An extensive evaluation is given in Appendix D.6. In addition, in Appendix E.4, we show that SOL and MAUVE correctly classify greedy, ancestral, and nucleus sampling in unconditional and conditional generation. In Section D.5 we show that collapsing token entropy does not improve SOL.

Since causal embedding models often exhibit positional biases (He et al., 2023; Springer et al., 2025), we further examine the different embedding models with respect to perturbations at different positions within the sequence in Appendix D.7 and Appendix D.9. Here, we see that bi-directional attention weights all positions within a sequence almost equally, unlike causal embeddings.

5.3 Correlation with Human Judges

We follow the human-evaluation setup of MAUVE (Pillutla et al., 2021). The study compares eight GPT-2 model–decoding configurations using pairwise human judgments of whether continuations are *human-like*, *interesting*, and *sensible*. For each criterion, the pairwise preferences are converted into system scores by fitting a Bradley–Terry model (Bradley & Terry, 1952). We reuse these human judgments and recompute the automatic metrics on fresh samples from the same eight configurations. We report the worst-case Spearman rank correlation $\rho_{\min}$ of SOL with the Bradley–Terry scores for different embeddings as well as MAUVE. Full experimental details are given in Appendix C.4.

The comparison in Table 2 serves two purposes. First, we compare SOL and MAUVE using the same GPT-2 Large representation. Under this controlled setting, SOL achieves slightly higher correlation with the human ranking than MAUVE. Second, we investigate whether stronger embedding models further improve SOL. Replacing GPT-2 Large with more recent models leads to substantially higher correlations, indicating that the choice of embedding model has a strong impact on SOL's agreement with human judgments. Because the original human study displayed at most 256 tokens, we additionally evaluate a sequence length of 256 in Table 15, where SOL again outperforms MAUVE.

5.4 Approximation in SOL

Numerical Stability. We study its numerical approximation with respect to the number of slices L_ξ, L_g and samples n. Comparing p_{gen} using an AR model (Sahoo et al., 2024) with the true OpenWeb Text p_{data} (Gokaslan et al., 2019), we vary L_g and L_ξ independently, fixing the other and jointly with $L_g = L_\xi$. Figure 6 displays a clear numerical stability of our randomized estimator for $L_g = L_\xi = 1024$. Varying the sample size n (Fig. 7) shows clear convergence, with sample variability dominating estimator variability up to 5k samples. This displays more stability than MAUVE (Fig. 2). For Dream, this is done in Section D.1.

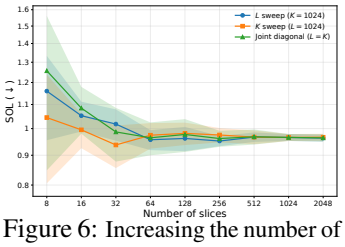


Figure 6: Increasing the number of slices stabilizes SOL.

Computational Cost. The main computational cost comes from sample and hidden-state generation. On an NVIDIA GeForce RTX 5090, computing the hidden states takes 71s for GPT-2 Large and 376s for Dream. Computing the DSW for 5000 samples takes about 1s with $L_\xi = L_g = 1024$ slices, see Table 5 for the runtime sweep.

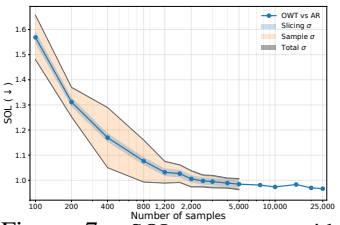


Figure 7: SOL converges with sample size, with std over slices and samples.

5.5 Diffusion Language Models on OWT

Diffusion language models, particularly continuous ones, remain difficult to evaluate. ELBO-based perplexities are unavailable for most continuous models, are not comparable across noise processes, and do not distinguish different samplers of the same discrete model. Gen. PPL is also problematic, as it can be improved by reducing the sample entropy (Franca & Tong, 2026). Since SOL only requires samples, it applies to all models and samplers alike. We evaluate the discrete diffusion models MDLM (Sahoo et al., 2024), DUO (Sahoo et al., 2025), LDDM (Jo et al., 2026), BD3-LM (Arriola et al., 2025), ADLM (Rout et al., 2025) and Soft-Masked (Hersche et al., 2026), and the continuous models vMF and VP (Chemseddine et al., 2026), LangFlow (Chen et al., 2026), FLM (Lee et al., 2026), FMLM* (Yoo et al., 2026) and ELF (Hu et al., 2026), together with an autoregressive baseline (Sahoo et al., 2024). ELF is trained on a differently tokenized corpus and is compared with its own reference (E). Full experimental details are given in Appendix C.5.

Table 3 shows that the sampler has a large influence, for the continuous model using SDE or PC and for the discrete ReMDM sampling. Soft-Masked outperforms AR. Continuous models are competitive with discrete ones from 2025, but removing self-conditioning degrade SOL. Further, Gen. PPL and SOL can disagree. LangFlow attains a much lower Gen. PPL than vMF with predictor-corrector sampling (36.9 vs. 56.3) at a lower entropy, while both reach a comparable SOL. Results for three embeddings, further sampler settings and a diagnosis of MAUVE's last-token bias are given in Appendix E.3.

Model	Setting	H	Gen. ppl.	SOL ↓
Real OWT^{G}	—	5.449	14.9	2.066
Real ELF OWT^{E}	—	5.402	17.0	2.649
Autoregressive				
AR^{G}	Temperature 1.0	5.582	34.9	22.207
Discrete diffusion				
DUO-base^{G}	Ancestral, $T = 1024$	5.537	76.9	32.642
DUO-distilled^{G}	Ancestral, $T = 32$	5.518	62.3	30.595
MDLM^{G}	Exact, $T = 1000$	5.445	59.4	36.394
LDDM-M^{G}	$T = 1024$	5.536	48.2	24.601
Soft-Masked^{G}	ReMDM, $T = 1024$	5.341	16.7	**18.966**
ADLM-large^{G}	ReMDM, $T = 1024$	5.352	28.9	23.601
BD3-LM, block 4^{B}	First-hitting, $T = 5000$	5.336	24.4	26.070
Continuous diffusion and flow				
vMF^{G}	ODE, steps= 1024	5.456	75.4	33.832
vMF^{G}	SDE, steps= 128	5.348	43.6	30.375
VP^{G}	ODE, steps= 1024	5.435	66.1	33.275
VP^{G}	SDE, steps= 128	5.355	50.9	31.972
LangFlow^{G}	ODE, steps= 1024	5.257	36.9	30.541
ELF-B^{E}	SDE, steps= 128	5.123	21.9	32.878
ELF-B^{E}	ODE, steps= 1024	5.090	24.0	33.320
Few-step models				
FMLM^{G}	2 steps	5.295	135.1	71.134
FLM^{G}	Euler, steps= 1024	5.345	63.1	48.387
ELF*^{G}	CDEQ, steps= 128, CFG= 5	5.384	60.8	35.950
FMLM*^{G}	2 steps, CFG= 5	5.458	97.4	41.120
FMLM*^{G}	128 steps, CFG= 5	5.456	79.0	38.482

Table 3: OWT baselines using 5,000 samples per distribution and a 1,024-token horizon, see for the full Table 22. SOL is using Dream-7B.

6 Conclusion

Current evaluation methods are insufficient for comparing modern language models. We propose SOL, a sample-based distance between text distributions that represents each sequence by the empirical measure of its hidden states and compares the resulting distributions using the double sliced Wasserstein distance. SOL is a metric for injective reference transformers and admits a stable estimator dominated by a single reference-model forward pass. Unlike Gen. PPL and entropy, SOL detects mode collapse, unlike MAUVE, it is less biased toward the end of the sequence and correlates better with human judgments. Because SOL only requires samples, it compares autoregressive, discrete, and continuous diffusion models on equal footing. On OpenWebText, we find that better Gen. PPL does not necessarily imply a better fit to the data distribution. We recommend 5,000 samples, $L_\xi = L_g = 8192$ slices, and the bidirectional DREAM backbone.

AI use disclosure. In this work, we used generative AI tools to assist with writing and editing code, formulating and refining mathematical claims, and writing mathematical proofs. We additionally used generative AI tools to draft and polish parts of the manuscript for clarity and readability. All AI-assisted content was manually reviewed by the authors. Mathematical claims, arguments, and proofs were checked for correctness, generated or edited code was inspected and tested, and manuscript text was verified against the underlying results. We take responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

REFERENCES

Danial Alihosseini, Ehsan Montahaei, and Mahdieh Soleymani Baghshah. Jointly measuring diversity and quality in text generation models. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pp. 90–98. Association for Computational Linguistics, 2019.

David Alvarez-Melis and Nicolo Fusi. Geometric dataset distances via optimal transport. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21428–21439. Curran Associates, 2020.

Marianne Arriola, Aaron Gokaslan, Justin T Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2503.09573.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

Quentin Berthet, Yu-Han Wu, Clement Crepy, Romuald Elie, Klaus Greff, and Michael Eli Sander. MIND: Monge Inception Distance for generative models evaluation. *arXiv preprint arXiv:2605.06797*, 2026.

Vladimir Igorevich Bogachev. *Gaussian measures*. Number 62. American Mathematical Soc., 1998.

Clément Bonet, Christophe Vauthier, and Anna Korba. Flowing datasets with Wasserstein over Wasserstein gradient flows. In *Proceedings of ICML'25*. OpenReview.net, 2025.

Clément Bonet, Elsa Cazelles, Lucas Drumetz, and Nicolas Courty. Busemann functions in the Wasserstein space: Existence, closed-forms, and applications to slicing. 300:1972–1980, 2026.

Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and Radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45, 2015.

Nicolas Bonnotte. *Unidimensional and Evolution Methods for Optimal Transportation*. PhD thesis, Université Paris Sud–Paris XI, Orsay, France, 2013. PhD Thesis.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444, 14643510. URL http://www.jstor.org/stable/2334029.

Lola Le Breton, Quentin Fournier, Mariam El Mezouar, John X. Morris, and Sarath Chandar. Neobert: A next-generation bert, 2025. URL https://arxiv.org/abs/2502.19587.

Florian Le Bronnec, Alexandre Verine, Benjamin Negrevergne, Yann Chevaleyre, and Alexandre Allauzen. Exploring precision and recall to assess the quality and diversity of llms, 2024. URL https://arxiv.org/abs/2402.10693.

Ciprian Chelba, Tomás Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, and Phillipp Koehn. One billion word benchmark for measuring progress in statistical language modeling. *CoRR*, abs/1312.3005, 2013. URL http://arxiv.org/abs/1312.3005.

J Chemseddine, G Kornhardt, and G Steidl. Spherical flows for sampling categorical data. *arXiv preprint arXiv:2605.05629*, 2026.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

Yuxin Chen, Chumeng Liang, Hangke Sui, Ruihan Guo, Chaoran Cheng, Jiaxuan You, and Ge Liu. Langflow: Continuous diffusion rivals discrete in language modeling, 2026. URL https://arxiv.org/abs/2604.11748.

Min Jin Chong and David Forsyth. Effectively unbiased FID and inception score and where to find them. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6070–6079. IEEE, 2020.

Antonin Clerc, Michael Quellmalz, Moritz Piening, Philipp Flotho, Gregor Kornhardt, and Gabriele Steidl. Hot-pot: Optimal transport for sparse stereo matching. *Journal of Mathematical Imaging and Vision*, 68(4):53, Aug 2026. ISSN 1573-7683. doi: 10.1007/s10851-026-01336-3. URL https://doi.org/10.1007/s10851-026-01336-3.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H. Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, Curtis Hawthorne, Rémi Leblond, Will Grathwohl, and Jonas Adler. Continuous diffusion for categorical data, 2022. URL https://arxiv.org/abs/2211.15089.

Yonatan Dukler, Wuchen Li, Alex Lin, and Guido Montufar. Wasserstein of Wasserstein loss for learning generative models. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1716–1725. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/dukler19a.html.

Ahmed Ewais, Ahmed Hashish, and Amr Ali. Just pass twice: Efficient token classification with llms for zero-shot ner. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11480–11497. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.526. URL http://dx.doi.org/10.18653/v1/2026.acl-long.526.

Antonio Franca and Alexander Tong. Hacking generative perplexity: Why unconditional text evaluation needs distributional metrics. *arXiv preprint arXiv:2606.08417*, 2026.

Jonathan Geuter, Gregor Kornhardt, Ingimar Tomasson, and Vaios Laschos. Universal neural optimal transport. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19196–19232. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/geuter25a.html.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

Samson Gourevitch, Yazid Janati, Dario Shariatian, Umut Simsekli, Eric Moulines, Eric P. Xing, and Alain Durmus. Uniform diffusion models revisited: Leave-one-out denoiser and absorbing state reformulation, 2026. URL https://arxiv.org/abs/2605.22765.

Ishaan Gulrajani and Tatsunori B. Hashimoto. Likelihood-based diffusion language models, 2023. URL https://arxiv.org/abs/2305.18619.

Tomomasa Hara, Hiroto Kurita, Masaaki Imaizumi, Kentaro Inui, and Sho Yokoi. Why mean pooling works: Quantifying second-order collapse in text embeddings. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 47180–47201. Association for Computational Linguistics, 2026.

Doron Haviv, Aram-Alexandre Pooladian, Dana Pe'er, and Brandon Amos. Wasserstein flow matching: Generative modeling over families of distributions. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=MRmI68k3gd.

Tianxing He, Jingyu Zhang, Tianle Wang, Sachin Kumar, Kyunghyun Cho, James Glass, and Yulia Tsvetkov. On the blind spots of model-based evaluation metrics for text generation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12067–12097, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.674. URL https://aclanthology.org/2023.acl-long.674/.

Michael Hersche, Samuel Moor-Smith, Thomas Hofmann, and Abbas Rahimi. Soft-masked diffusion language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Gba02UMvrG.

Johannes Hertrich, Tim Jahn, and Michael Quellmalz. Fast summation of radial kernels via QMC slicing. In *Proceedings of the ICLR'25*. OpenReview.net, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, 2017.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Keya Hu, Linlu Qiu, Yiyang Lu, Hanhong Zhao, Tianhong Li, Yoon Kim, Jacob Andreas, and Kaiming He. Elf: Embedded language flows, 2026. URL https://arxiv.org/abs/2605.10938.

Mingyu Jo, Jaesik Yoon, Justin Deschenaux, Caglar Gulcehre, and Sungjin Ahn. Loopholing discrete diffusion: Deterministic bypass of the sampling wall. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=wM3Mdk2Pa5.

Wolfgang Kreitmeier. Optimal vector quantization in terms of Wasserstein distance. *Journal of Multivariate Analysis*, 102(8):1225–1239, 2011. doi: 10.1016/j.jmva.2011.04.005.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 957–966. PMLR, 2015.

Chanhyuk Lee, Jaehoon Yoo, Manan Agarwal, Sheel Shah, Jerry Huang, Aditi Raghunathan, Seunghoon Hong, Nicholas M. Boffi, and Jinwoo Kim. Flow map language models: One-step language modeling via continuous denoising, 2026. URL https://arxiv.org/abs/2602.16813.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-lm improves controllable text generation, 2022. URL https://arxiv.org/abs/2205.14217.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023. URL https://arxiv.org/abs/2308.03281.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution, 2024. URL https://arxiv.org/abs/2310.16834.

Kimia Nadjahi, Alain Durmus, Lénaïc Chizat, Soheil Kolouri, Shahin Shahrampour, and Umut Simsekli. Statistical and topological properties of sliced probability divergences. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15657–15669. Curran Associates, 2020.

Khai Nguyen and Peter Mueller. Summarizing nonparametric Bayesian mixture posteriors—sliced optimal transport metrics for Gaussian mixtures. *Journal of Computational and Graphical Statistics*, 2026. doi: 10.1080/10618600.2026.2634831.

Khai Nguyen, Nicola Bariletto, and Nhat Ho. Quasi-Monte Carlo for 3d sliced Wasserstein. In *Proceedings of the ICLR'24*. OpenReview.net, 2024.

Khai Nguyen, Hai Nguyen, Tuan Pham, and Nhat Ho. Lightspeed geometric dataset distance via sliced optimal transport. In *Proceedings of ICML'25*. OpenReview.net, 2025.

Giorgos Nikolaou, Tommaso Mencattini, Donato Crisostomi, Andrea Santilli, Yannis Panagakis, and Emanuele Rodolà. Language models are injective and hence invertible. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0kHbD6ad07.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious, 2025. URL https://arxiv.org/abs/2501.00656.

OpenAI. GPT-2 Output Dataset. https://github.com/openai/gpt-2-output-dataset, 2019. GitHub repository.

M Piening, R Duong, and G Steidl. Generalized Wasserstein flow matching: Transport plans, everywhere, all at once, 2026.

Moritz Piening and Robert Beinert. Slicing the Gaussian mixture Wasserstein distance. *Transactions on Machine Learning Research*, 2025.

