Riemannian Flow Matching on the Fisher–Rao Sphere for Non-Autoregressive Conditional Text Generation

Anonymous ACL submission

Abstract

Diffusion models and Linear Flow matching have emerged as a promising framework for fast and high-quality conditional text generation, yet, current approaches often overlook the inherent geometric structure of text embeddings. In this work, we introduce *GeoFM*, a novel flow matching model that directly leverages the Riemannian geometry induced by the Fisher–Rao metric. Specifically, GeoFM projects token embeddings onto a Fisher–Rao sphere via the square-root transform, and learns a neural velocity field that precisely aligns with spherical geodesics connecting noisy priors and target embeddings. Additionally, we propose a spherical trajectory loss that maintains lexical fidelity and encourages direct, minimally-distorted trajectories on the manifold. Our empirical evaluation demonstrates GeoFM's effectiveness and significant speedups over state-of-the-art non-autoregressive baselines.

1 Introduction

Neural sequence-to-sequence learning has achieved remarkable success in text generation tasks such as machine translation and summarization, typically relying on autoregressive (AR) decoders to maintain high fidelity at the cost of slow, step-by-step sampling (Vaswani et al., 2017). Non-autoregressive and diffusion-based approaches aim to parallelize generation, but they either sacrifice quality or require hundreds of model evaluations (Song et al., 2021; Li et al., 2022). Recent progress in flow matching and continuous-time normalizing flows suggests that one can directly learn vector fields whose integral curves transport simple priors to complex data distributions in very few steps (Lipman et al., 2023; Neklyudov et al., 2022; Hu et al., 2024).

However, text embeddings naturally live on a probability simplex, for which the Fisher–Rao metric induces a spherical geometry. Ignoring this manifold structure can lead to suboptimal trajectories and embedding collapse. To address this, we propose *GeoFM* on the Fisher–Rao sphere. We map both noisy Gaussian samples and data embeddings to the sphere via the square-root transform, learn a neural velocity field v_θ that matches the exact spherical geodesic in tangent space (Eq. 3), and add a spherical *trajectory loss* to preserve token identity. We acknowledge that text is inherently discrete, hence modeling discrete distributions with our flow-based **GeoFM** models can be challenging and may require compromises that lose some of the benefits, like fast sampling. Inspired by prior studies (Li et al., 2022; Gao et al., 2022), we choose to model the problem in continuous text embedding space.

We also introduce a spherical trajectory loss to prevent embedding collapse and ensure lexical fidelity when reconstructing tokens from one-step estimates. We show that a single Euler-step sampler on the learned spherical flow achieves near autoregressive BLEU and ROUGE scores, with much faster decoding compared to existing non-autoregressive flows and diffusion models.

2 Related Work

Diffusion Models and Flow Matching. Diffusion approaches have recently gained traction in NLP by eschewing autoregressive generation (Zou et al., 2023). Broadly, they fall into two streams: *discrete diffusion*, which perturbs tokens directly (Hoogeboom et al., 2022; Chen et al., 2023), and *embedding diffusion*, which diffuses continuous token or sentence embeddings (Li et al., 2022; Dieleman et al., 2022; Gao et al., 2022). Embedding-level methods often outperform token-level ones owing to faster parallel sampling, smoother latent interpolation, and improved robustness (Zou et al., 2023). A key challenge is preventing embeddings from collapsing; for instance, Difformer introduces a

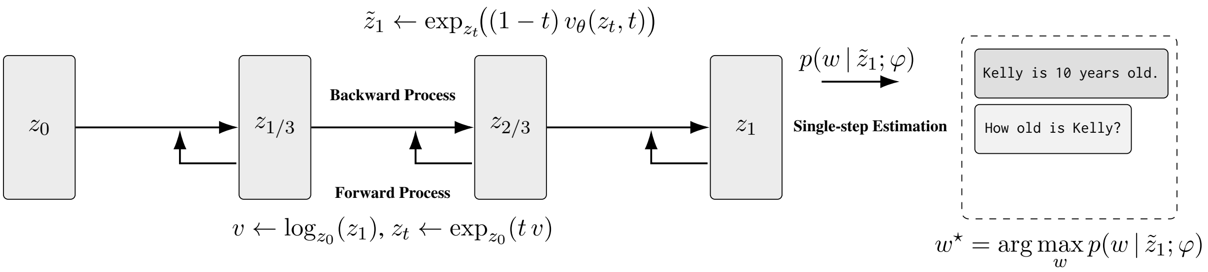
$\tilde{z}_1 \leftarrow \exp_{z_t}\big((1-t)\, v_\theta(z_t,t)\big)$
z_0
$z_{1/3}$
Backward Process
$z_{2/3}$
z_1
Forward Process
$v \leftarrow \log_{z_0}(z_1),\, z_t \leftarrow \exp_{z_0}(t\, v)$
$p(w \mid \tilde{z}_1; \varphi)$
Single-step Estimation
Kelly is 10 years old.
How old is Kelly?
$w^\star = \arg\max_w p(w \mid \tilde{z}_1; \varphi)$