Moritz Piening and Robert Beinert. A novel sliced fused gromov-wasserstein distance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24828–24836, 2026a.

Moritz Piening and Robert Beinert. Slicing Wasserstein over Wasserstein via functional optimal transport. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=l3KtyVZde3.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. MAUVE: Measuring the gap between neural text and human text using divergence frontiers. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=Tqx7nJp7PR.

Krishna Pillutla, Lang Liu, John Thickstun, Sean Welleck, Swabha Swayamdipta, Rowan Zellers, Sewoong Oh, Yejin Choi, and Zaid Harchaoui. MAUVE scores for generative models: Theory and practice. *Journal of Machine Learning Research*, 24(356):1–92, 2023.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Balram S Rajput and Stamatis Cambanis. Gaussian processes and Gaussian measures. *The Annals of Mathematical Statistics*, pp. 1944–1952, 1972.

Litu Rout, Constantine Caramanis, and Sanjay Shakkottai. Anchored diffusion language model. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=E8adS5srds.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models, 2024. URL https://arxiv.org/abs/2406.07524.

Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin T Chiu, and Volodymyr Kuleshov. The diffusion duality. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=9P9Y8FOSOk.

Subham Sekhar Sahoo, Jean-Marie Lemercier, Zhihan Yang, Justin Deschenaux, Jingyu Liu, John Thickstun, and Ante Jukić. Scaling beyond masked diffusion language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=xKJG1CNtQM.

Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In *Advances in Neural Information Processing Systems*, volume 31, pp. 5228–5237. Curran Associates, 2018.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data, 2025. URL https://arxiv.org/abs/2406.04329.

Jacob Mitchell Springer, Suhas Kotha, Daniel Fried, Graham Neubig, and Aditi Raghunathan. Repetition improves language model embeddings. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Ahlrf2HGJR.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training, 2024. URL https://arxiv.org/abs/2212.03533.

Jiannan Xiang, Yahui Liu, Deng Cai, Huayang Li, Defu Lian, and Lemao Liu. Assessing dialogue systems with distribution distances. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 2192–2198. Association for Computational Linguistics, 2021.

Toshiaki Yachimura, Hanbo Wang, Yusuke Imoto, Momoko Yoshida, Sohei Tasaki, Yoji Kojima, Yukihiro Yabuta, Mitinori Saitou, and Yasuaki Hiraoka. scegot: single-cell trajectory inference framework based on entropic Gaussian mixture optimal transport. *BMC bioinformatics*, 25(1): 388, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models, 2025. URL https://arxiv.org/abs/2508.15487.

Jaehoon Yoo, Wonjung Kim, Floor Eijkelboom, Chanhyuk Lee, Nicholas M. Boffi, Seunghoon Hong, and Jinwoo Kim. Self-conditioned flow map language models via fixed-point flows, 2026. URL https://arxiv.org/abs/2607.00714.

Mikhail Yurochkin, Sebastian Claici, Edward Chien, Farzaneh Mirzazadeh, and Justin M. Solomon. Hierarchical optimal transport for document representation. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, 2019.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *Proceedings of the ICLR'20*. OpenReview.net, 2020.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 563–578. Association for Computational Linguistics, 2019.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *Proceedings of the ICLR'25*. OpenReview.net, 2025.

CONTENTS

1 Introduction **1**

2 Preliminaries **2**

3 Current Evaluation Methods for Language Models **3**

3.1 Perplexity-Based Methods 3

3.2 Euclidean Latent Space Methods 4

3.2.1 Text Representation in the Euclidean Space 4

3.2.2 MAUVE 4

4 Evaluation of Language Models by Wasserstein Distances **5**

4.1 Lifting a Sequence to a Probability Measure 5

4.2 Double-Sliced Wasserstein Distance 5

4.3 SOL - Sliced Optimal Transport for Language 7

5 Numerical Experiments **7**

5.1 Generative Perplexity and Entropy are Blind to Mode Collapse 7

5.2 Does the SOL behave as expected? 7

5.3 Correlation with Human Judges 8

5.4 Approximation in SOL 8

5.5 Diffusion Language Models on OWT 9

6 Conclusion **9**

A Proofs and Theoretical Details **18**

A.1 SOL is a Metric 18

A.2 Approximation of the Double-Sliced Wasserstein Distance 18

A.3 Double-Sliced Wasserstein compared vs Mean Hidden Token Sliced Wasserstein . 23

A.4 Approximation of SOL 23

B Extended Related Work **25**

B.1 Detailed Description of MAUVE 25

B.2 Overview of Related Work 25

B.2.1 Comparing Text Representations via Latent Geometries 25

B.2.2 Optimal Transport 26

C Experimental Setup and Implementation **27**

C.1 Datasets and Preprocessing 27

C.2 Encoding Models 27

C.3 SOL Estimator and Hyperparameters 27

C.4 Correlation with Human Judgments 27
C.5 OpenWebText Benchmark Configurations 28
C.6 Runtime 31

D Additional Ablations **32**
D.1 Number of Slices and Samples 32
D.2 Quantile Resolution 32
D.3 Gaussian Process Bandwidth 32
D.4 Mean-SW versus SOL under Structural Perturbations 33
D.5 Sensitivity to Sampling Temperature 33
D.6 Pollution and Corruption 33
D.7 Positional Sensitivity 34
D.7.1 Positional bias of last-token SW 35
D.8 Mode Coverage 35
D.9 Alternative to Bidirectional Attention 35
D.10 Text Embedding Models 36

E Additional Evaluation Results **38**
E.1 Human-Judgment Results 38
E.2 MAUVE Comparison 38
E.2.1 Mean Hidden Representation 38
E.2.2 LM1B Experiment 38
E.2.3 Comparing MAUVE with SOL 38
E.3 OpenWebText complete Model Table 39
E.3.1 Mauve Depends on Last Hidden Tokens 39
E.3.2 Extended Table 40
E.4 Additional Sampler 41

F Implementation Details **42**
F.1 Numerical implementation of SOL 42

A Proofs and Theoretical Details

A.1 SOL is a Metric

Proposition A.1. *Assume that $\mathbf{h}^\theta$ is injective and $\gamma \in P_2(L_2(0,1))$ has full support. For empirical distributions $p, q \in \mathcal{P}(\mathcal{V}^{\leq L})$ with uniform weights, we have $\mathrm{SOL}_\theta(p,q) = 0$ if and only if $p = q$.*

Proof. We will show that the injectivity of the last hidden-token representation implies injectivity of T^θ. Then its push forward $T^\theta_\#$ is injective as well and the assertion follows from the fact that DSW is a metric on empirical meta-measures, see Piening & Beinert (2026b).

It remains to prove that T^θ is indeed injective. Let $\mathbf{x}, \mathbf{y} \in \mathcal{V}^{\leq L}$ satisfy $T^\theta(\mathbf{x}) = T^\theta(\mathbf{y})$. By the injectivity of the last hidden-token representation it follows that the components of $(h^\theta_{\mathbf{x}\leq 1}, \dots, h^\theta_{\mathbf{x}\leq|\mathbf{x}|})$ are pairwise distinct and similarly for $\mathbf{y}$. By $T^\theta(\mathbf{x}) = T^\theta(\mathbf{y})$, we thus have that $|\mathbf{x}| = |\mathbf{y}| = n$. Further, it follows that there exits $j \leq n$ such that $h^\theta_{\mathbf{x}\leq n} = h^\theta_{\mathbf{y}\leq j}$. By injectivity of the last hidden-token representation this implies that $\mathbf{x}_{\leq n} = \mathbf{x} = \mathbf{y}_{\leq j}$, so that $j = n$ and we are done. □

A.2 Approximation of the Double-Sliced Wasserstein Distance

Throughout this section, let $X \subset B_R(0) \subset \mathbb{R}^d$ be nonempty and Borel for some $R > 0$. We equip $\mathcal{P}_2(X)$ with the Wasserstein distance W. On the sphere, we use the uniform measure $\sigma := \mathcal{U}(\mathbb{S}^{d-1})$. On $L_2([0,1])$, we fix a measure $\gamma \in \mathcal{P}_2(L_2([0,1]))$ and denote its moments by

$$m_r(\gamma) := \int \|g\|_{L_2}^r \,\mathrm{d}\gamma(g), \qquad r \in \{2,4\}.$$

The second moment is finite by assumption, while $m_4(\gamma) = \infty$ is allowed.

Recall from the main text that each pair of slices (ξ, g) defines the map

$$M_{\xi,g} \colon \mathcal{P}_2(X) \to \mathbb{R}, \qquad M_{\xi,g}(\mu) = \langle Q_{\mathbf{e}_\xi \mu}, g\rangle_{L_2},$$

where $\mathbf{e}_\xi \mu = (S_\xi)_\# \mu$ is the projection of μ under $S_\xi(x) := \langle \xi, x\rangle$. For $\boldsymbol{\lambda}, \boldsymbol{\eta} \in \mathcal{P}_2(\mathcal{P}_2(X))$, the corresponding one-dimensional cost is

$$c_{\boldsymbol{\lambda},\boldsymbol{\eta}}(\xi, g) := \mathrm{W}^2\big((M_{\xi,g})_\# \boldsymbol{\lambda}, (M_{\xi,g})_\# \boldsymbol{\eta}\big),$$

and the double-sliced Wasserstein distance reads

$$\mathrm{DSW}(\boldsymbol{\lambda}, \boldsymbol{\eta} \mid \gamma) = \left(\int c_{\boldsymbol{\lambda},\boldsymbol{\eta}} \,\mathrm{d}(\sigma \otimes \gamma)\right)^{1/2}.$$

In practice, the slicing measures are replaced by empirical measures. For $L_\xi, L_g \geq 1$ and slices $\xi_i \in \mathbb{S}^{d-1}$ and $g_j \in L_2([0,1])$, we set

$$\sigma_{L_\xi} := \frac{1}{L_\xi} \sum_{i=1}^{L_\xi} \delta_{\xi_i}, \qquad \gamma_{L_g} := \frac{1}{L_g} \sum_{j=1}^{L_g} \delta_{g_j}. \tag{10}$$

The finite-slice approximation is then given by

$$\mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\lambda}, \boldsymbol{\eta} \mid \gamma) := \left(\int c_{\boldsymbol{\lambda},\boldsymbol{\eta}} \,\mathrm{d}(\sigma_{L_\xi} \otimes \gamma_{L_g})\right)^{1/2}.$$

We bound the error that arises when both the slicing measures and the measures $\boldsymbol{\lambda}, \boldsymbol{\eta}$ are replaced by empirical measures. This is done in two steps. Proposition A.3 treats the slicing error for fixed measures, and Proposition A.5 treats the sampling error for fixed slices. Proposition A.6 combines both. All three results rely on the regularity properties collected in the following lemma.

Lemma A.2. *Let $\xi, \xi' \in \mathbb{S}^{d-1}$, $g, h \in L_2([0,1])$, and $\mu, \nu \in \mathcal{P}_2(X)$. Then*

$$|M_{\xi,g}(\mu) - M_{\xi',h}(\nu)| \leq \|g\|_{L_2}\big(W(\mu,\nu) + R\|\xi - \xi'\|_2\big) + R\|g - h\|_{L_2}. \tag{11}$$

In particular, the map $(\xi, g, \mu) \mapsto M_{\xi,g}(\mu)$ is jointly continuous, and it holds $|M_{\xi,g}(\mu)| \le R\|g\|_{L_2}$. Moreover, for every $\boldsymbol{\rho} \in \mathcal{P}_2(\mathcal{P}_2(X))$, the pushforwards satisfy

$$W\big((M_{\xi,g})_{\#}\boldsymbol{\rho}, (M_{\xi',h})_{\#}\boldsymbol{\rho}\big) \le R\|g\|_{L_2}\|\xi - \xi'\|_2 + R\|g - h\|_{L_2}. \tag{12}$$

Finally, the function $c_{\boldsymbol{\lambda},\boldsymbol{\eta}}$ is continuous with

$$0 \le c_{\boldsymbol{\lambda},\boldsymbol{\eta}}(\xi, g) \le 4R^2\|g\|_{L_2}^2. \tag{13}$$

Proof. Estimates for the quantile functions. Set $q_{\xi,\mu} := Q_{\mathbf{e}_\xi \mu}$. The quantile representation of W on $\mathbb{R}$ and the triangle inequality in $L_2([0,1])$ give

$$\begin{aligned}\|q_{\xi,\mu} - q_{\xi',\nu}\|_{L_2} &\le \|q_{\xi,\mu} - q_{\xi,\nu}\|_{L_2} + \|q_{\xi,\nu} - q_{\xi',\nu}\|_{L_2} \\ &= \mathrm{W}\big((S_\xi)_{\#}\mu, (S_\xi)_{\#}\nu\big) + \mathrm{W}\big((S_\xi)_{\#}\nu, (S_{\xi'})_{\#}\nu\big).\end{aligned}$$

We bound the two terms separately. Since S_ξ is 1-Lipschitz, pushing an optimal coupling of μ and ν forward under (S_ξ, S_ξ) yields

$$\mathrm{W}\big((S_\xi)_{\#}\mu, (S_\xi)_{\#}\nu\big) \le \mathrm{W}(\mu, \nu).$$

For the second term, we use the coupling $(S_\xi, S_{\xi'})_{\#}\nu$. Since $\|x\|_2 \le R$ on X, this gives

$$\mathrm{W}\big((S_\xi)_{\#}\nu, (S_{\xi'})_{\#}\nu\big) \le \left(\int_X |\langle \xi - \xi', x\rangle|^2 \,\mathrm{d}\nu(x)\right)^{1/2} \le R\|\xi - \xi'\|_2.$$

In the same way, the norm of each quantile function is bounded by

$$\|q_{\xi,\mu}\|_{L_2}^2 = \int_X |\langle \xi, x\rangle|^2 \,\mathrm{d}\mu(x) \le R^2.$$

Continuity of M. We decompose

$$M_{\xi,g}(\mu) - M_{\xi',h}(\nu) = \langle q_{\xi,\mu} - q_{\xi',\nu}, g\rangle_{L_2} + \langle q_{\xi',\nu}, g - h\rangle_{L_2}.$$

Applying Cauchy–Schwarz to both terms and inserting the estimates above proves (11). The joint continuity and the bound on $|M_{\xi,g}(\mu)|$ follow directly.

Continuity of the pushforwards. Let $\boldsymbol{\rho} \in \mathcal{P}_2(\mathcal{P}_2(X))$. Since $M_{\xi,g}$ and $M_{\xi',h}$ are continuous, the measure $(M_{\xi,g}, M_{\xi',h})_{\#}\boldsymbol{\rho}$ is a coupling of the two pushforwards. Using this coupling in the definition of W gives

$$\mathrm{W}^2\big((M_{\xi,g})_{\#}\boldsymbol{\rho}, (M_{\xi',h})_{\#}\boldsymbol{\rho}\big) \le \int |M_{\xi,g}(\mu) - M_{\xi',h}(\mu)|^2 \,\mathrm{d}\boldsymbol{\rho}(\mu).$$

By (11) with $\nu = \mu$, the integrand is bounded by the square of the right-hand side of (12). This proves (12).

Properties of c. The triangle inequality for W gives

$$|\mathrm{W}(a, b) - \mathrm{W}(a', b')| \le \mathrm{W}(a, a') + \mathrm{W}(b, b')$$

for all $a, b, a', b' \in \mathcal{P}_2(\mathbb{R})$. Together with (12) for $\boldsymbol{\rho} = \boldsymbol{\lambda}$ and $\boldsymbol{\rho} = \boldsymbol{\eta}$, this shows that $\sqrt{c_{\boldsymbol{\lambda},\boldsymbol{\eta}}}$ is continuous, and hence so is $c_{\boldsymbol{\lambda},\boldsymbol{\eta}}$. Finally, both slice laws in the definition of $c_{\boldsymbol{\lambda},\boldsymbol{\eta}}(\xi, g)$ are supported in the interval $[-R\|g\|_{L_2}, R\|g\|_{L_2}]$. Their squared Wasserstein distance is therefore at most the squared length of this interval, which proves (13). □

Approximation of the integrals. We first keep $\boldsymbol{\mu}, \boldsymbol{\nu}$ fixed and approximate only the slicing measures σ and γ.

Proposition A.3. *Let $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathcal{P}_2(\mathcal{P}_2(X))$ and assume $m_4(\gamma) < \infty$. Let the slices $\xi_1, \dots, \xi_{L_\xi}$ be iid with law σ, let $g_1, \dots, g_{L_g}$ be iid with law γ, and let both families be independent. Then*

$$\mathbb{E}_s\left|\mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma)\right| \le 2R\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right)^{1/4}, \tag{14}$$

where $\mathbb{E}_s$ denotes the expectation over the slices.

Proof. Decomposition. Write $c := c_{\boldsymbol{\mu},\boldsymbol{\nu}}$ and denote the exact and the approximate integral by

$$I := \int c \, \mathrm{d}(\sigma \otimes \gamma), \qquad \widehat{I} := \int c \, \mathrm{d}(\sigma_{L_\xi} \otimes \gamma_{L_g}).$$

We replace the two slicing measures one after the other and split $\widehat{I} - I = A + B$, where

$$A := \int c \, \mathrm{d}\big((\sigma_{L_\xi} - \sigma) \otimes \gamma\big), \qquad B := \int c \, \mathrm{d}\big(\sigma_{L_\xi} \otimes (\gamma_{L_g} - \gamma)\big).$$

The term A replaces σ by σ_{L_ξ}. The term B then replaces γ by γ_{L_g}.

A variance identity. Let ρ be a probability measure, and let ρ_m be its empirical approximation based on m iid samples. For every $f \in L_2(\rho)$, expanding the square gives

$$\mathbb{E}\left|\int f \, \mathrm{d}(\rho_m - \rho)\right|^2 = \frac{1}{m}\left(\int f^2 \, \mathrm{d}\rho - \left(\int f \, \mathrm{d}\rho\right)^2\right) \le \frac{1}{m}\int f^2 \, \mathrm{d}\rho. \tag{15}$$

Bounding A and B. For A, we apply (15) to σ_{L_ξ} with the function

$$\xi \mapsto \int c(\xi, g) \, \mathrm{d}\gamma(g).$$

By (13), this function is bounded by $4R^2 m_2(\gamma)$. Hence

$$\mathbb{E}_s A^2 \le \frac{1}{L_\xi}\int \left(\int c(\xi, g) \, \mathrm{d}\gamma(g)\right)^2 \mathrm{d}\sigma(\xi) \le \frac{16R^4 m_2(\gamma)^2}{L_\xi}.$$

For B, we fix $\xi_1, \dots, \xi_{L_\xi}$ and apply (15) to γ_{L_g} with the function

$$g \mapsto \int c(\xi, g) \, \mathrm{d}\sigma_{L_\xi}(\xi).$$

By (13), this function is bounded by $4R^2\|g\|_{L_2}^2$. Taking the expectation over the ξ_i afterwards gives

$$\mathbb{E}_s B^2 \le \frac{1}{L_g}\mathbb{E}_s \int \left(\int c(\xi, g) \, \mathrm{d}\sigma_{L_\xi}(\xi)\right)^2 \mathrm{d}\gamma(g) \le \frac{16R^4 m_4(\gamma)}{L_g}.$$

This is the only place where the fourth moment enters.

The cross term. For every fixed collection of ξ_i, the expectation of B with respect to $g_1, \dots, g_{L_g}$ vanishes. Since A depends only on the ξ_i, it follows that $\mathbb{E}_s[AB] = 0$. Hence

$$\mathbb{E}_s|\widehat{I} - I|^2 = \mathbb{E}_s A^2 + \mathbb{E}_s B^2 \le 16R^4\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right).$$

Taking square roots. For $a, b \ge 0$, it holds $|\sqrt{a} - \sqrt{b}| \le |a - b|^{1/2}$. Together with Jensen's inequality for the concave map $t \mapsto t^{1/4}$, we obtain

$$\mathbb{E}_s|\sqrt{\widehat{I}} - \sqrt{I}| \le \mathbb{E}_s|\widehat{I} - I|^{1/2} \le \big(\mathbb{E}_s|\widehat{I} - I|^2\big)^{1/4}.$$

Inserting the previous bound proves the assertion. □

Approximation of the measures. We next keep the slices fixed and approximate the measures $\boldsymbol{\mu}, \boldsymbol{\nu}$. The key ingredient is the following estimate on $\mathbb{R}$.

Lemma A.4. *Let $\alpha \in \mathcal{P}_2(\mathbb{R})$ be supported in $[-B, B]$ for some $B \ge 0$, and let α_n be its empirical measure based on $n \ge 1$ iid samples. Then*

$$\mathbb{E}\, W^2(\alpha_n, \alpha) \le \frac{2B^2}{\sqrt{n}}.$$

Proof. Let F_n and F denote the distribution functions of α_n and α. Both quantile functions take values in $[-B, B]$, so that $|Q_{\alpha_n} - Q_\alpha| \le 2B$. Using the quantile representation of W, we obtain

$$\mathrm{W}^2(\alpha_n, \alpha) = \int_0^1 |Q_{\alpha_n} - Q_\alpha|^2 \, \mathrm{d}s \le 2B \int_0^1 |Q_{\alpha_n} - Q_\alpha| \, \mathrm{d}s = 2B \int_{-B}^{B} |F_n(t) - F(t)| \, \mathrm{d}t.$$

The last identity is the representation of the one-dimensional 1-Wasserstein distance by distribution functions. For fixed t, the random variable $nF_n(t)$ is binomially distributed with parameters n and $F(t)$. By Cauchy–Schwarz,

$$\mathbb{E}|F_n(t) - F(t)| \le \sqrt{\frac{F(t)(1 - F(t))}{n}} \le \frac{1}{2\sqrt{n}}.$$

Taking expectations in the first estimate and using Tonelli's theorem, we conclude

$$\mathbb{E}\,\mathrm{W}^2(\alpha_n, \alpha) \le 2B \int_{-B}^{B} \mathbb{E}|F_n(t) - F(t)| \, \mathrm{d}t \le \frac{2B^2}{\sqrt{n}}.$$

□

Proposition A.5. *Let $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathcal{P}_2(\mathcal{P}_2(X))$. Let $\mu_1, \dots, \mu_n$ be iid with law $\boldsymbol{\mu}$, let $\nu_1, \dots, \nu_n$ be iid with law $\boldsymbol{\nu}$, and consider the empirical measures*

$$\boldsymbol{\mu}_n := \frac{1}{n} \sum_{k=1}^{n} \delta_{\mu_k}, \qquad \boldsymbol{\nu}_n := \frac{1}{n} \sum_{k=1}^{n} \delta_{\nu_k}.$$

Then, for arbitrary fixed slices in (10)*, it holds*

$$\mathbb{E}_n \left| \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}_n, \boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma) \right| \le \frac{\sqrt{8}\, R}{n^{1/4}} \left(\frac{1}{L_g} \sum_{j=1}^{L_g} \|g_j\|_{L_2}^2 \right)^{1/2}, \quad (16)$$

where $\mathbb{E}_n$ denotes the expectation over the samples. The same bound holds for DSW *if the empirical second moment on the right-hand side is replaced by $m_2(\gamma)$.*

Proof. Reduction to one measure. By the triangle inequality for W and Minkowski's inequality, $\mathrm{DSW}_{L_\xi, L_g}$ is a pseudometric on $\mathcal{P}_2(\mathcal{P}_2(X))$. Hence

$$\left| \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}_n, \boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma) \right| \le \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}_n, \boldsymbol{\mu} \mid \gamma) + \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\nu}_n, \boldsymbol{\nu} \mid \gamma).$$

It suffices to bound the expectation of the first term. The second term is treated in the same way.

Reduction to one dimension. Fix (ξ, g). The pushforward of $\boldsymbol{\mu}_n$ is

$$(M_{\xi,g})_{\#} \boldsymbol{\mu}_n = \frac{1}{n} \sum_{k=1}^{n} \delta_{M_{\xi,g}(\mu_k)}.$$

Since the $M_{\xi,g}(\mu_k)$ are iid with law $(M_{\xi,g})_{\#}\boldsymbol{\mu}$, this is the empirical measure of $(M_{\xi,g})_{\#}\boldsymbol{\mu}$. By Lemma A.2, both measures are supported in $[-R\|g\|_{L_2}, R\|g\|_{L_2}]$. Lemma A.4 with $B = R\|g\|_{L_2}$ therefore gives

$$\mathbb{E}_n \, c_{\boldsymbol{\mu}_n, \boldsymbol{\mu}}(\xi, g) \le \frac{2R^2 \|g\|_{L_2}^2}{\sqrt{n}}.$$

Conclusion. By Jensen's inequality for the square root, we obtain

$$\mathbb{E}_n \, \mathrm{DSW}_{L_\xi, L_g}(\boldsymbol{\mu}_n, \boldsymbol{\mu} \mid \gamma) \le \left(\int \mathbb{E}_n \, c_{\boldsymbol{\mu}_n, \boldsymbol{\mu}} \, \mathrm{d}(\sigma_{L_\xi} \otimes \gamma_{L_g}) \right)^{1/2} \le \frac{\sqrt{2}\, R}{n^{1/4}} \left(\int \|g\|_{L_2}^2 \, \mathrm{d}\gamma_{L_g}(g) \right)^{1/2}.$$

Adding the same bound for $\boldsymbol{\nu}$ proves (16).

The case of DSW. Here we replace $\sigma_{L_\xi} \otimes \gamma_{L_g}$ by $\sigma \otimes \gamma$ in the argument above. The exchange of $\mathbb{E}_n$ and the integral is then justified by Tonelli's theorem. Indeed, by Lemma A.2, the atoms $M_{\xi,g}(\mu_k)$ and the measure $(M_{\xi,g})_{\#}\boldsymbol{\mu}$ depend continuously on $(\xi, g, \mu_1, \dots, \mu_n)$. Hence the map

$$(\xi, g, \mu_1, \dots, \mu_n) \mapsto c_{\boldsymbol{\mu}_n, \boldsymbol{\mu}}(\xi, g)$$

is continuous on a product of separable metric spaces and thus jointly Borel measurable. □

Combined approximation. Finally, we approximate the slicing measures and the measures simultaneously.

Proposition A.6. *Let the assumptions of Propositions A.3 and A.5 hold, and assume in addition that the samples are independent of the slices. Then*

$$\mathbb{E}\left|\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu}_n,\boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right| \leq \frac{\sqrt{8}\,R\sqrt{m_2(\gamma)}}{n^{1/4}} + 2R\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right)^{1/4}, \tag{17}$$

where $\mathbb{E}$ denotes the expectation over samples and slices.

Proof. We insert the intermediate quantity $\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)$ and apply the triangle inequality. This gives

$$\begin{aligned}\mathbb{E}\left|\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu}_n,\boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right| &\leq \mathbb{E}\left|\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu}_n,\boldsymbol{\nu}_n \mid \gamma) - \mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right| \\ &\quad + \mathbb{E}_s\left|\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right|.\end{aligned}$$

The second term is bounded by Proposition A.3.

For the first term, joint measurability in slices and samples follows as at the end of the proof of Proposition A.5. Since the samples are independent of the slices, Fubini's theorem allows us to first take the expectation over the samples with the slices fixed. Proposition A.5 then bounds the first term by

$$\frac{\sqrt{8}\,R}{n^{1/4}}\,\mathbb{E}_s\left(\int \|g\|_{L_2}^2 \,\mathrm{d}\gamma_{L_g}(g)\right)^{1/2}.$$

By Jensen's inequality and since the g_j have law γ,

$$\mathbb{E}_s\left(\int \|g\|_{L_2}^2 \,\mathrm{d}\gamma_{L_g}(g)\right)^{1/2} \leq \left(\mathbb{E}_s \int \|g\|_{L_2}^2 \,\mathrm{d}\gamma_{L_g}(g)\right)^{1/2} = \sqrt{m_2(\gamma)}.$$

Combining both bounds proves the assertion. □

Remarks. *Sample sizes.* The proofs of Propositions A.5 and A.6 use neither independence between the μ_k and the ν_k nor equal sample sizes. With m samples of $\boldsymbol{\nu}$, the factor $\sqrt{8}\,n^{-1/4}$ in (16) and (17) becomes

$$\sqrt{2}\left(n^{-1/4} + m^{-1/4}\right).$$

Slice rates. The slice rates $L_\xi^{-1/4}$ and $L_g^{-1/4}$ in (14) stem from the estimate $|\sqrt{a} - \sqrt{b}| \leq |a-b|^{1/2}$. This estimate is only tight when $\mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)$ is close to zero. For the squared distance, the proof of Proposition A.3 yields the Monte Carlo rate

$$\mathbb{E}_s\left|\mathrm{DSW}^2_{L_\xi,L_g}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma) - \mathrm{DSW}^2(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right| \leq 4R^2\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right)^{1/2}.$$

If $\mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma) > 0$, the estimate $|\sqrt{a} - \sqrt{b}| \leq |a-b|/\sqrt{b}$ transfers this rate to DSW itself. More precisely,

$$\mathbb{E}_s\left|\mathrm{DSW}_{L_\xi,L_g}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)\right| \leq \frac{4R^2}{\mathrm{DSW}(\boldsymbol{\mu},\boldsymbol{\nu} \mid \gamma)}\left(\frac{m_2(\gamma)^2}{L_\xi} + \frac{m_4(\gamma)}{L_g}\right)^{1/2}.$$

Sample rate. In contrast, the rate in Lemma A.4 cannot be improved in general. Consider $\alpha = \frac{1}{2}(\delta_{-B} + \delta_B)$. Then

$$\mathrm{W}^2(\alpha_n,\alpha) = 4B^2\left|\alpha_n(\{B\}) - \frac{1}{2}\right|,$$

and the expectation of the right-hand side is of order $B^2 n^{-1/2}$.

A.3 Double-Sliced Wasserstein compared vs Mean Hidden Token Sliced Wasserstein

The relation between DSW and mean-SW follows directly from the quantile representation of the inner Wasserstein distance. Let μ_x denote the token cloud of a document x and define its mean by

$$m_x := \int z \, \mathrm{d}\mu_x(z).$$

For $\xi \in \mathbb{S}^{d-1}$, decompose the quantile function of the projected token cloud as

$$Q_{\xi\#\mu_x}(u) = \langle \xi, m_x \rangle + r_{\xi,x}(u), \qquad \int_0^1 r_{\xi,x}(u) \, \mathrm{d}u = 0.$$

Hence, for a second slicing function $g \in L_2([0,1])$, the corresponding DSW feature satisfies

$$M_{\xi,g}(\mu_x) := \left\langle Q_{\xi\#\mu_x}, g \right\rangle_{L_2} = a_g \langle \xi, m_x \rangle + \langle r_{\xi,x}, g \rangle_{L_2}, \qquad a_g := \int_0^1 g(u) \, \mathrm{d}u.$$

The first term depends only on the centroid of the token cloud, whereas the second term captures its centered projected shape.

In particular, for the constant function $g \equiv 1$,

$$M_{\xi,1}(\mu_x) = \langle \xi, m_x \rangle,$$

so restricting the second slicing stage to constant functions recovers mean-SW, up to the chosen normalization. Conversely, if two token clouds satisfy $m_x = m_y$, mean-SW cannot distinguish them, while

$$M_{\xi,g}(\mu_x) - M_{\xi,g}(\mu_y) = \langle r_{\xi,x} - r_{\xi,y}, g \rangle_{L_2}$$

can be nonzero. Thus, DSW adds information precisely through nonconstant test functions that detect differences in the projected quantile shapes. This also motivates the ablation between constant g, nonconstant centered g, and the full family of cutting functions. We note that the degeneration to constant test functions is already discussed in the original DSW construction.

A.4 Approximation of SOL

Proposition 4.2 concerns the exact SOL distance. In practice, we approximate it from finitely many text samples, finitely many slicing directions, and a finite representation of the second-level slicing functions. We separate these approximation effects below.

Finite samples and slices. Let $\boldsymbol{\mu}_n, \boldsymbol{\nu}_n$ denote empirical meta-measures based on n independent samples, and let $\mathrm{DSW}_{L_\xi, L_g}$ denote the estimator obtained by replacing the two slicing measures by L_ξ and L_g Monte Carlo samples. Under the assumptions of Propositions A.3 and A.5, the sample and slicing approximations converge independently as $n, L_\xi, L_g \to \infty$.

Finite resolution of the second slicing functions. The exact second-level feature is

$$M_{\xi,g}(\mu) = \left\langle Q_{\mathbf{e}_\xi \mu}, g \right\rangle_{L_2([0,1])}.$$

For $M \geq 1$, partition $[0,1]$ into $I_m = \left(\frac{m-1}{M}, \frac{m}{M}\right]$, $s_m = \frac{m-\frac{1}{2}}{M}$, and define the piecewise-constant approximation $(\Pi_M g)(s) = g(s_m)$, We keep the empirical quantile itself exact and only replace g by $\Pi_M g$. If the token clouds are supported in the ball $B_R(0) \subset \mathbb{R}^d$, then

$$|M_{\xi,g}(\mu) - M_{\xi,\Pi_M g}(\mu)| \leq \|Q_{\mathbf{e}_\xi \mu}\|_{L_2} \|g - \Pi_M g\|_{L_2} \leq R \|g - \Pi_M g\|_{L_2}.$$

Consequently, if DSW_M denotes the corresponding distance with g replaced by $\Pi_M g$, then

$$|\mathrm{DSW}_M(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma) - \mathrm{DSW}(\boldsymbol{\mu}, \boldsymbol{\nu} \mid \gamma)| \leq 2R \left(\mathbb{E}_{g\sim\gamma} \|g - \Pi_M g\|_{L_2}^2\right)^{1/2}.$$

For the RBF Gaussian process used in our experiments, $k_\lambda(s,t) = \exp\left(-(s-t)^2/(2\lambda^2)\right)$, we have

$$\mathbb{E}\|g - \Pi_M g\|_{L_2}^2 = \sum_{m=1}^{M} \int_{I_m} 2\left(1 - \exp\left(-\frac{(s-s_m)^2}{2\lambda^2}\right)\right) \mathrm{d}s \leq \frac{1}{12\lambda^2 M^2},$$

and hence

$$|\mathrm{DSW}_M - \mathrm{DSW}| \leq \frac{R}{\sqrt{3}\,\lambda M}.$$

Thus the finite grid introduces an $O(M^{-1})$ approximation error for the RBF second-level slicing measure.