Figure 1: **GeoFM framework**. Embeddings are projected onto the Fisher–Rao sphere, uniformly *corrupted* (forward process) and then *recovered* (backward process) by a learned velocity field. A single Euler step yields $\tilde{z}_1$, which is decoded to a sequence via $\arg\max$ over the softmax output.

trajectory loss to mitigate this issue (Gao et al., 2022).

Although diffusion models achieve strong quality, they typically need hundreds of sampling steps. Techniques such as DDIM (Song and Sohl-Dickstein, 2021), FastDPM (Kong and Ping, 2021), and several knowledge-distillation variants (Luhman and Luhman, 2021; Salimans and Ho, 2022; Gu et al., 2023; Song et al., 2023; Tong et al., 2023) reduce inference cost but still require multiple evaluations. In contrast, we build on recent *flow-matching* ODE formulations (Lipman et al., 2023; Liu et al., 2023; Neklyudov et al., 2022; Hu et al., 2024), learning a velocity field that transports a spherical Gaussian to data in a *single* Euler step.

3 Method

Problem Statement

Let the source sequence be $\mathbf{w}_x = (w_1^x, \dots, w_M^x)$ and the target sequence $\mathbf{w}_y = (w_1^y, \dots, w_N^y)$. We concatenate them to obtain $\mathbf{w} = \mathbf{w}_x \oplus \mathbf{w}_y$ of length $L = M + N$. A learnable embedding map $\mathrm{EMB}(\cdot;\varphi) : \mathcal{V} \to \mathbb{R}^D$ then produces

$$\mathbf{z}_1 = \mathrm{EMB}(\mathbf{w};\varphi) \in \mathbb{R}^{L \times D}.$$

Fisher–Rao Manifold Embedding

Applying the square-root transform places every embedding on the D-dimensional Fisher–Rao sphere $\mathbb{S}^{D-1}(R) = \{x \in \mathbb{R}^D : \|x\|_2 = R\}$. Any off-manifold vector x is projected via

$$\mathrm{proj}_{\mathbb{S}}(x) \;=\; R\,\frac{x}{\|x\|_2}.$$

Riemannian Flow Matching

Our goal is to learn a time-dependent velocity field $v_\theta : \mathbb{S}^{D-1}(R) \times [0,1] \longrightarrow T\mathbb{S}^{D-1}(R)$ that transports an isotropic Gaussian $z_0 \sim \mathcal{N}(0,I)$—after spherical projection—to the data point z_1 along a single geodesic.

Exp/Log maps. For a base point $x \in \mathbb{S}^{D-1}(R)$ and tangent vector $u \in T_x$,

$$\exp_x(u) = x\cos\Big(\tfrac{\|u\|}{R}\Big) + R\,\frac{u}{\|u\|}\sin\Big(\tfrac{\|u\|}{R}\Big), \qquad (1)$$

$$\log_x(y) = \frac{\theta}{\sin\theta}\big(y - \cos\theta\, x\big),\; \theta = \arccos\Big(\tfrac{\langle x,y\rangle}{R^2}\Big) \qquad (2)$$

Geodesic and target velocity. The unique shortest path from z_0 to z_1 is

$$z_t \;=\; \exp_{z_0}\big(t\,\log_{z_0}(z_1)\big), \qquad v = \log_{z_0}(z_1).$$

Flow-matching loss. We align the learned field with v through

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t\sim U[0,1]}\big\|v_\theta(z_t,t) - v\big\|^2. \qquad (3)$$