The numerical implementation in Appendix F.1 realizes $M_{\xi,\Pi_M g}$ by integrating the empirical quantile exactly over each bin. In particular, the finite-M approximation acts only on the second-level slicing function and not on the empirical quantile itself. The finite estimator need not itself satisfy identity of indiscernibles, the metric statement in Proposition 4.2 concerns the exact SOL distance.

B Extended Related Work

B.1 Detailed Description of MAUVE

MAUVE (Pillutla et al., 2021; 2023) compares two text distributions after quantization in a Euclidean latent space. Given samples $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim p_{\text{data}}$ and $\mathbf{y}_1, \ldots, \mathbf{y}_m \sim p_{\text{gen}}$, it proceeds as follows:

1. **Embedding.** Map the texts to $\mathbb{R}^d$, classically using the last-token representation $\mathbf{h}^\theta$.
2. **Quantization.** Jointly cluster the embedded real and generated samples by K-means into cells $C_1, \ldots, C_K \subset \mathbb{R}^d$.
3. **Histograms.** Form cluster weights
$$\widetilde{p}_{\text{data},k} := n^{-1} \sum_{i=1}^{n} \mathbf{1}_{\{\mathbf{h}^\theta(\mathbf{x}_i) \in C_k\}} \quad \text{and} \quad \widetilde{p}_{\text{gen},k} := m^{-1} \sum_{i=1}^{m} \mathbf{1}_{\{\mathbf{h}^\theta(\mathbf{y}_i) \in C_k\}}.$$
4. **KL divergence frontier.** For $\widetilde{r}_\lambda := \lambda \widetilde{p}_{\text{data}} + (1-\lambda)\widetilde{p}_{\text{gen}}$, define
$$\gamma_1(\lambda) := \exp\left(-c \sum_{k=1}^{K} \widetilde{p}_{\text{gen},k} \log \frac{\widetilde{p}_{\text{gen},k}}{\widetilde{r}_{\lambda,k}}\right), \quad \gamma_2(\lambda) := \exp\left(-c \sum_{k=1}^{K} \widetilde{p}_{\text{data},k} \log \frac{\widetilde{p}_{\text{data},k}}{\widetilde{r}_{\lambda,k}}\right).$$
5. **Aggregation.** MAUVE is the area under the divergence frontier, completed by the endpoints $(0,1)$ and $(1,0)$ and approximated from finitely many λ. Its optimal value is 1.

MAUVE* (Pillutla et al., 2023) uses the same procedure, but replaces the empirical histograms by their Krichevsky–Trofimov smoothed versions,
$$\widetilde{p}_k^\star = \frac{N_k + \frac{1}{2}}{N + \frac{K}{2}},$$
before computing the divergence frontier. Interestingly, the K-means step itself is an optimal K-point quantization problem and can equivalently be formulated as approximation in the Wasserstein distance (Kreitmeier, 2011).

B.2 Overview of Related Work

B.2.1 Comparing Text Representations via Latent Geometries

Latent Text Representations. Classical bag-of-words representations describe a document through its token frequencies (Kusner et al., 2015). Modern transformers instead provide context-dependent representations. In a causal transformer, the final hidden state depends on the complete preceding sequence and can, under suitable assumptions, even uniquely determine it (Nikolaou et al., 2026). In contrast, dedicated text encoders such as BGE, GTE, and E5 (Chen et al., 2024; Li et al., 2023; Wang et al., 2024) are primarily optimized by contrastive objectives for semantic similarity, retrieval, or classification and commonly compress a text into a single dense vector. This is advantageous for semantic retrieval but less suited to our goal of detecting arbitrary distributional changes: such representations are designed to be invariant to semantically minor variations and necessarily discard part of the token-level distributional structure. In contrast, we use the complete collection of contextual states of a causal or bidirectional encoder.

Text-generation evaluation metrics. Reference-based metrics such as BERTScore (Zhang et al., 2020) and MoverScore (Zhao et al., 2019) compare individual generated texts with paired references using contextual token similarities or optimal transport, and therefore do not directly measure the fit between two unpaired text distributions. Distributional metrics address this explicitly. Fréchet BERT Distance (FBD) (Alihosseini et al., 2019; Xiang et al., 2021) fits Gaussian distributions to pretrained text representations and compares them by the Fréchet distance. This restricts the comparison to first and second moments and inherits the statistical difficulties of estimating high-dimensional covariances. Indeed, the analogous FID estimator in vision is known to be finite-sample biased and typically requires large sample sizes (Heusel et al., 2017; Chong & Forsyth, 2020; Berthet et al.,

2026). PRD (Sajjadi et al., 2018; Xiang et al., 2021) separates quality and coverage, but its practical estimator quantizes the latent distributions by K-means. MAUVE (Pillutla et al., 2021; 2023) similarly quantizes pretrained language-model representations before comparing them through divergence frontiers, introducing dependence on both sample and cluster size. Moreover, its standard last-token representation can be insensitive to perturbations earlier in the sequence (He et al., 2023). Finally, Gen. PPL and token entropy are sequence-wise averages rather than discrepancies between distributions and can remain favorable even under complete corpus-level mode collapse.

B.2.2 Optimal Transport

Relation to document optimal transport. Previous work has been done on evaluating text via optimal transport. The Word Mover's Distance (WMD) (Kusner et al., 2015) represents each document as a bag of pretrained word embeddings and compares two documents by optimal transport between their words. Hierarchical Optimal Topic Transport (HOTT) (Yurochkin et al., 2019) is closer in spirit to our approach: documents are distributions over LDA topics, themselves distributions over words, and are compared by OT on the topic space with WMD as ground distance. Our goal is different: we compare *distributions of documents* for generative-model evaluation, represent each document directly by its distribution of contextual transformer states rather than static word embeddings or learned topics, and use Double Sliced Wasserstein to avoid solving the resulting nested OT problem directly. Thus, while HOTT introduces a closely related hierarchical representation, our hierarchy operates at the corpus level and exploits contextual sequence representations together with slicing for efficient computation.

Optimal transport dataset distances. A related line of work compares entire datasets by optimal transport. OTDD (Alvarez-Melis & Fusi, 2020) represents each label by its class-conditional feature distribution and uses the Wasserstein distance between these distributions inside an outer OT problem. To reduce the cost of the inner transports, the practical Bures-OTDD variant approximates each class-conditional distribution by a Gaussian, yielding closed-form W_2 distances between labels. More recently, s-OTDD (Nguyen et al., 2025) replaces this nested computation by a sliced construction based on one-dimensional projections. Unlike the moment-based Gaussian approximation in Bures-OTDD, DSW retains the full class-conditional distributions and has been observed to yield stronger agreement with exact nested transport distances. Note that DSW avoids such moment approximations and produces a stronger correlation with the exact nested transport distance than both Gaussian OTDD and s-OTDD (Piening & Beinert, 2026b).

Further work on nested optimal transport. More generally, several works develop tractable approximations for Wasserstein-type distances between distributions of distributions. For Gaussian mixtures, slicing has been developed from different geometric viewpoints (Piening & Beinert, 2025; Nguyen & Mueller, 2026; Bonet et al., 2026). Piening et al. (2026) further study sliced and linearized Wasserstein surrogates inside Wasserstein-over-Wasserstein constructions, while Bonet et al. (2025) combine Wasserstein-over-Wasserstein geometry with tractable MMD objectives based on Sliced-Wasserstein kernels. Applications of hierarchical optimal transport beyond text and dataset comparison include stereo vision (Clerc et al., 2026), graph comparison (Piening & Beinert, 2026a) and generative modeling in imaging (Dukler et al., 2019; Geuter et al., 2025) and biology (Yachimura et al., 2024; Haviv et al., 2025).

C Experimental Setup and Implementation

C.1 Datasets and Preprocessing

We use five English text datasets, with preprocessing adapted to each experiment and the training format of the evaluated generators.

OpenWebText (OWT). OpenWebText (Gokaslan et al., 2019) is a webtext corpus used for the diffusion-model benchmark, numerical-stability experiments, and controlled corruption tests. The main benchmark uses held-out GPT-2-tokenized sequences packed into blocks of $1{,}024$ tokens, retaining end-of-text boundaries. ELF is evaluated separately against a uniform sample from its released T5-tokenized OWT training corpus, decoded with native special tokens removed. This reference is training-derived rather than held out. Real–real controls compare disjoint subsets.

WebText. We used the released WebText test OpenAI (2019) split for the GPT-2 model-size, generation-length, temperature, and decoding experiments. For human-judgment comparison, we reuse MAUVE's released pairwise annotations (Pillutla et al., 2021) and regenerate continuations from the first $5{,}000$ test documents. Each prompt contains the first 35 GPT-2 tokens, these are removed from both reference and generated texts before continuation-only evaluation, and empty generated continuations are discarded.

AG News. AG News (Zhang et al., 2015) contains news texts labeled World, Sports, Business, or Sci/Tech. We sample class-controlled mixtures to test mode removal and support expansion. Each candidate set contains $5{,}000$ unique examples and is disjoint from the reference set, we repeat sampling over five seeds.

LM1B. The One Billion Word Benchmark Chelba et al. (2013) provides the reference for the MAUVE sample-size and cluster-count study. We construct $25{,}000$ sequences from its held-out test split using the generator's BERT tokenizer and 128-token packing scheme, preserving the same special-token formatting as in generated samples.

For metric evaluation, decoded texts are tokenized with the embedding model's native tokenizer and truncated to the experiment-specific horizon, padding positions are excluded. Word-level corruptions are applied before tokenization, whereas token-level positional perturbations operate on tokenized sequences.

C.2 Encoding Models

We evaluate SOL using several frozen transformer backbones. GPT-2 Large Radford et al. (2019) is our default encoder and also serves as the reference model for generative perplexity. We additionally consider the causal models Qwen3-0.6B-Base Yang et al. (2025) and OLMo-2-0425-1B OLMo et al. (2025), as well as Dream-v0-Base-7B Ye et al. (2025), which uses bidirectional attention. Each model is used to generate the hidden states of the last-layer on the evaluated text, using its native tokenizer. Since the representation spaces differ across the backbones, absolute SOL values are only compared within the same encoder. Additional comparisons with dedicated text encoders BGE-base Chen et al. (2024), GTE-base Li et al. (2023), E5-base Wang et al. (2024), and NeoBERT Breton et al. (2025) are provided in the appendix.

C.3 SOL Estimator and Hyperparameters

Unless stated otherwise, we use the estimator described in Appendix F.1 with the settings in Table 4. The same first- and second-level slicing directions are used for all corpora within a comparison. We report the ordinary SOL distance, i.e., the square root of the squared estimator.

C.4 Correlation with Human Judgments

We evaluate agreement with the released human judgments from MAUVE (Pillutla et al., 2021), using its WebText test data, eight model–sampling configurations, and raw pairwise Amazon Mechanical Turk annotations for *human-like*, *interesting*, and *sensible* text. Following the released analysis notebook, we exclude responses with an estimated annotation time (`Answer.te`) of at most 25 seconds, retaining $1{,}698$ of $3{,}240$ annotation rows.

Parameter	Dream-7B	All other encoders	Description
L_ξ	8192	1024	First-level projection directions
L_g	8192	1024	Second-level GP directions
M		64	Resolution of second-level functions
λ		0.1	RBF-GP length scale
α		d	Dimension correction factor
Projection seeds		3	Independent ξ draws
GP seeds		3	Independent second-level draws

Table 4: Default SOL estimator parameters by encoder. Here, d denotes the encoder's hidden dimension. We evaluate all 3×3 combinations of projection and GP seeds. Experiment-specific deviations are stated in the corresponding sections.

Human preference scores. For each criterion, we fit a Bradley–Terry model to the pairwise human judgments. Each configuration i is assigned a latent score s_i, and the probability that configuration i is preferred over configuration j is modeled as

$$\Pr(i \succ j) = \frac{\exp(s_i)}{\exp(s_i) + \exp(s_j)}.$$

The released pairwise annotations are treated as observed preference outcomes under this model, and the scores are estimated by maximum likelihood. Higher scores therefore correspond to stronger estimated human preference.

Models and sampling. We evaluate four model sizes, each with two sampling methods: GPT-2 and GPT-2 Medium with nucleus sampling at $p = 0.9$ and ancestral sampling, and GPT-2 Large and GPT-2 XL with nucleus sampling at $p = 0.95$ and ancestral sampling.

While the annotations are still publicly available, the set of continuations seemed to be incomplete. Therefore, we resample all GPT-2 models with the corresponding sampling strategies. For each of the eight configurations, we regenerate continuations for the first $5{,}000$ WebText test documents using five generation seeds. Generation uses the first 35 GPT-2 tokens as the prompt and a maximum total length of $1{,}024$ tokens, including the prompt. In particular, we regenerate all configurations because the available released archive contains only GPT-2 and GPT-2 Large samples, whose metadata records 10-token prompts, not 35. Our rerun follows the paper's stated 35-token prompt protocol.

Text evaluation methods. For each generation seed, we average the full-corpus score over all nine combinations of three token-projection seeds and three GP seeds.

Rank correlation. We measure agreement using Spearman rank correlation between the automatic metric scores and the Bradley–Terry scores across the eight configurations. We negate SOL scores, using $-\mathrm{DSW}^2$, so that higher values indicate stronger human preference for every metric.

We report the worst-case correlation $\rho_{\min}$, following Pillutla et al. (2023). For each configuration, we compute the mean and standard deviation of its metric score across the five generation seeds. We then independently shift each mean by either plus or minus one standard deviation and report the lowest Spearman correlation among the resulting combinations.

C.5 OpenWebText Benchmark Configurations

Data preprocessing. We use two reference families. The GPT-2 reference family (G) follows the OpenWebText preprocessing of Sahoo et al. (2024) and uses the final 100,000 OpenWebText documents as a held-out split. Documents are GPT-2-tokenized, terminated with an end-of-text (EOT) token, concatenated within preprocessing batches, and divided into 1,022-token chunks. Adding an EOT at each end gives 1,024-token blocks, incomplete remainders are discarded. The ELF family (E) (Hu et al., 2026) uses a uniform sample without replacement from the released T5-tokenized OpenWebText training corpus. ELF text decoding removes native special tokens, including EOS, PAD, and UNK, and retains the release's whitespace cleanup. For the algorithm see Alg. 1.

For G, GPT-2-large consumes native IDs where available and retokenized archived text otherwise, OLMo and Dream consume decoded text. Decoding from available GPT-2 IDs preserves EOT markers and disables whitespace cleanup. ELF text is retokenized for every evaluator. Evaluator inputs are truncated to 1,024 tokens, with padding excluded, this horizon need not equal 1,024 native generator tokens. Each model comparison uses 5,000 generated sequences and 5,000 rows selected from a 10,000-row reference pool, real–real controls use disjoint subsets. SOL comparisons are made within the same reference family and encoder. LDDM uses the same OWT source and split with different packing-worker boundaries, LangFlow's historical training split is not independently verified.

Algorithm 1 OpenWebText reference preprocessing

Require: Generator configuration G, SOL encoder E, evaluation horizon H
Ensure: Reference corpus matched to the generator's training distribution
1: Identify the OpenWebText source and preprocessing used to train G
2: Reproduce the corresponding tokenization, packing, and special-token conventions
3: Select held-out examples from the same data distribution whenever available
4: **if** the SOL encoder E does not consume the generator's native representation **then**
5: Decode the reference sequences to text using the generator-specific convention
6: Retokenize the decoded text with the native tokenizer of E
7: **end if**
8: Exclude padding positions from the token cloud
9: Use the same preprocessing pipeline for the generated and reference corpora

Evaluation and sampling budgets. We denote discrete-diffusion sampling steps by T and continuous predictor, integration, or flow-map steps by S. For BD3-LM, T is the maximum number of updates per block. NFE counts neural-network evaluations, including correction, initialization, cache-update, guidance, and final-decoding calls; hence S and T need not equal NFE. PC denotes predictor–corrector, ODE ordinary differential equation, and SDE stochastic differential equation. EMA denotes exponential moving-average weights, and KV denotes key–value caching. In the ELF and starred-model rows, CFG denotes self-conditioning guidance, called w or c in the original notation; CFG$= 1$ means no guidance amplification. External-conditioning guidance is fixed to 1. The parameter γ denotes CTM noise mixing or ELF SDE churn, according to the sampler. We cross three direction seeds with three Gaussian-process seeds and report the mean and population standard deviation of the nine SOL estimates. Auxiliary entropy and perplexity retain the evaluator conventions specified in the table. Training data, model sizes, guidance, and sampling budgets differ across configurations.