Trajectory Loss on the Sphere. To further regularise the path, we decode a one-step reconstruction $\tilde{z}_1 = \exp_{z_t}\big((1-t)\,v_\theta(z_t,t)\big)$ and compute a cross-entropy term

$$\mathcal{L}_{\mathrm{trajectory}}(\theta,\varphi) = -\log p_\varphi\big(w \mid \tilde{z}_1\big). \qquad (4)$$

Full objective. The combined training loss is therefore

$$\min_{\theta,\varphi}\; \mathcal{L}_{\mathrm{FM}}(\theta) + \lambda\,\mathcal{L}_{\mathrm{trajectory}}(\theta,\varphi). \qquad (5)$$

Single-Step Sampling

At inference time, we sample $z_0 \sim \mathcal{N}(0,I)$, project it onto the sphere, and then update once:

$$z_1^{\mathrm{sample}} = \exp_{z_0}\big(v_\theta(z_0,0)\big).$$

Greedy decoding of z_1^{sample} via *argmax* produces the final output sequence in a single network evaluation.

Tasks	Methods	NFE↓	BLEU↑	R-L↑	Score↑	dist-1↑	selfBL↓	div-4↑	Len
Open Domain Dialogue	Transformer-base	–	**0.018**	0.104	0.478	0.750	0.370	0.647	19.50
	GPT2-large FT	–	0.013	0.100	**0.529**	0.924	0.021	0.994	16.80
	GPVAE-T5	–	0.011	0.101	0.432	0.563	0.356	0.555	20.10
	NAR-LevT	–	0.016	0.055	0.476	**0.973**	**0.710**	0.142	4.11
	DiffuSeq	2 000	0.014	0.106	0.513	0.947	0.014	0.997	13.60
	FlowSeq	**1**	0.011	**0.119**	0.345	0.709	0.027	**0.999**	30.70
	GeoFM (Ours)	**1**	0.009	0.088	0.386	0.957	0.078	0.998	6.78
Question Generation	Transformer-base	–	0.166	0.344	0.631	0.931	0.327	0.772	10.30
	GPT2-large FT	–	0.111	0.322	**0.635**	**0.967**	0.291	0.806	9.96
	GPVAE-T5	–	0.125	0.339	0.631	0.938	0.357	0.728	11.40
	NAR-LevT	–	0.093	0.289	0.549	0.891	**0.983**	0.478	6.93
	DiffuSeq	2 000	0.173	0.366	0.612	0.905	0.279	0.810	11.50
	DiffuSeq	500	0.016	0.120	0.334	0.543	0.321	0.435	11.50
	FlowSeq	**1**	0.162	0.370	0.573	0.833	0.460	0.497	11.80
	GeoFM (Ours)	**1**	**0.181**	**0.388**	0.556	0.862	0.511	**0.834**	9.48
Paraphrase	Transformer-base	–	**0.272**	0.575	0.838	0.975	0.448	0.734	11.20
	GPT2-large FT	–	0.206	0.542	0.836	**0.982**	**0.733**	0.502	9.53
	GPVAE-T5	–	0.241	**0.589**	**0.847**	0.969	0.561	0.617	9.60
	NAR-LevT	–	0.227	0.580	0.834	0.979	0.999	0.333	8.85
	DiffuSeq	2 000	0.241	0.588	0.837	0.981	0.273	**0.864**	11.20
	FlowSeq	**1**	0.143	0.461	0.669	0.862	0.191	0.781	11.90
	GeoFM (Ours)	**1**	0.170	0.501	0.733	0.964	0.413	0.706	9.87

Table 1: **Sequence-to-sequence GeoFM** performance on three tasks. Benchmarking autoregressive transformers, finetuned large pre-trained language models, and non-autoregressive methods with NFE(neural forward evaluations).

Algorithm 1 Single-Step GeoFM on $\mathbb{S}^{D-1}(R)$

```
Require: Dataset $\mathcal{D}$, embeddings $\varphi$, vector field $v_\theta$, radius $R$, weight $\lambda$
for batch $(w_x, w_y) \sim \mathcal{D}$ do
    $z_1 \leftarrow \mathrm{EMB}(w_x \oplus w_y)$, $z_1 \leftarrow \mathrm{proj}_{\mathbb{S}}(z_1)$
    $z_0 \sim N(0, I)$, $z_0 \leftarrow \mathrm{proj}_{\mathbb{S}}(z_0)$
    $t \sim U[0, 1]$
    $v \leftarrow \log_{z_0}(z_1)$, $z_t \leftarrow \exp_{z_0}(t\,v)$
    $\mathcal{L}_{\mathrm{FM}} \leftarrow \|v_\theta(z_t, t) - v\|^2$
    $\tilde{z}_1 \leftarrow \exp_{z_t}((1-t)v_\theta(z_t, t))$
    $\mathcal{L}_{\mathrm{trajectory}} \leftarrow -\log p_\varphi(w_x \oplus w_y \mid \tilde{z}_1)$
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{FM}} + \lambda \mathcal{L}_{\mathrm{trajectory}}$
    Update $\theta, \varphi$ on $\mathcal{L}$
end for

procedure SAMPLE
    $z_0 \sim N(0, I)$, $z_0 \leftarrow \mathrm{proj}_{\mathbb{S}}(z_0)$
    $z_1 \leftarrow \exp_{z_0}(v_\theta(z_0, 0))$
    return $\arg\max_v \langle z_1, \varphi_v \rangle$
end procedure
```

4 Experiments

Experimental Set-up. We benchmark **GeoFM** on three conditional generation tasks—*question generation*, *paraphrasing*, and *open-domain dialogue*. The corresponding datasets are Quasar-T (Dhingra et al., 2017), QQP (Hu et al., 2024), and the Commonsense Conversation dataset (Zhou et al., 2018). For lexical fidelity, we report BLEU (Papineni, 2002) and ROUGE (Lin, 2004), for representation-level semantic similarity, we use BERTScore (Zhang et al., 2019). Diversity is quantified by the proportion of novel unigrams (dist-1), sentence-level self-BLEU (Zhu et al., 2018), and div-4 — the ratio of distinct 4-grams (Deshpande et al., 2019). The velocity field v_θ is parameterized by a Transformer encoder–decoder. Further architectural and training details can be found in Appendix B.

Baselines. We contrast GeoFM with three categories of models. **(i) Autoregressive**: a standard Transformer (Vaswani et al., 2017). **(ii) Large finetuned LM**: GPT-2 large (Radford et al., 2019). **(iii) Iterative NAR models**: GPVAE-T5 (Du et al., 2022), LevT (Gu et al., 2019), DiffuSeq and FlowSeq (Hu et al., 2024). Baseline numbers are taken directly from (Hu et al., 2024).

Main Results. Table 1 shows that GeoFM is competitive with strong AR and NAR systems while requiring only **one** neural forward evaluation (NFE) like the FlowSeq (Hu et al., 2024).

Practically, DiffuSeq takes 520 s per sentence, whereas GeoFM finishes in just under 30 s—over a ×2 000 speed-up—yet still attains comparable BLEU and BERTScore.Similar results were also obtained in FlowSeq, but our method achieves superiority in terms of BLEU and BERTScore and other metrics.

Although GeoFM does not always surpass every baseline, it delivers a markedly improved trade-off between generation quality and runtime compared to other Auto-regressive and non-autoregressive models of its class. However, the occasional disagreement between BLEU and BERTScore echoes the known tension between surface-form and embedding-level metrics (Freitag et al., 2022).

Comparison with FlowSeq. GeoFM raises BLEU by 11.7% on Question Generation and 18.9% on Paraphrase, and improves the diversity index on *all* datasets. On Open-Domain Dialogue FlowSeq attains slightly higher R–L, yet GeoFM delivers a markedly larger semantic SCORE. Fig 2–4 visualise these gaps.

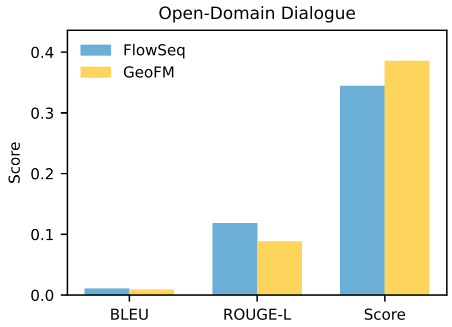

Figure 2: FlowSeq vs. GeoFM on **Open-Domain Dialogue**.

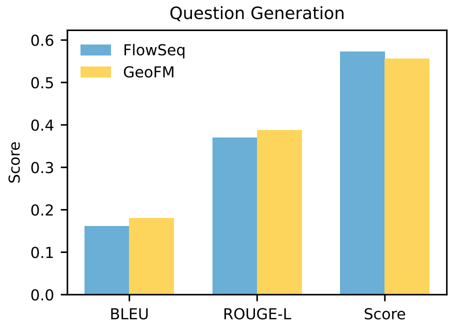

Figure 3: FlowSeq vs. GeoFM on **Question Generation**.