AR Sahoo et al. (2024). The autoregressive model samples token by token at temperature 1.0, requiring one next-token evaluation per decoding step. For the fixed 1,024-token output with an initial BOS token, this gives 1,023 next-token evaluations. The initial BOS token is not sampled.

MDLM Sahoo et al. (2024). The masked diffusion language model uses the cached denoising diffusion probabilistic model (DDPM) sampler (`ddpm_cache`) with $T = 1000$. We compare the released FP32 Gumbel sampling implementation with exact multinomial categorical sampling under the same sampler configuration. Both include final noise removal. Cached predictions can skip model calls, so $T = 1000$ is not a fixed NFE count.

DUO Sahoo et al. (2025). The base model uses ancestral sampling with $T = 1024$. The distilled model uses $T = 32$, with separate configurations for ancestral and greedy final noise removal.

LDDM Jo et al. (2026). The loopholing discrete diffusion models LDDM-M and LDDM-U use the released samplers with $T = 1024$, EMA weights, FP64 categorical probabilities, and nucleus threshold $p = 1.0$. A final denoising call gives 1,025 actual model evaluations.

Soft-Masked Hersche et al. (2026). The released scratch-trained checkpoint uses top-3 soft feedback and nucleus threshold $p = 0.9$. We evaluate the MDLM sampler and the loop sampler of the

remasking diffusion model (ReMDM-loop), both with $T = 1024$ and final noise removal. ReMDM-loop uses $\eta = 0.02$, $t_{\text{on}} = 0.55$, $t_{\text{off}} = 0.05$, and $\alpha_{\text{on}} = 0.9$. Caching can skip model calls, so actual NFE is sample dependent.

ADLM Rout et al. (2025). The anchored diffusion language model uses the released large checkpoint with EMA weights. We generate 1,024 native GPT-2 tokens using ReMDM-loop with $T = 1024$, nucleus threshold $p = 0.9$, $\eta = 0.02$, $t_{\text{on}} = 0.55$, $t_{\text{off}} = 0.05$, and $\alpha_{\text{on}} = 0.9$, followed by final argmax noise removal. The sampler caches predictions when its token state is unchanged; with generation batch size four, the number of model calls can depend on the batch. The evaluation uses reference family G.

BD3-LM Arriola et al. (2025). Block discrete denoising diffusion language models generate blocks autoregressively and denoise within each block. We evaluate the released OWT checkpoints with block sizes $b \in \{4, 16\}$, denoted BD3-LM block 4 and block 16. Each output has 1,024 native GPT-2 tokens, including the sampler's initial BOS token. We use the released first-hitting sampler with $T = 5000$, nucleus threshold $p = 0.9$, KV caching, and generation batch size one. First-hitting updates unmask one token at a time and stop when the current block is complete; $T = 5000$ is therefore not 5,000 evaluations per block. The implementation additionally makes a model call to store the KV cache after completing each block. We retain the release's rejection rule: after the generated prefix exceeds 256 tokens, reject an attempt if the unigram entropy of its last 256 tokens is below 4 nats, then retry. Thus, total generation cost also includes rejected attempts. There is no separate final decoder call. These checkpoints use reference family B, which retains document EOT separators but omits artificial special tokens at the boundaries of training chunks.

vMF and VP Chemseddine et al. (2026). The three von Mises–Fisher (vMF) PC configurations with $S = 128$ use the warped predictor–corrector sampler (`pc_softmax_warp`), with one corrector update every two predictor steps. The non-self-conditioned and base self-conditioned configurations use corrector step size $\epsilon = 10^{-5}$; the additional self-conditioned configuration uses $\epsilon = 2 \times 10^{-4}$. Predictor and corrector temperatures are both 1.0.

For both vMF and variance-preserving (VP) models, we additionally report warped ODE sampling with $S = 1024$ and warped predictor–corrector sampling with $S = 512$, using one corrector update after every predictor step. The latter therefore uses 512 predictor and 512 corrector evaluations; its corrector step size is 10^{-5} for vMF and 10^{-3} for VP. These larger-budget ODE and PC runs use self-conditioning onset 0.5. Finally, the self-conditioned SDE configurations use 128 warped Euler–Maruyama steps, with noise amplitude $\sigma = 0.01$ for vMF and $\sigma = 0.1$ for VP, decay exponent $q = 1$, and self-conditioning onset 0. Predictor steps exclude final readout.

LangFlow Chen et al. (2026). The released OWT checkpoint uses its native Euler–EDM solver, learned Gumbel schedule, and self-conditioning. We evaluate NFE budgets of 128 and 1,024, each including the final model call used for argmax decoding.

FLM and FMLM Lee et al. (2026). The flow language model (FLM) uses the Euler sampler (`flm_euler`) with $S = 1024$ and 1,024 recorded NFE, temperature 1.0, nucleus threshold $p = 1.0$, attention softcap 50, and logit cap 30, without an additional noise-removal step. The flow map language model (FMLM) uses the released flow-map sampler with $S \in \{1, 2, 4\}$, temperature 1.0, and $\gamma = 1$; the measured NFE equals the respective step count.

FMLM* Yoo et al. (2026). The released flow-map model distilled through the ELF* teacher uses consistency trajectory model (CTM) γ-sampling with $S \in \{1, 2, 5\}$, each evaluated at CFG $\in \{1, 5\}$. We use $\gamma = 0.75$ for two steps and $\gamma = 1$ for one or five steps. We also report deterministic flow-map sampling with $S \in \{8, 128\}$, CFG$= 5$, and $\gamma = 0$. The guidance scale is supplied as model conditioning. The released student uses a zero self-conditioning input at every step, without carrying predictions between steps. Sampling uses one model call per step and a separate final decoding call, giving NFE$= S + 1$. All configurations use reference family G for SOL evaluation.

ELF* Yoo et al. (2026). The released fixed-point denoiser uses consistency deep equilibrium (CDEQ) distillation. We report its deterministic CDEQ sampler with $S = 128$, CFG $\in \{1, 5\}$,

and $\gamma = 0$, corresponding to Euler integration without stochastic churn. Each step uses one student forward pass with zero self-conditioning input at the cold-start phase $\tau_0 = 0.002$, without an inner fixed-point iteration or cross-step self-conditioning. Final decoding requires one additional model call, giving NFE$= 129$. The starred ELF and FMLM checkpoints use a GPT-2 tokenizer and GPT-2-large latent encoder; both are evaluated against reference G. Our starred-model runs use a deterministic logit-normal quantile grid, with location -1.5 and scale 0.8.

ELF Hu et al. (2026). The original T5-based embedded language flows (ELF), with checkpoints ELF-B/M/L, are evaluated against reference family E. The 32-step ELF-B configurations use SDE sampling with $\gamma = 1.5$ and $\mathrm{CFG} \in \{1, 3\}$. Additional ELF-B settings use ODE sampling with $S \in \{128, 1024\}$ and $\gamma = 0$, and SDE sampling with $S = 128$ and $\gamma = 1$, all at CFG$= 1$.

We also evaluate ELF-B, ELF-M, and ELF-L with 64 integration steps. The shared configuration uses SDE sampling with $(\gamma, \mathrm{CFG}) = (1, 1)$ for all three sizes. The calibrated configurations use SDE $(0.5, 0.5)$ for B, ODE $(0, 1.25)$ for M, and SDE $(0.75, 1)$ for L. These were selected in a 256-sample pilot by minimizing generative perplexity among settings within 0.02 nats of the reference token entropy.

These original ELF runs use randomly sampled, sorted logit-normal time grids with location -1.5 and scale 0.8, and a separate final decoding call. The reported guidance scales are self-conditioning guidance; the external-conditioning guidance scale is fixed to 1. Integration steps exclude any additional guidance and final-decoding evaluations.

The separately retrained GPT-2 ELF-B teacher from the ELF* release uses reference family G. Its additional run uses ODE sampling with $S = 128$, CFG$= 1$, $\gamma = 0$, and the deterministic logit-normal quantile grid, giving 128 denoising calls plus one final decoder call.

C.6 Runtime

$K = L$	8	16	32	64	128	256	512	1024	2048	4096	8192	16384
Time [s]	0.38	0.39	0.38	0.39	0.41	0.47	0.63	1.02	2.16	5.87	18.95	61.16

Table 5: Median post-hidden DSW runtime across three repeats on a single NVIDIA GeForce RTX 5090, with 5,000 texts per arm and $M = 64$. Hidden states remain on GPU across direction blocks, the initial CPU-to-GPU transfer is included.

We tested the runtime on a single NVIDIA GeForce RTX 5090 using 5,000 samples and report the median over three runs. The DSW runtime is given in Table 5. Computing the hidden states takes 71 s for GPT-2 Large, 65 s for Qwen3-0.6B, 89 s for OLMo-2-1B, and 375.8 s for Dream-7B.

D ADDITIONAL ABLATIONS

Because the reference models operate in hidden spaces of different dimensionality, absolute score values are not directly comparable across models, only relative comparisons within each reference model are meaningful.

D.1 NUMBER OF SLICES AND SAMPLES

As we have compared the GPT-2 model in Section 5.4. We use Dream-v0-Base-7B as the embedding model and compare held-out OpenWebText (OWT) texts with autoregressive generations. For the slice-count experiment, we fix 5000 texts per corpus, using generations from MDLM's autoregressive baseline and a maximum length of 1024 Dream tokens. SOL stabilizes after a few hundred slices see Figure 8, while variability across projection seeds decreases with increasing slice count. Varying the inner and outer slice counts separately or jointly yields consistent limiting values.

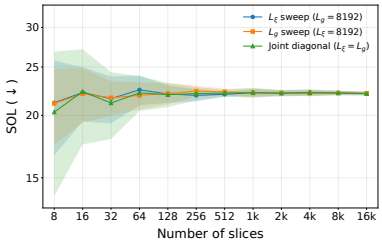


Figure 8: Increasing the number of slices stabilizes SOL, using Dream as embedding model.

For the sample-size experiment, we compared OWT texts with the same AR model, using the same Dream embedding model with a maximum length of 256 tokens. We fix $L_\xi = L_g = 8192$ and vary the number of texts per corpus from 100 to 25,000. We separate uncertainty due to random projections from uncertainty due to data subsampling, reporting both contributions alongside the total standard deviation, see Figure 9.

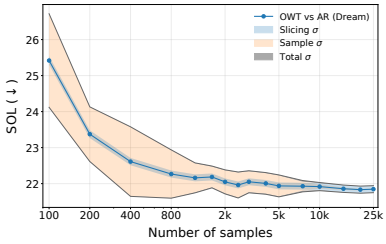


Figure 9: SOL converges with sample size, with std over slices and samples, using Dream as embedding model.

D.2 QUANTILE RESOLUTION

We vary $M \in \{8, 16, 32, 64, 128, 256, 512, 1024\}$ while fixing $L = K = 1024$ and using the same 5,000 samples per distribution. We use GPT-2-large token embeddings with a 256-token cap and cross three direction seeds with eight GP seeds, coupling the draws across resolutions. Increasing M has only a small effect once the resolution is sufficiently large: the plotted SOL values vary by less than 0.1% for $M \geq 16$, see Figure 10.

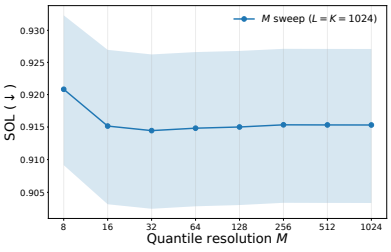


Figure 10: Increasing the M for the quantile resolution has a small impact beyond $M = 64$.

The shaded bands reflect variability across the 24 projection-seed pairs, rather than sample resampling. They show the mean squared SOL $\pm$ one standard deviation, followed by a square-root transformation. Since L and K remain fixed, increasing M improves quantile resolution without increasing the number of random slices, so this variability need not decrease.

D.3 GAUSSIAN PROCESS BANDWIDTH

The RBF length scale λ controls the smoothness of the second-level slicing functions. For fixed grid resolution M, small λ approaches independent Gaussian noise across quantile bins, whereas large λ approaches a constant function and hence mean-SW. For $M = 64$ and $\lambda = 0.01$, adjacent bins have correlation 0.30, which already drops to 0.008 at a separation of two bins. At the opposite extreme, for $\lambda = 3$, even the two endpoints of $[0, 1]$ have correlation 0.95, so the sampled functions are nearly constant. Our default $\lambda = 0.1$ lies between these regimes: one correlation length spans approximately $M\lambda = 6.4$ bins.

D.4 MEAN-SW VERSUS SOL UNDER STRUCTURAL PERTURBATIONS

As discussed in Appendix A.3, restricting the second slicing function to $g \equiv 1$ recovers mean-SW at the level of exact quantile integrals. Mean-SW therefore depends only on the centroid of each token cloud, whereas nonconstant second-stage functions additionally probe the centered shape of its projected token distribution. Since mean-SW is already a strong baseline, we ask whether retaining this additional token-cloud information increases the separation from the finite-sample real-text baseline under controlled structural changes.

For each perturbation c, we normalize both metrics by their respective clean real–real distance,

$$R_D(c) := \frac{D(P, Q_c)}{D(P, Q_{\text{clean}})}.$$

We use the ordinary distance, i.e. take the square root of the squared estimator before forming the ratio, and use the same projections in the numerator and denominator. For the passage-reorganization experiments, Q_{clean} denotes the corresponding unperturbed passage construction.

We repeat the experiment on five disjoint pairs of $5{,}000$ reference and $5{,}000$ candidate documents, using GPT-2 Large, a horizon of 128 tokens, and $L_\xi = L_g = 1024$. We use three first-level projection seeds and three GP seeds for SOL. To compare the relative responses, we report

$$\Delta(c) := \frac{1}{5}\sum_{b=1}^{5} \left(\mathbb{E}_{\text{sliced-seed}} \log R_{\text{SOL}}(c) - \mathbb{E}_{\text{sliced-seed}} \log R_{\text{mean-SW}}(c)\right).$$

Thus, $e^{\Delta(c)}$ is the geometric-mean relative response of SOL compared with mean-SW.

SOL has the larger clean-normalized response for nine of the ten perturbations, see Table 6. The difference is strongest for changes in word order and larger-scale passage organization. For selected-word shuffling, $\Delta = 0.390$, corresponding to $e^{\Delta} = 1.48$, i.e., a $1.48\times$ larger geometric-mean relative response for SOL. In contrast, duplicate/delete repetition produces a slightly larger relative response for mean-SW. Hence, the additional token-cloud information does not uniformly amplify all perturbations, but primarily increases the response to changes in within-document contextual structure.

Perturbation	mean-SW	SOL	Δ
Selected-word shuffle	10.66	**15.75**	0.390
Adjacent word swaps	7.09	**9.63**	0.305
Short-window shuffle	5.63	**7.61**	0.301
Reverse short spans	7.77	**10.81**	0.329
Clause permutation	2.61	**3.36**	0.252
Sentence permutation	2.12	**2.60**	0.203
Move a phrase	2.14	**2.83**	0.277
Duplicate/delete repetition	**8.83**	8.62	-0.024
Reorganize passage mixtures	5.82	**8.79**	0.413
Reorganize repetitions	9.12	**10.47**	0.139

Table 6: Clean-normalized ordinary-distance responses of mean-SW and SOL under structural perturbations. Response columns are arithmetic means of paired ratios. Δ is the mean paired log-response contrast.

Since SOL is generally more sensitive to structural perturbations while adding little runtime beyond hidden-state computation, we use SOL in the main experiments

D.5 SENSITIVITY TO SAMPLING TEMPERATURE

We evaluate GPT-2-large samples generated at five sampling temperatures against WebText, see Table 7 . All encoders behave as expected, assigning lower SOL scores to intermediate temperatures, with slightly different optima: GPT-2-large and Dream prefer $t = 1.0$, while Qwen3-0.6B and OLMo-2-1B prefer $t = 0.9$.