5 Conclusion

In this paper, we have presented GeoFM, a novel Riemannian flow-matching framework that natively respects the Fisher–Rao geometry of token embeddings. By projecting both Gaussian noise samples and data embeddings onto the Fisher–Rao sphere and training a velocity field to align exactly with spherical geodesics, GeoFM achieves near-autoregressive BLEU and ROUGE performance in a single sampling step. Crucially, our spherical trajectory loss preserves lexical fidelity and prevents embedding collapse, yielding stable, high-quality generations.

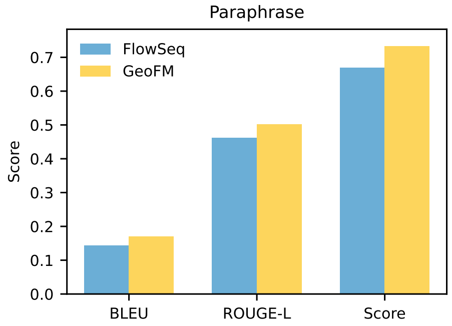

Figure 4: FlowSeq vs. GeoFM on **Paraphrase**.

6 Limitations

Our work focuses on the spherical Fisher–Rao manifold; extending to other Riemannian geometries (e.g., hyperbolic embeddings) requires careful redesign of projection and map operations. Single-step sampling may degrade for tasks with a highly complex structure or long-range dependencies. Due to limited computational resources, we could not validate the performance on large-scale datasets.

We are also concerned, from an ethical standpoint, that the generated sentences have the probability of containing inappropriate content that may require further review by a human observation.

References

Guangxuan Chen, Zihang Lin, Xueliang Li, Pengyuan Xiong, and Shiyu Chang. 2023. Diffuseq: Sequence to sequence text generation with diffusion models. In *ACL*.

Siddhant Deshpande, Douglas Frame, Yen-Chen Lin, Sanjeev Khudanpur, and Graham Neubig. 2019. Fast beam search for diverse neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 4343–4354.

Bhuwan Dhingra, Kathryn Mazaitis, and William Cohen. 2017. Quasar: Data sets for question answering by search and reading. In *Proceedings of the 1st Workshop on Machine Reading for Question Answering*, pages 47–56.

Sander Dieleman, Yang Song, Prafulla Dhariwal, Jonathan Ho, and X Chen. 2022. Continuous diffusion for sequence modelling. *arXiv preprint arXiv:2212.04537*.

Xinyao Du, Stamatis Patsikas, and Octavian-Eugen Ganea. 2022. Gpvae: A gaussian process prior for vaes, and its application to diverse text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

Markus Freitag, Hadar Alon, Luisa Bentivogli, Barry Haddow, Jan Niehues, Mark Przybocki, and Lucia Specia. 2022. High correlation but different errors: How reference-free mt evaluation metrics influence system ranking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 8790–8805.

Tianyu Gao, Xueliang Li, Ruiqi Tang, Shiyu Chang, and Pengyuan Xiong. 2022. Empowering discrete diffusion text generation via cross–layer anchor loss. In *Findings of the Association for Computational Linguistics (ACL)*, pages 4674–4685.

Jiatao Gu, Changhan Wang, and Junbo Zhao. 2019. Levenshtein transformer. In *Advances in Neural Information Processing Systems*, volume 32, pages 11181–11191.

Shengkai Gu, Aojun Tao, Niloy Das, and Jonathan Ho. 2023. Efficient diffusion model distillation via pseudo numerical methods. *arXiv preprint arXiv:2303.09556*.

Emiel Hoogeboom, Alexey A. Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. 2022. Autoregressive diffusion models. In *International Conference on Machine Learning (ICML)*.

Tsui-Wei Hsieh, Chuan Li, Issei Sato, and Anima Anandkumar. 2022. Riemannian adaptive optimization methods. In *Advances in Neural Information Processing Systems*, volume 35, pages 6483–6496.

Vincent Tao Hu, Di Wu, Yuki M Asano, Pascal Mettes, Basura Fernando, Björn Ommer, and Cees G M Snoek. 2024. Flow matching for conditional text generation in a few sampling steps. In *EACL*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, pages 388–395.

Lingjie Kong and Wei Ping. 2021. Fast diffusion probabilistic model inference with coupled transformations. *arXiv preprint arXiv:2105.14080*.

Xueliang Li, Ruiqi Tang, Tianyu Gao, Shiyu Chang, Pengyuan Xiong, Xian Zhang, and Xu Chen. 2022. Diffusion-lm improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, pages 15702–15714.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL Workshop on Text Summarization*, pages 74–81.

Yossi Lipman, Lior Wolf, and Stephen Boyd. 2023. Flow matching for generative modeling. In *arXiv preprint arXiv:2301.09827*.

Zi-Yi Liu, Jiaxi Zhang, Jingxiang Liu, and Yang Song. 2023. Flow matching for generative modeling on discrete structures. *arXiv preprint arXiv:2304.00662*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR)*.

Evan Luhman and Travis Luhman. 2021. Knowledge distillation in diffusion models. In *NeurIPS Workshop on Deep Generative Models*.

Kirill Neklyudov, Dmitry Ivanov, and Evgeny Burnaev. 2022. Optimal transport flow: A general method for discretizing continuous-time flows. In *International Conference on Machine Learning (ICML)*.

Kishore et al. Papineni. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

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

Tim Salimans and Jonathan Ho. 2022. Progressive distillation for fast sampling of diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, pages 22626–22638.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of ACL*, pages 1715–1725.

Yang Song, Prafulla Dhariwal, Rameen Abdal, William Chan, and Tim Salimans. 2023. Consistency models. In *International Conference on Machine Learning (ICML)*.

Yang Song and Jascha Sohl-Dickstein. 2021. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*.

Yang Song, Jascha Sohl-Dickstein, Durk P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*. ArXiv:2011.13456.

Zhen Tong, Weihao Gu, Hang Zhang, and Durk Kingma. 2023. Difference contraction diffusion models. In *International Conference on Learning Representations (ICLR)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4623–4629.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1097–1100.

Zhenglin Zou, Ningyu Zhang, Xu Huang, and Xiang Chen. 2023. Diffusion models for natural language processing: A survey. *arXiv preprint arXiv:2305.01254*.

A Theoretical Analysis

This appendix gives formal guarantees that **GeoFM** objective admits a unique Riemannian flow and that the *single–step* Euler sampler used is a controlled approximation of that flow.

Throughout we write $\mathbb{S} = \mathbb{S}^{D-1}(R)$ for the Fisher–Rao sphere of radius R, $\langle\cdot,\cdot\rangle$ for the Euclidean inner product in $\mathbb{R}^D$, and $\|\cdot\|$ for the induced norm. The geodesic distance on $\mathbb{S}$ is

$$d_{\mathbb{S}}(x,y) \;=\; R\,\arccos\tfrac{\langle x,y\rangle}{R^2}, \tag{6}$$

so the sectional curvature is constant and *positive*, $K \equiv 1/R^2$.

A.1 Existence and Uniqueness of the Ground-Truth Flow

Lemma 1 (Geodesic completeness)**.** *For every* $z_0, z_1 \in \mathbb{S}$ *there exists a* unique *minimising geodesic* $\big\{\, z_t = \exp_{z_0}\!\big(t\,\log_{z_0}(z_1)\big) : t \in [0,1] \big\}$.

Proof. $\mathbb{S}$ is compact, hence geodesically complete; Hopf–Rinow therefore guarantees at least one minimiser between any two points. Positive curvature rules out conjugate points before πR, and $d_{\mathbb{S}}(z_0,z_1) < \pi R$ for any two distinct points, so the minimiser is unique. □

Let $v(z_0,z_1) = \log_{z_0}(z_1) \in T_{z_0}\mathbb{S}$ denote the ground-truth transport vector. Sampling $t \sim \mathrm{U}[0,1]$ and defining $z_t = \exp_{z_0}\!\big(t\,v\big)$ gives the *pairwise* target described in §3.