Encoder	Length	$t = 0.7$	$t = 0.9$	$t = 1.0$	$t = 1.1$	$t = 1.3$
GPT-2-large	512	2.901	1.066	**0.994**	1.038	1.355
	1024	2.872	1.091	**1.014**	1.070	1.434
Qwen3-0.6B	512	6.755	**5.085**	5.204	5.440	6.332
	1024	6.779	**5.179**	5.257	5.565	6.538
OLMo-2-1B	512	7.719	**4.051**	4.864	5.947	8.211
	1024	8.562	**4.103**	4.758	6.051	8.561
Dream-7B	512	27.769	21.576	**21.090**	22.066	25.889
	1024	27.229	21.495	**21.091**	21.980	25.790

Table 7: SOL evaluating generated text by GPT-2 Large at different decoding temperatures using differnet embedding models.

D.6 POLLUTION AND CORRUPTION

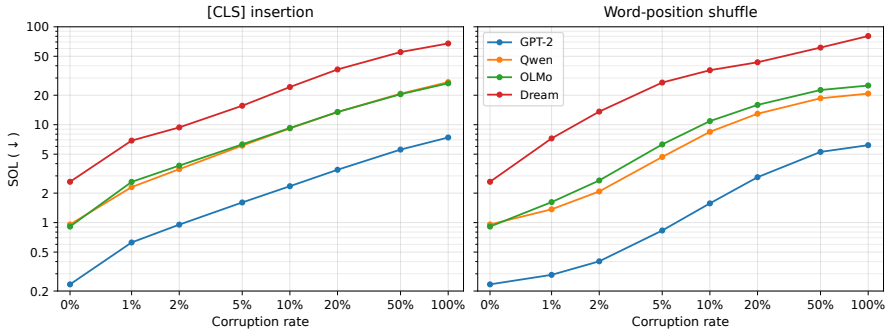


Figure 11: We are increasing shuffling the tokens or inserting a CLS token, all generative language models similarly weight [CLS] insertion to shuffling. The values are normalized to the 0 % value for comparability.

We further investigate whether the response of DSW to textual perturbations is specific to a particular language-model representation. To this end, we repeat the corruption experiment using four substantially different backbone models: GPT-2-large, Qwen3-0.6B-Base, OLMo-2-1B, and Dream-v0-Base-7B. For all encoders, we use the same split of OpenWebText consisting of 5,000 clean reference documents and 5,000 disjoint candidate documents. Thus, differences between the resulting curves arise from the representation induced by the encoder rather than from different underlying documents.

Corruption	[CLS] pollution	Word shuffling
0%	0.9702	0.9702
1%	0.9367	0.9701
2%	0.9009	0.9667
5%	0.7888	0.9527
10%	0.6559	0.8423
20%	0.4558	0.3555
50%	0.1977	0.0353
100%	0.0545	0.0180

Table 8: MAUVE under [CLS] pollution and word shuffling using last hidden representation.

We consider two types of corruption. For the *insertion* experiment, literal [CLS] strings are inserted at randomly chosen whitespace-word boundaries, where the corruption rate denotes the number of insertions relative to the original number of words. For the *shuffle* experiment, the prescribed fraction of whitespace-word positions is selected and the corresponding words are randomly permuted. Hence, shuffling preserves the word multiset while increasingly disrupting local and global word order. In both cases, we use corruption rates

$$0\%,\ 1\%,\ 2\%,\ 5\%,\ 10\%,\ 20\%,\ 50\%,\ 100\%.$$

Importantly, the corruption is applied to the raw text before tokenization. Each corrupted document is subsequently processed using the respective model's own tokenizer and truncated to a horizon of 256 tokens. We construct the token cloud from the final hidden-state representations without pooling. Consequently, the textual corruption is identical across backbones, whereas differences induced by tokenization, hidden dimension, and truncation remain part of the encoder comparison.

This experiment therefore shows that the continuous corruption response of DSW is not specific to a single representation space. We emphasize that the four-backbone comparison itself is DSW-only, the MAUVE results in Table 8 serve as a separate metric comparison and were not recomputed for each of the four encoder backbones.

D.7 Positional Sensitivity

We shuffle 20% of the tokens within either the first or last 5% of each sequence. Table 9 shows a pronounced prefix sensitivity for the three causal encoders: prefix shuffling raises their scores substantially, while suffix shuffling leaves them close to the clean baseline.

In contrast, the bidirectional Dream encoder responds similarly at both locations, with a slightly larger suffix response.

Encoder	Clean	Prefix	Suffix
OLMo-2-1B	0.870	2.048	0.913
Dream-7B	2.493	**4.496**	**4.621**
Qwen3-0.6B	0.914	1.937	0.946
GPT-2-large	0.224	0.340	0.226

Table 9: Shuffle 20% of the tokens the first or last 5% of the text.

This is consistent with causal contextualization that amplifies early edits through subsequent token states. The comparison concerns positional balance within each encoder, since absolute score scales differ across encoders.

D.7.1 POSITIONAL BIAS OF LAST-TOKEN SW

We evaluate sliced Wasserstein distance on GPT-2-large's final-token hidden states using 5,000 disjoint reference and 5,000 candidate WikiText documents, a 256-token horizon, and 1,024 projection directions. We replace or shuffle ten tokens at the beginning, middle, or end, averaging over five data/transformation seeds and three projection seeds. For random replacement, the dimension-scaled SW distance increases from a clean baseline of 0.97 to 2.55, 3.75, and 23.92 for beginning, middle, and end corruption, respectively. For shuffling, the corresponding scores are 3.50, 2.65, and 23.56. After subtracting the clean baseline, end corruption produces approximately $14.5\times$ and $8.9\times$ the beginning-corruption response for replacement and shuffling, respectively, demonstrating a strong positional bias toward the final tokens. This means it suffers the same problem as MAUVE, see Figure 3.

D.8 MODE COVERAGE

AG News (Zhang et al., 2015) contains four classes: WORLD, SPORTS, BUSINESS, and SCI/TECH. We define the reference distribution as $P = \frac{1}{2}\text{WORLD} + \frac{1}{2}\text{SPORTS}$. The three candidate distributions are $Q_1 = \text{WORLD}$, representing missing diversity, $Q_2 = \frac{1}{2}\text{WORLD} + \frac{1}{2}\text{SPORTS}$, which matches the class distribution of P but contains disjoint examples, and $Q_3 = \frac{1}{4}(\text{WORLD} + \text{SPORTS} + \text{BUSINESS} + \text{SCI/TECH})$, representing additional mass outside the reference support.

Metric	$Q_2 \to Q_1$	$Q_2 \to Q_3$
SOL	1.000	1.000
mean-SW	1.000	0.9995

Table 10: Spearman correlations with the target share along the $Q_2 \to Q_1$ and $Q_2 \to Q_3$ interpolation paths.

Holding P fixed, we interpolate independently from Q_2 toward Q_1 and Q_3:

$$Q_{2\to j}(\lambda) = (1-\lambda)Q_2 + \lambda Q_j, \qquad j \in \{1, 3\}.$$

We evaluate $\lambda \in \{0, .01, .02, .05, .10, .20, \ldots, .90, .95, .98, .99, 1\}$. Each arm contains $5{,}000$ unique examples and is disjoint from the $5{,}000$-example reference set. Results are averaged over five independently sampled dataset seeds.

Both metrics increase monotonically away from the matched distribution Q_2. The two paths represent distinct errors: Q_1 removes the SPORTS mode, whereas Q_3 introduces the previously absent BUSINESS and SCI/TECH modes. For mean-sliced Wasserstein, the rank correlation are not always monotone in this test, see Table 10.

D.9 ALTERNATIVE TO BIDIRECTIONAL ATTENTION

As seen in Section D.7, causal transformers prefer changes in the beginning. Bi-directional attention circumvents this problem, but we also test two alternative proposed methods. As proposed in Springer et al. (2025) (also used in Ewais et al. (2026)), one can also repeat the whole sequence $\mathbf{x}$ and pass $[\mathbf{x}, \texttt{<sep>}, \mathbf{x}]$ to make each token conditioned on the whole sequence.

Encoder	Length	$t = 0.7$	$t = 0.9$	$t = 1.0$	$t = 1.1$	$t = 1.3$
GPT-2-large	512	2.141	1.299	1.138	1.035	**1.032**
	1024	–	–	–	–	–
Qwen3-0.6B	512	6.083	4.310	**4.288**	4.347	4.847
	1024	6.695	4.513	**4.470**	4.645	5.311
OLMo-2-1B	512	7.822	5.452	5.016	**4.808**	5.203
	1024	9.413	6.440	5.421	4.916	**4.884**

Table 11: SOL scores using the $[\mathbf{x}, \mathbf{x}]$ second-pass representation across sampling temperatures.

Repeat the sequence. We compare normal encoding $[\mathbf{x}]$ with repetition $[\mathbf{x}, \mathbf{x}]$ and EOS-separated repetition $[\mathbf{x}, \texttt{<sep>}, \mathbf{x}]$ in Table 11, using OLMo-2-1B. Repeated inputs are encoded in one causal forward pass, and only second-copy final-layer states enter the uniform token measure, The EOS remains visible to attention, but is not scored and does not reset attention.

All scores average three projection seeds crossed with three GP seeds. For regional corruption, we use disjoint 5,000-document reference and candidate sets and match the corrupted tokens across representations.

Table 12 compares identical corruption strengths applied to the prefix, middle, or suffix. With standard encoding, prefix corruption increases the score by 88% relative to the clean baseline, compared with only 4% for suffix corruption. Repetition without a separator makes these responses more similar but reverses their ordering: prefix corruption increases the score by 59%, whereas suffix corruption increases it by 118%, roughly twice as much. EOS-separated repetition retains a strong imbalance, with increases of 126% for prefix corruption and 12% for suffix corruption. The same reversal without a separator occurs at a sequence length of 256. All three methods respond monotonically to increasing replacement strength within these 5% regions. Thus, repetition without a separator reduces positional imbalance but does not eliminate it.

Region	$[\mathbf{x}]$	$[\mathbf{x}, \mathbf{x}]$	$[\mathbf{x}, \texttt{<sep>}, \mathbf{x}]$
Clean	0.910	0.732	0.853
Prefix	1.710	1.167	1.926
Middle	1.335	1.347	1.375
Suffix	0.942	1.599	0.958

Table 12: OLMo regional corruption, cap 1,024. Mean SOL for 10% replacement within a 5% region, taking the square root per seed before averaging, clean scores use uncorrupted candidate text.

The temperature sweep also reveals a change in ranking (Table 13). We score unconditional, natural-length GPT-2-large generations against OpenWebText at a maximum sequence length of 1,024. Standard encoding and EOS-separated repetition both favor temperature 0.9, whereas repetition without a separator favors 1.3. This preference for a higher temperature runs counter to the expectation that moderate sampling temperatures provide a better balance between coherence and diversity. Repetition also changes the score scale, so lower raw scores across methods do not by themselves indicate improvement. The injection of EOS restores the normal temperature, but the sensitivity is not completely restored.

Temp.	$[\mathbf{x}]$	$[\mathbf{x}, \mathbf{x}]$	$[\mathbf{x}, \texttt{<sep>}, \mathbf{x}]$
0.7	8.562	9.413	8.152
0.9	**4.103**	6.440	**4.060**
1.0	4.758	5.421	4.703
1.1	6.051	4.916	5.931
1.3	8.561	**4.884**	8.205

Table 13: SOL: GPT-2-large generations scored with OLMo at cap 1,024. Square root of the mean squared score across seeds. Bold marks the minimum within each method.

D.10 Text Embedding Models

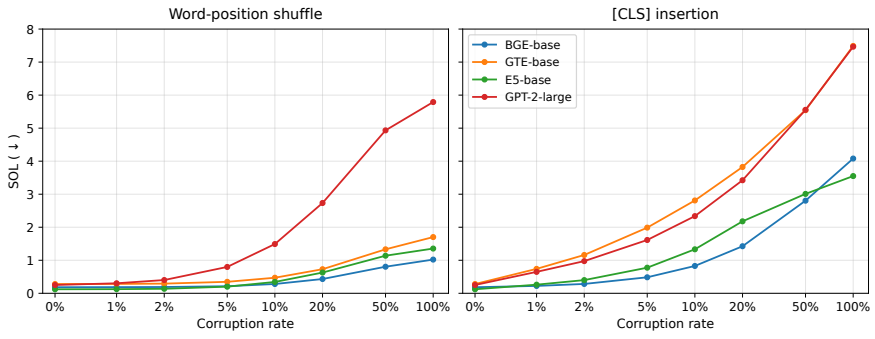


Figure 12: SOL under increasing shuffling of tokens (left) and inserting a CLS token (right). Compared to GPT2-Large, the text encoders are less sensitive to shuffling than GPT-2 Large.

As an alternative to the causal GPT-2 representation, we evaluate modern text encoders, including BGE-base Chen et al. (2024), GTE-base Li et al. (2023), and E5-base Wang et al. (2024). For this experiment, we split OpenWebText into two disjoint sets of 5,000 documents, using one as a clean fixed reference and the other as the candidate distribution. Candidate documents are increasingly corrupted either by shuffling a fraction of their words or by inserting the literal string `[CLS]` into a fraction of word boundaries. Each corruption level is generated independently of the original clean candidate set. The resulting texts are retokenized with the native tokenizer of each encoder and truncated to the first 256 tokens, and DSW is computed from all final-layer token representations.

As shown in Figure 12, the text encoders are substantially less sensitive to small perturbations than GPT-2 Large. For example, at only 1% word shuffling, DSW^2 increases by a factor of 1.459 over the clean–clean floor for GPT-2 Large, compared with only 1.022, 1.064, and 1.026 for GTE, E5, and BGE, respectively. Thus, the text encoders do detect sufficiently strong corruption, but resolve small local perturbations much less clearly. This is undesirable for our purpose, since differences in the underlying text distribution should already be reflected at low corruption rates.

NeoBERT Breton et al. (2025) exhibits an even stronger failure mode under `[CLS]` insertion. As shown in Table 14, increasing the corruption rate from 20% to 50% decreases DSW^2 from 2321.64 to 2060.10, although the candidate texts are more heavily corrupted. These results suggest that representations optimized primarily for text embedding can be comparatively invariant to precisely the local token-level perturbations that a distributional text metric should detect.

Insertion rate	SOL (↓)
20%	1
50%	0.942

Table 14: NeoBERT SOL under increasing `[CLS]` insertion, normalized to the 20% value.

E Additional Evaluation Results

E.1 Human-Judgment Results

	MAUVE			SOL		
Criterion ↑	GPT-2 Large	OLMo2 1B	Dream 7B	GPT-2 Large	OLMo2 1B	Dream 7B
Human-like	0.881	0.929	0.905	0.952	0.952	**0.976**
Interesting	0.643	0.786	0.690	0.762	0.762	**0.857**
Sensible	0.833	0.881	0.881	0.929	0.929	**0.976**

Table 15: Human judgement comparison at sequence length 256. Worst-case Spearman rank correlation between metric-induced model rankings and the Bradley–Terry (BT) human ranking. Results are shown for MAUVE and SOL using GPT-2 Large, OLMo2 1B, and Dream 7B embeddings. Reported are the minimum correlations $\rho_{\min}$ across the evaluated settings at sequence length 256.

Table 15 reports agreement with the Bradley–Terry human rankings at a sequence length of 256 tokens, matching the maximum amount of text shown in the original human study. Across all three criteria, SOL attains higher worst-case rank correlation than MAUVE for each evaluated backbone. The strongest correlations are obtained with Dream for all three criteria in this setting.

E.2 MAUVE Comparison

E.2.1 Mean Hidden Representation

In MAUVE we can also use the mean-hidden pooling (4) instead of the last hidden representation (3). This setup was also used in (Franca & Tong, 2026), but has a large drawback. Polluting the sequence with `[CLS]` or random tokens is not linearly degrading the performance as for MAUVE with last hidden representation Table 8, but quickly changing from really good to bad, see Table 16.

Corruption	`[CLS]` pollution	Word shuffling
0%	0.9725	0.9725
1%	0.9590	0.9710
2%	0.9183	0.9697
5%	**0.4709**	0.9597
10%	**0.0624**	**0.8236**
20%	0.0181	**0.1208**
50%	0.0127	0.0191
100%	0.0118	0.0162

Table 16: MAUVE under `[CLS]` pollution and word shuffling using mean pooling.

E.2.2 LM1B Experiment

We evaluated 25,000 samples generated by the vMF model from (Chemseddine et al., 2026) using 64 predictor steps and one Langevin corrector step per predictor step, corresponding to 128 function evaluations. We constructed 25,000 reference sequences from the held-out LM1B test split using the model's BERT tokenizer and 128-token packing scheme, while retaining the same special-token formatting in both corpora. We extracted GPT-2-large features in FP32 from the final hidden state at the last non-padding token, using a maximum sequence length of 256 tokens.

We computed MAUVE on nested subsets of $n \in 500, 1000, 2000, 5000, 10000, 25000$ samples per corpus. We varied the number of clusters over $k \in 50, 100, 200, 500, 1000, 2500$ and additionally considered the automatic setting $k = n/10$, omitting fixed-k configurations for which $n < 2k$. We report means and standard deviations over three clustering seeds, using five k-means restarts and at most 500 iterations per restart.