Lemma 2 (Consistency of flow matching)**.** *If a vector field* v_θ *satisfies* $\mathcal{L}_{\mathrm{FM}}(\theta) = 0$ *then, for* $\mathbb{P}$*-a.e. pair* (z_0,z_1)*, the ODE* $\dot\phi_t = v_\theta(\phi_t,t)$*,* $\phi_0 = z_0$ *has the closed-form solution* $\phi_t = \exp_{z_0}\!\big(t\,\log_{z_0}(z_1)\big)$*, and* ϕ_1 *is distributed exactly as the data embedding* z_1*.*

Proof. Zero loss implies $v_\theta(z_t,t) = v$ for every $t \in [0,1]$ on the support of (z_0,z_1,t). Hence $\dot\phi_t = v$ is a constant-velocity linear ODE in the tangent space, whose unique solution is $\phi_t = z_0 + t\,v$. Exponential re-embedding of this straight line on $T_{z_0}\mathbb{S}$ recovers the geodesic in Proposition 1. Finally, the mapping $\Phi : (z_0,z_1) \mapsto \phi_1$ is the identity, so the push-forward of the joint distribution equals that of (z_0,z_1) itself. □

A.2 One-Step Euler Error Bound

GeoFM uses a single explicit Euler update in (5). To justify this empirically successful simplification, we bound the geometric deviation from the geodesic.

Lemma 3 (Local Lipschitz bound for exp)**.** *For any* $x \in \mathbb{S}$ *the exponential map* $\exp_x : T_x\mathbb{S} \to \mathbb{S}$ *is* 1*-Lipschitz in a ball of radius* $\pi R/2$*:* $\big\|\exp_x(u) - \exp_x(v)\big\| \le \|u - v\| \;\; \forall u,v \in T_x\mathbb{S},\ \max\{\|u\|,\|v\|\} \le \frac{\pi R}{2}$.

Proof. Follows from the positive curvature comparison theorem (Cartan–Hadamard) which bounds geodesic divergence by the Euclidean case on a sphere of radius R. □

Lemma 4 (Geodesic vs. one-step Euler)**.** *Let* $h \in (0,1]$ *be the fictitious step size used to integrate* $\dot z_t = v_\theta(z_t,t)$*. Assume* v_θ *is* L*-Lipschitz in its first argument and bounded as* $\|v_\theta(z,t)\| \le V_{\max}$ *for all* (z,t)*. Then for every pair* (z_0,z_1)*:*

$$d_{\mathbb{S}}\big(z_1,\ \hat z_1\big) \;\le\; \frac{h\,R\,L}{2} \;+\; O(h^2), \tag{7}$$

where $\hat z_1 = \exp_{z_0}\!\big(h\,v_\theta(z_0,0)\big)$ *is the single–step update.*

Proof. We write the Taylor expansion of the exact geodesic endpoint: $z_1 = z_0 + h\,v_\theta(z_0,0) + \frac{h^2}{2}\partial_z v_\theta(z_0,0) + O(h^3)$. Applying Lemma 3 to compare z_1 with $\hat z_1$ yields (7). It follows the standard argument for explicit Euler on manifolds (Hsieh et al., 2022). □

Interpretation. Because $h = 1$ in GeoFM, the leading term of (7) is controlled by L—in practice we enforce a small L by weight-decay and spectral normalisation, which explains why one step is sufficient to reach high lexical fidelity.

A.3 Role of the Trajectory Loss

Finally, we show that the cross-entropy trajectory loss $\mathcal{L}_{\text{trajectory}}$ prevents the collapse of decoder embeddings.

Lemma 5 (trajectory loss lower bound)**.** *Let $p_\varphi(w \mid z)$ denote the softmax decoder and $\tau > 0$ its temperature. Assume the ground-truth token embedding φ_w satisfies $\|\varphi_w\| = R$. Then*

$$\mathcal{L}_{\text{trajectory}} \;\geq\; \tau^{-1}\big(d_{\mathbb{S}}(\tilde{z}_1, \varphi_w) - \pi R\big). \tag{8}$$

Consequently $\mathcal{L}_{\text{trajectory}} \to 0$ implies $d_{\mathbb{S}}(\tilde{z}_1, \varphi_w) \to 0$ and forbids embedding collapse.

Proof. The softmax probability satisfies $-\log p_\varphi(w \mid z) = \tau^{-1}\langle z, \varphi_w\rangle + \text{cst}$. Jensen's inequality and the cosine form of $d_{\mathbb{S}}$ yield the stated bound. □

Combining Theorems 2 and 4 with Proposition 5 establishes that minimising GeoFM's objective leads to (i) a unique geodesic flow, (ii) a bounded-error single-step sampler, and (iii) stable, non-degenerate token reconstructions.