E.2.3 Comparing MAUVE with SOL

We recap the different drawbacks with the use of the MAUVE score. Throughout the paper, we have seen the affects of different design choices on MAUVE.

Cluster stability. MAUVE appears to be heavily dependent on the number of clusters chosen. As shown in Figure 2, keeping a fixed ratio of clusters to samples decreases the MAUVE score as the sample size increases. In contrast when comparing true data in Figure 13, keeping the number of clusters fixed while increasing the number of samples causes the score to converge

to 1. This raises the question of which score to report and makes it hard to compare MAUVE scores. SOL, however, converges as we increase the number of samples (GPT-2 Large: Figure 7, Dream: Figure 9), the number of slices (GPT-2 Large: Figure 6, Dream: Figure 8), or the number of integration points (Figure 10). SOL is therefore numerically much more stable. In particular, inexpensive SOL estimates based on only a few samples and slices already provide relatively reliable approximations to the full evaluation, making them suitable for monitoring during training.

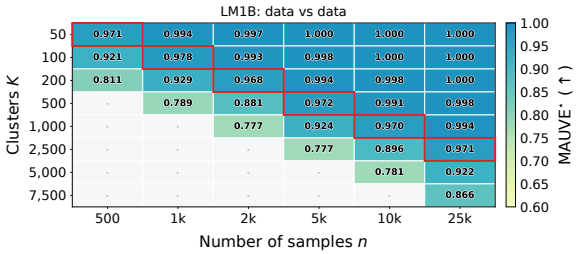


Figure 13: For two disjoint LM1B corpora, the reported MAUVE score depends on the number of evaluation samples n and clusters K. In comparison to generated vs true data the ratio $n/10 = K$ is stable, but increasing n converges to 1.

Effects of the hidden representation. MAUVE with the (standard) *last-token hidden representation* has a strong bias towards the end of a sequence. As shown in Figure 2, replacing the beginning with random tokens has little effect on the score. This is also visible in Section E.3.1 (Diffusion Language Model comparison), where the collapse of the last-token entropy leads to a significantly worse MAUVE score, even when the middle remains comparable. With a *mean-pooled representation*, repeating tokens 10% of the time produces a MAUVE score of almost 0, indicating a very poor match, see Table 16. For the last-token hidden representation, the same test yields a MAUVE score of 0 at repetition rates above 50% (Table 8). In the same experiment, SOL shows an almost linear degradation (Figure 11).

Human evaluation. We use the human evaluation experiment from Pillutla et al. (2021), see Section C.4. At sequence lengths of 256 and 1024 tokens (Tables 15 and 2), SOL outperforms MAUVE for almost all embedding models. This indicates that SOL correlates better with human judgments than MAUVE.

E.3 OpenWebText complete Model Table

We denote discrete-diffusion sampling steps by T and continuous predictor steps by S. NFE counts actual neural-network evaluations, including calls for correction, initialization, final decoding, and guidance. Consequently, S and T need not equal NFE. The SOL parameters L_g, L_ξ, and M specify metric estimation and are independent of generation cost.

E.3.1 MAUVE Depends on Last Hidden Tokens

Duo. Comparing MAUVE and SOL on OWT reveals a substantial disagreement for DUO's greedy-tail sampler. To investigate, we measure token entropy across documents at a selected position:

$$H_{\text{pos}}(t) = -\sum_{v\in\mathcal{V}} p_t(v)\log p_t(v), \qquad p_t(v) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left\{x_i^{(t)} = v\right\},$$

where $N = 5000$ and $x_i^{(t)}$ denotes the token at the selected position in document i. For the endpoint measurement, we select each document's last valid token. This differs from the main table's entropy, which averages token-frequency entropy computed within each document. Greedy DUO's positional entropy falls from 6.32 nats at the middle position to 5.01 at the endpoint, whereas real OWT remains near 6.58 at both positions, see Table 17.

Model	Middle	Final
Real OWT	6.578	6.572
DUO ancestral	6.455	6.441
DUO greedy	6.317	**5.011**

Table 17: Across-document token entropy H_{pos} in nats ($N = 5000$), measured at position 512 and the last valid token.

To test whether the final-token distribution drives MAUVE, we reweight the generated documents by their last token. Overrepresented final tokens get less weight, underrepresented ones get more. This makes the weighted final-token frequencies match real OWT without changing the

texts. After this correction, MAUVE rises from 0.492 to 0.591 for greedy DUO, but only from 0.836 to 0.843 for ancestral DUO. SOL changes by only 0.29% and 0.07%, respectively. This shows that MAUVE is much more sensitive to the final-token distribution than SOL.

For the implementation of our reweighting, we group documents by their final token. We treat each final-token ID that appears at least five times in both corpora as a separate category. All remaining tokens are pooled into an "other" category. Each generated document i gets weight $w_i = p_{c(i)}/q_{c(i)}$, where p_c and q_c are the final-token frequencies in the reference and generated corpora. We keep all features and cluster assignments fixed. We then recompute MAUVE with weighted generated-cluster counts and the original reference histogram. This changes only the document weights, not the texts.

	MAUVE (↑)		SOL (↓)	
Model	Base	Reweighted	Base	Reweighted
DUO ancestral	0.836	0.843	1.7431	1.7418
DUO greedy	0.492	0.591	1.8592	1.8538

Table 18: Final-token reweighting on matched corrected inputs ($N = 5000$). GPT-2-large SOL values average nine projection-seed combinations.

After matching the final-token frequencies to real OWT, MAUVE improves from 0.492 to 0.591 for greedy DUO. For ancestral DUO, it changes only from 0.836 to 0.843, see Table 18. This shows that MAUVE's positional bias toward the last token directly affects its evaluation of generative models. SOL is much less affected because it uses hidden states from all token positions.

ELF. We evaluate ELF-M and ELF-L with shared sampling settings (SDE, 64 steps, $\gamma = c = 1$), using 5,000 texts per distribution and GPT-2-large MAUVE features at a 1,024-token cap. We remove the final eight tokens from each generated and reference sequence, retaining one token for the three references shorter than nine tokens. MAUVE uses the hidden state at the resulting endpoint. This reduces the M–L score gap from 0.168 to 0.024, while the real–real control remains similar (0.962 versus 0.965).

Model	Original	Remove final 8
ELF-M	0.455 ± 0.011	0.823 ± 0.017
ELF-L	0.287 ± 0.006	0.798 ± 0.013

Table 19: MAUVE (↑) for shared ELF sampling. Truncation removes eight tokens from both generated and reference texts. Mean ± SD over three clustering seeds.

The tokens at the ends of the texts change much more. For example, the fraction of L texts ending with the period token falls from 54.1% to 2.7%. MAUVE extracts each text's representation at this final position. Its large score change therefore demonstrates sensitivity to where the text ends, despite almost unchanged token frequencies overall, see Table 19.

E.3.2 Extended Table

The complete OWT table is given in Table 22.

Increasing the number of steps. Increasing the sampling budget generally improves SOL, but the changes in Gen. PPL and entropy can be much larger. For LangFlow, increasing NFE from 128 to 1024 improves Dream SOL from 32.488 to 30.541 and Gen. PPL from 59.0 to 36.9, while entropy moves away from the data value, from 5.425 to 5.257. Similarly, ELF-B with ODE sampling improves Dream SOL only from 33.677 to 33.320 between 128 and 1024 steps, despite Gen. PPL falling from 45.4 to 24.0 and entropy moving from 5.308 to 5.090.

Changing the sampler and guidance. Sampler choice substantially affects distributional quality. For Soft-Masked, replacing MDLM sampling with ReMDM-loop reduces Dream SOL from 21.643 to 18.966 and Gen. PPL from 23.6 to 16.7, despite a small increase in the entropy gap to the data. For vMF at 1024 steps, switching from ODE to predictor–corrector sampling improves Dream SOL from 33.832 to 30.583 and Gen. PPL from 75.4 to 56.3, while entropy remains close to the data value (5.456 versus 5.432). Guidance similarly improves ELF-B at 32 SDE steps, reducing Dream SOL from 39.144 to 37.584 and Gen. PPL from 29.4 to 24.0.

Self-conditioning. The vMF comparison provides direct evidence for the benefit of self-conditioning. At the same 128-step predictor–corrector setting, the base self-conditioned model improves Dream SOL from 45.900 to 31.379 and Gen. PPL from 141.9 to 45.1. This improvement is not explained by entropy matching, which changes from 5.533 to 5.315 and therefore moves slightly farther from the data entropy. The distilled ELF* model still exhibits a gap to ELF-B, although this comparison does not isolate self-conditioning from other model and training differences.

Comparison with autoregressive generation. Soft-Masked with ReMDM-loop improves over the AR baseline on all three SOL encoders, from $(1.064, 7.450, 22.207)$ to $(0.6101, 3.526, 18.966)$. It also has lower Gen. PPL (16.7 versus 34.9) and entropy closer to the data (5.341 versus 5.582). LDDM-U, however, illustrates that these criteria need not agree: it has both lower Gen. PPL (28.2) and entropy closer to the data (5.409), but improves over AR only under GPT-2-large and OLMo, while Dream SOL is higher (24.209 versus 22.207). Among the evaluated continuous models, self-conditioned vMF with predictor–corrector sampling achieves the lowest SOL, but remains above AR on all three encoders.

Few-step language generation. Within FMLM*, additional CTM steps substantially improve generation while keeping entropy close to the ELF data distribution. At $w = 5$, increasing from one to five steps lowers Gen. PPL from 116.0 to 70.8 and Dream SOL from 45.239 to 39.691, with entropy remaining near the reference value (5.368 and 5.415). Further computation gives diminishing returns: increasing flow-map sampling from 8 to 128 steps changes Gen. PPL only from 82.8 to 79.0 and leaves Dream SOL essentially unchanged (37.512 versus 37.661).

Perplexity, entropy, and distributional quality can disagree. Gen. PPL and entropy should be interpreted jointly: Gen. PPL should be small, while entropy should remain close to that of the data. Even together, however, they do not determine distributional quality. For distilled DUO, ancestral and greedy-tail sampling have almost identical entropy deviations from the data ($|5.518 - 5.449| = 0.069$ and $|5.378 - 5.449| = 0.071$). Greedy-tail sampling nevertheless lowers Gen. PPL substantially, from 62.3 to 46.5, while worsening SOL on all three encoders, including Dream from 30.595 to 31.802. More strongly, exact MDLM has both lower Gen. PPL than VP with ODE sampling (59.4 versus 66.1) and entropy closer to the data (5.445 versus 5.435), yet has worse SOL on all three encoders: $(2.185, 14.119, 36.394)$ versus $(1.775, 11.997, 33.275)$. Thus, even favorable Gen. PPL and entropy do not guarantee a better match to the text distribution.

E.4 Additional Sampler

We investigate how sampling strategies affect GPT-2-XL's agreement with the WebText distribution. We compare ancestral sampling, which draws directly from the model's next-token distribution, and nucleus sampling with $p = 0.95$ (Holtzman et al., 2020), which restricts sampling to the smallest set of highest-probability tokens with cumulative probability at least p, renormalizing their probabilities. We also evaluate greedy decoding, which deterministically selects the most likely token, in the conditional setting.

Conditional generation. We use $5{,}000$ WebText documents, providing the first 35 GPT-2 tokens of each document as context. Generation stops at the first $\langle\mathrm{EOT}\rangle$ token or when the total sequence length, including the prefix, reaches 256 tokens. All three samplers receive identical prefixes and are evaluated against WebText at the same horizon.

Sampler	MAUVE (↑)	SOL (↓)
Greedy	0.0697 ± 0.0012	3.7755 ± 0.1190
Ancestral	0.9407 ± 0.0028	0.5083 ± 0.0113
Nucleus ($p = 0.95$)	$\mathbf{0.9607 \pm 0.0017}$	$\mathbf{0.2712 \pm 0.0065}$

Table 20: GPT-2-XL conditional completion on $5{,}000$ WebText documents, using shared 35-token prefixes and a maximum total sequence length of 256 tokens. SOL uses GPT-2-large.

Both metrics favor nucleus sampling (Table 20): SOL decreases from 0.5083 to 0.2712, while MAUVE increases from 0.9407 to 0.9607. Thus, nucleus truncation improves agreement with Web-

Text in this conditional setting. Greedy decoding performs substantially worse than both stochastic samplers, with MAUVE of 0.0697 and SOL of 3.7755.

Unconditional generation. We initialize GPT-2-XL with $\langle\text{EOT}\rangle$ and generate until the first subsequently generated $\langle\text{EOT}\rangle$ token or a maximum of 256 generated tokens. With this fixed initial context, greedy decoding always produces the same continuation, yielding a fully collapsed generated distribution.

Sampler	MAUVE (↑)	SOL (↓)
Ancestral	$\mathbf{0.8832 \pm 0.0071}$	$\mathbf{0.8358 \pm 0.0245}$
Nucleus ($p = 0.95$)	0.8740 ± 0.0073	0.9072 ± 0.0279

Table 21: GPT-2-XL unconditional generation evaluated against $5{,}000$ WebText documents. Each of the $5{,}000$ samples starts from the end-of-text token and stops at EOS or 256 generated tokens, the initial seed token is excluded from evaluation. SOL uses GPT-2-large.

The ordering reverses in the unconditional setting (Table 21). Ancestral sampling achieves lower SOL (0.8358 versus 0.9072) and higher MAUVE (0.8832 versus 0.8740) than nucleus sampling.

Overall, SOL and MAUVE agree on the sampling-strategy ordering in both settings. Nucleus sampling improves conditional completion, whereas ancestral sampling achieves better unconditional agreement with WebText in this experiment.

F Implementation Details

F.1 Numerical implementation of SOL

Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^{n_x}$ and $\mathcal{Y} = \{\mathbf{y}_j\}_{j=1}^{n_y}$ denote two corpora with empirical text distributions

$$\widehat{p}_{\mathcal{X}} = \frac{1}{n_x}\sum_{i=1}^{n_x}\delta_{\mathbf{x}_i}, \qquad \widehat{p}_{\mathcal{Y}} = \frac{1}{n_y}\sum_{j=1}^{n_y}\delta_{\mathbf{y}_j}.$$

Following Section 4.3, every sequence is mapped by a frozen reference transformer $\mathbf{H}^\theta$ to its token cloud $T^\theta(\mathbf{x}) \in \mathcal{P}_2(\mathbb{R}^d)$.

We write $\widetilde{h}^\theta_{\mathbf{x},r} = h^\theta_{\mathbf{x}_{\le r}}$ for a causal reference transformer and $\widetilde{h}^\theta_{\mathbf{x},r} = h^{\theta,\mathrm{bi}}_{\mathbf{x},r}$ for a bidirectional reference transformer. We retain all non-padding token positions up to the encoder horizon T and denote their number by t_x.

The exact distance

$$\mathrm{SOL}^2_\theta(\widehat{p}_{\mathcal{X}}, \widehat{p}_{\mathcal{Y}}) = d\,\mathrm{DSW}^2\left(T^\theta_\#\widehat{p}_{\mathcal{X}}, T^\theta_\#\widehat{p}_{\mathcal{Y}} \mid \gamma\right)$$

is approximated using L_ξ first-level directions, L_g second-level directions, and resolution M for the second-level slicing functions. The same sampled directions are used for both corpora.

First slicing. We sample $a_\ell \sim \mathcal{N}(0, I_d)$ and set $\xi_\ell = a_\ell/\|a_\ell\|$ for $\ell = 1, \dots, L_\xi$, such that $\xi_\ell \sim \mathcal{U}(\mathbb{S}^{d-1})$. For each sequence $\mathbf{x}$,

$$\mathbf{e}_{\xi_\ell} T^\theta(\mathbf{x}) = \frac{1}{t_x}\sum_{r=1}^{t_x}\delta_{v_{x,\ell,r}}, \qquad v_{x,\ell,r} = \left\langle \widetilde{h}^\theta_{\mathbf{x},r}, \xi_\ell \right\rangle. \tag{18}$$

Let $v_{x,\ell,(1)} \le \cdots \le v_{x,\ell,(t_x)}$ denote the corresponding order statistics.

Bin-averaged empirical quantiles. We partition $[0,1]$ into $I_m = ((m-1)/M, m/M]$ with midpoint $s_m = (m - \frac{1}{2})/M$ for $m = 1, \dots, M$. For every projected token cloud, we compute the exact average of its empirical inverse-CDF quantile within each bin,

$$\bar{q}_{x,\ell,m} = M\int_{I_m} Q_{\mathbf{e}_{\xi_\ell} T^\theta(\mathbf{x})}(s)\,\mathrm{d}s = M\sum_{r=1}^{t_x} v_{x,\ell,(r)}\left|I_m \cap \left(\frac{r-1}{t_x}, \frac{r}{t_x}\right]\right|. \tag{19}$$

The bin averages are computed efficiently from the sorted projected values using cumulative sums. This construction preserves empirical mass exactly, since

$$\frac{1}{M}\sum_{m=1}^{M}\bar{q}_{x,\ell,m}=\frac{1}{t_x}\sum_{r=1}^{t_x}v_{x,\ell,r}.$$

Second slicing. We draw the second-level slicing functions from an RBF Gaussian process with covariance $k_\lambda(s,t)=\exp(-(s-t)^2/(2\lambda^2))$. On the midpoint grid, we define $C_{ab}=k_\lambda(s_a,s_b)$ for $a,b=1,\dots,M$ and compute $RR^\top=C+\varepsilon I$ by Cholesky factorization, with a small diagonal jitter ε for numerical stability.

For $\tilde{\ell}=1,\dots,L_g$, we sample $\omega_{\tilde{\ell}}\sim\mathcal{N}(0,I_M)$ and set $g_{\tilde{\ell}}=R\omega_{\tilde{\ell}}$. Thus $g_{\tilde{\ell}}$ follows the finite-dimensional Gaussian-process distribution on the grid. We interpret it as the piecewise-constant function $g_{\tilde{\ell}}^{(M)}(s)=g_{\tilde{\ell},m}$ for $s\in I_m$.

The L_g directions are sampled independently of $\xi_1,\dots,\xi_{L_\xi}$ and reused for all first-level directions. Hence, we evaluate all $L_\xi L_g$ pairs $(\xi_\ell,g_{\tilde{\ell}})$.

For every sequence and pair of directions, we compute

$$z_{x,\ell,\tilde{\ell}}=\widehat{M}^{(M)}_{\xi_\ell,g_{\tilde{\ell}}}\big(T^\theta(\mathbf{x})\big)=\frac{1}{M}\sum_{m=1}^{M}\bar{q}_{x,\ell,m}g_{\tilde{\ell},m}=\int_0^1 Q_{\mathbf{e}_{\xi_\ell}T^\theta(\mathbf{x})}(s)g_{\tilde{\ell}}^{(M)}(s)\,\mathrm{d}s. \tag{20}$$

Thus the empirical quantile is integrated exactly against the piecewise-constant second-level slicing function.

One-dimensional Wasserstein distance. For fixed $(\ell,\tilde{\ell})$, define

$$\widehat{\mu}^{\mathcal{X}}_{\ell,\tilde{\ell}}=\frac{1}{n_x}\sum_{i=1}^{n_x}\delta_{z_{x_i,\ell,\tilde{\ell}}},\qquad \widehat{\mu}^{\mathcal{Y}}_{\ell,\tilde{\ell}}=\frac{1}{n_y}\sum_{j=1}^{n_y}\delta_{z_{y_j,\ell,\tilde{\ell}}}.$$

Let $z^{\mathcal{X}}_{(i),\ell,\tilde{\ell}}$ and $z^{\mathcal{Y}}_{(j),\ell,\tilde{\ell}}$ denote their order statistics. If $n_x=n_y=n$, then

$$\mathrm{W}^2\left(\widehat{\mu}^{\mathcal{X}}_{\ell,\tilde{\ell}},\widehat{\mu}^{\mathcal{Y}}_{\ell,\tilde{\ell}}\right)=\frac{1}{n}\sum_{i=1}^{n}\left(z^{\mathcal{X}}_{(i),\ell,\tilde{\ell}}-z^{\mathcal{Y}}_{(i),\ell,\tilde{\ell}}\right)^2. \tag{21}$$

All main experiments use equal numbers of reference and generated samples.

For completeness, the implementation also supports unequal corpus sizes. Let $N=\max(n_x,n_y)$ and $r_a=(a-\frac{1}{2})/N$ for $a=1,\dots,N$. We use

$$\widehat{\mathrm{W}}^2_N=\frac{1}{N}\sum_{a=1}^{N}\left(z^{\mathcal{X}}_{(\lceil n_x r_a\rceil),\ell,\tilde{\ell}}-z^{\mathcal{Y}}_{(\lceil n_y r_a\rceil),\ell,\tilde{\ell}}\right)^2,$$

which reduces to (21) when $n_x=n_y$.

Monte Carlo estimator. Combining the two slicing levels gives

$$\widehat{\mathrm{SOL}}^2_{\theta,L_\xi,L_g,M}(\mathcal{X},\mathcal{Y})=\frac{\alpha}{L_\xi L_g}\sum_{\ell=1}^{L_\xi}\sum_{\tilde{\ell}=1}^{L_g}\widehat{\mathrm{W}}^2_N\left(\widehat{\mu}^{\mathcal{X}}_{\ell,\tilde{\ell}},\widehat{\mu}^{\mathcal{Y}}_{\ell,\tilde{\ell}}\right), \tag{22}$$

where $\alpha>0$ is a fixed scaling factor. We use $\alpha=d$, which compensates for

$$\mathbb{E}_{\xi\sim\mathcal{U}(\mathbb{S}^{d-1})}\left[\langle v,\xi\rangle^2\right]=\frac{\|v\|^2}{d}.$$

Since α is fixed for a given reference model, it does not affect the ranking of generated corpora.

Algorithm 2 Numerical approximation of SOL

Require: Corpora $\mathcal{X}, \mathcal{Y}$, frozen reference transformer $\mathbf{H}^\theta$, horizon T
Require: Number of slices L_ξ, L_g, resolution M, GP length scale λ, scale α
Ensure: $\widehat{\mathrm{SOL}}^2_{\theta,L_\xi,L_g,M}(\mathcal{X},\mathcal{Y})$

```
Construct bins I_1, ..., I_M and midpoints s_1, ..., s_M
Sample ξ_1, ..., ξ_{L_ξ} ~ U(S^{d-1})
Sample RBF-GP directions g_1, ..., g_{L_g} on {s_m}_{m=1}^M
for all x ∈ X ∪ Y do
    Compute the retained hidden states of H^θ(x) up to horizon T
    for ℓ = 1, ..., L_ξ do
        Project and sort the token states using (18)
        Compute q̄_{x,ℓ,1:M} using (19)
        for ℓ̃ = 1, ..., L_g do
            Compute z_{x,ℓ,ℓ̃} using (20)
        end for
    end for
end for
for ℓ = 1, ..., L_ξ and ℓ̃ = 1, ..., L_g do
    Sort the document scalars in X and Y
    Compute the one-dimensional Wasserstein term
end for
return (22)
```

Precision and memory. The reference transformer is evaluated in batches. Hidden states are cast to FP32 before projection. Token projections, bin-average computation, second-level projections, sorting, and Wasserstein reductions are performed in FP32. The Gaussian-process covariance and Cholesky factorization are computed in FP64, after which the sampled GP values are cast to FP32.

For each first-level projection seed, the bin-average tensors of shape $n \times L_\xi \times M$ are stored as FP32 memory-mapped arrays. The second-level computation is performed in blocks of first-level directions and reuses the reference tensors across generated corpora. Thus neither all $nL_\xi L_g$ document scalars nor a pairwise document cost matrix need to be stored simultaneously. Changing only the GP seed requires no additional transformer evaluation.

Model	Setting	H@1024	Gen. ppl.	SOL (↓) GPT-2-large	OLMo-2-1B	Dream-7B
Real OWT[G]	—	5.449	14.9	0.2002 ± 0.0116	0.7800 ± 0.0552	2.066 ± 0.109
Real ELF OWT[E]	—	5.402	17.0	0.2457 ± 0.0239	0.8857 ± 0.0636	2.649 ± 0.219
Autoregressive model Sahoo et al. (2024)						
AR[G]	Temperature 1.0	5.582	34.9	1.064 ± 0.008	7.450 ± 0.132	22.207 ± 0.099
Discrete diffusion						
DUO Sahoo et al. (2025)						
DUO-distilled, $T = 32$[G]	Ancestral	5.518	62.3	1.670 ± 0.019	11.063 ± 0.228	30.595 ± 0.082
DUO-distilled, $T = 32$[G]	Greedy tail	5.378	46.5	1.748 ± 0.024	11.729 ± 0.253	31.802 ± 0.242
DUO-base[G]	Ancestral, $T = 1024$	5.537	76.9	1.948 ± 0.024	12.730 ± 0.250	32.642 ± 0.084
MDLM Sahoo et al. (2024)						
MDLM, exact sampling[G]	DDPM-cache, $T = 1000$	5.445	59.4	2.185 ± 0.028	14.119 ± 0.246	36.394 ± 0.086
MDLM, as released[G]	FP32 Gumbel, $T = 1000$	5.292	42.8	2.340 ± 0.035	14.275 ± 0.238	38.444 ± 0.112
LDDM Jo et al. (2026)						
LDDM-M[G]	Loopholing, $T = 1024$	5.536	48.2	1.082 ± 0.011	8.266 ± 0.160	24.601 ± 0.038
LDDM-U[G]	Loopholing, $T = 1024$	5.409	28.2	1.009 ± 0.034	6.851 ± 0.168	24.209 ± 0.095
Soft-Masked Hersche et al. (2026)						
Soft-Masked[G]	MDLM, $T = 1024$, $p = 0.9$	5.364	23.6	0.7415 ± 0.0090	5.390 ± 0.110	21.643 ± 0.103
Soft-Masked[G]	ReMDM-loop, $T = 1024$, $p = 0.9$	5.341	16.7	$\mathbf{0.6101 \pm 0.0135}$	$\mathbf{3.526 \pm 0.066}$	$\mathbf{18.966 \pm 0.126}$
ADLM Rout et al. (2025)						
ADLM-large[G]	ReMDM-loop, $T = 1024$, $p = 0.9$	5.352	28.9	1.183 ± 0.014	7.534 ± 0.135	23.601 ± 0.088
BD3-LM Arriola et al. (2025)						
BD3-LM, block 4[B]	First-hitting, $T = 5000$, $p = 0.9$	5.336	24.4	1.082 ± 0.023	8.407 ± 0.180	26.070 ± 0.057
BD3-LM, block 16[B]	First-hitting, $T = 5000$, $p = 0.9$	5.344	32.9	1.458 ± 0.027	10.709 ± 0.218	30.315 ± 0.091
Continuous diffusion						
vMF Chemseddine et al. (2026)						
vMF, self-cond.[G]	PC-warp, $S = 128$	5.222	30.2	1.354 ± 0.034	8.607 ± 0.197	29.225 ± 0.089
vMF, self-cond. (base)[G]	PC-warp, $S = 128$	5.315	45.1	1.524 ± 0.022	10.657 ± 0.221	31.379 ± 0.076
vMF, self-cond.[G]	SDE-warp, $S = 128$, $\sigma = 0.01$, $q = 1$	5.348	43.6	1.386 ± 0.022	9.884 ± 0.196	30.375 ± 0.074
vMF, no self-cond.[G]	PC-warp, $S = 128$	5.533	141.9	2.928 ± 0.036	18.157 ± 0.343	45.900 ± 0.130
vMF[G]	ODE-warp, $S = 1024$	5.456	75.4	1.901 ± 0.024	12.781 ± 0.268	33.832 ± 0.113
vMF[G]	PC-warp, $S = 1024$	5.432	56.3	1.546 ± 0.021	10.777 ± 0.233	30.583 ± 0.099
VP Chemseddine et al. (2026)						
VP, self-cond.[G]	SDE-warp, $S = 128$, $\sigma = 0.1$, $q = 1$	5.355	50.9	1.606 ± 0.023	10.898 ± 0.222	31.972 ± 0.072
VP[G]	ODE-warp, $S = 1024$	5.435	66.1	1.775 ± 0.022	11.997 ± 0.247	33.275 ± 0.076
VP[G]	PC-warp, $S = 1024$	5.430	59.0	1.642 ± 0.022	11.168 ± 0.232	31.664 ± 0.068
LangFlow Chen et al. (2026)						
LangFlow[G]	NFE= 128	5.425	59.0	1.604 ± 0.018	11.930 ± 0.226	32.488 ± 0.066
LangFlow[G]	NFE= 1024	5.257	36.9	1.371 ± 0.019	9.886 ± 0.193	30.541 ± 0.071
ELF Hu et al. (2026)						
ELF-B[E]	SDE, 32 steps, CFG= 3	5.156	24.0	2.066 ± 0.055	12.817 ± 0.314	37.584 ± 0.218
ELF-B[E]	SDE, 32 steps, CFG= 1	5.219	29.4	2.283 ± 0.057	13.633 ± 0.332	39.144 ± 0.232
ELF-B[E]	ODE, 128 steps, CFG= 1	5.308	45.4	1.874 ± 0.029	13.565 ± 0.295	33.677 ± 0.116
ELF-B[E]	ODE, 1024 steps, CFG= 1	5.090	24.0	1.778 ± 0.031	12.357 ± 0.292	33.320 ± 0.188
ELF-B[E]	SDE, 128 steps, CFG= 1	5.123	21.9	1.654 ± 0.045	10.980 ± 0.257	32.878 ± 0.260
ELF-B[E]	SDE, 64 steps, CFG= 0.5, $\gamma = 0.5$	5.385	56.3	2.181 ± 0.035	14.796 ± 0.318	35.648 ± 0.130
ELF-M[E]	ODE, 64 steps, CFG= 1.25, $\gamma = 0$	5.399	53.0	1.695 ± 0.023	12.805 ± 0.268	32.153 ± 0.099
ELF-L[E]	SDE, 64 steps, CFG= 1, $\gamma = 0.75$	5.398	35.7	1.591 ± 0.045	10.039 ± 0.232	32.013 ± 0.274
ELF-B[E]	SDE, 64 steps, CFG= 1, $\gamma = 1$	5.216	28.5	1.788 ± 0.043	11.952 ± 0.271	33.689 ± 0.228
ELF-M[E]	SDE, 64 steps, CFG= 1, $\gamma = 1$	5.309	33.4	1.665 ± 0.033	11.719 ± 0.261	32.671 ± 0.205
ELF-L[E]	SDE, 64 steps, CFG= 1, $\gamma = 1$	5.378	32.9	1.601 ± 0.048	9.829 ± 0.227	31.969 ± 0.290
Few-Step Models						
FLM Lee et al. (2026)						
FLM[G]	Euler, 1024 steps	5.345	63.1	3.458 ± 0.068	20.126 ± 0.387	48.387 ± 0.189
FMLM[G]	1 step	5.168	167.8	6.857 ± 0.100	30.806 ± 0.647	79.042 ± 0.658
FMLM[G]	2 steps	5.295	135.1	5.683 ± 0.085	27.826 ± 0.544	71.134 ± 0.494
FMLM[G]	4 steps	5.361	111.0	4.743 ± 0.070	25.303 ± 0.492	63.074 ± 0.409
*FMLM** Yoo et al. (2026)						
FMLM*[G]	CTM, 1 step, CFG= 1, $\gamma = 1$	5.319	120.3	3.690 ± 0.054	18.808 ± 0.373	48.859 ± 0.169
FMLM*[G]	CTM, 1 step, CFG= 5, $\gamma = 1$	5.368	116.0	3.431 ± 0.046	18.160 ± 0.382	46.273 ± 0.099
FMLM*[G]	CTM, 2 steps, CFG= 1, $\gamma = 0.75$	5.411	99.5	2.932 ± 0.042	16.491 ± 0.320	43.430 ± 0.195
FMLM*[G]	CTM, 2 steps, CFG= 5, $\gamma = 0.75$	5.458	97.4	2.737 ± 0.035	15.941 ± 0.322	41.120 ± 0.139
FMLM*[G]	CTM, 5 steps, CFG= 1, $\gamma = 1$	5.294	66.5	2.860 ± 0.054	15.556 ± 0.295	43.731 ± 0.188
FMLM*[G]	CTM, 5 steps, CFG= 5, $\gamma = 1$	5.415	70.8	2.486 ± 0.042	14.869 ± 0.291	40.177 ± 0.114
FMLM*[G]	Flow map, 8 steps, CFG= 5, $\gamma = 0$	5.464	82.8	2.370 ± 0.032	14.438 ± 0.295	38.233 ± 0.158
FMLM*[G]	Flow map, 128 steps, CFG= 5, $\gamma = 0$	5.456	79.0	2.313 ± 0.032	14.268 ± 0.285	38.482 ± 0.152
ELF*[G]	CDEQ, 128 steps, CFG= 1, $\gamma = 0$	5.414	73.8	2.222 ± 0.034	14.245 ± 0.267	37.827 ± 0.128
ELF*[G]	CDEQ, 128 steps, CFG= 5, $\gamma = 0$	5.384	60.8	1.952 ± 0.029	13.105 ± 0.252	35.950 ± 0.103

Table 22: OpenWebText baselines using 5,000 samples per distribution and a 1,024-token horizon. The standard SOL hyperparameters are used; see Section C.3. Superscripts G, E and B identify the evaluation reference family. SOL is reported as mean ± population std over nine crossed direction/GP seed pairs.