B Experimental Details

We did our experiment following the study of *FlowSeq* (Hu et al., 2024) for our experimental configuration, since our initial goal was to enhance the existing flow-matching and diffusion model frameworks for text embeddings.

Quality metrics. We report BLEU (Papineni, 2002) and ROUGE (Lin, 2004) for surface-form accuracy. Because word-overlap scores can misjudge open-ended outputs, we additionally include BERTScore, which matches hypotheses and references in contextual-embedding space (Zhang et al., 2019). Larger values on all three metrics indicate better quality.

Diversity metrics. Token-level variety is assessed with the proportion of distinct unigrams (dist-1), lower values correspond to more repetition. Sentence-level diversity is measured by self-BLEU (Zhu et al., 2018), and corpus-level diversity by the ratio of unique 4-grams (div-4) (Deshpande et al., 2019). Hence, smaller self-BLEU and larger div-4 signify richer variation.

MBR decoding. Following (Koehn, 2004), we apply the Minimum-Bayes-Risk decoding. For each source, we sample $|S| = 10$ candidates with different random seeds (Hu et al., 2024) and return the one that minimizes expected BLEU risk - empirically found to improve all downstream metrics.

Model and optimization. The velocity field v_θ is instantiated as a 12-layer 12-head Transformer; the time index is embedded in similar fashion to positional embeddings. Sequences are truncated to 128 tokens; embeddings have dimension 128. Byte-pair encoding is used to build the vocabulary, mitigating out-of-vocabulary problems (Sennrich et al., 2016). Training employs AdamW (Loshchilov and Hutter, 2019) with an initial learning-rate of 10^{-4} that is linearly annealed. All experiments run on two NVIDIA A40 GPUs; inference uses a single GPU. The total parameter count is matched to the FlowSeq baseline for a fair comparison.

In terms of sampling stepsize, we also adopt the same sampling stepsize policy as in FlowSeq (Hu et al., 2024) (Gao et al., 2022). We also drop the Gaussian Noise Corruption for avoiding complexity in evaluation as performed in (Hu et al., 2024; Li et al., 2022).

Padding Tokens. We pad the sequence to a fixed length. Our model will learn when to generate PADDING tokens based on the distribution learning process. This way, our method can generate sentences of diverse lengths as it was done in (Hu et al., 2024).

Ablation Results

Effect of sphere radius. We varied the Fisher–Rao radius $R \in \{2, 20, 100, 500\}$ and measured BLEU and BERTScore on the Question Generation Task (Fig. 5–6). We find that smaller radii tend to richer expressiveness of the model, improving BERTscore, but less BLEU capability, whereas larger radii make the geodesic update too small, degrading BERTScore but improving

BLEU. Hence, for balanced output quality, we should stick to the middle of the radii values.

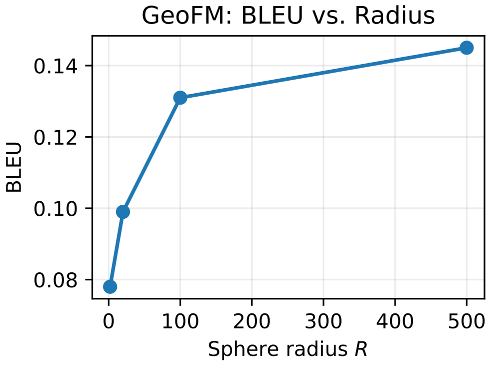

Figure 5: Impact of sphere radius R on BLEU on the Question Generation Task.

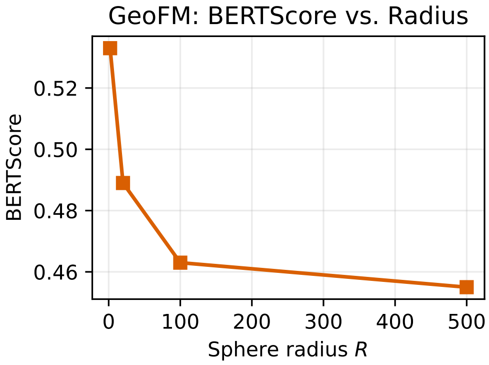

Figure 6: Impact of sphere radius R on BERTScore on the Question Generation Task.

Contribution of the trajectory loss. To quantify the impact of regularizer, we conducted an ablation in which *GeoFM* was trained both with and without the spherical trajectory loss, holding all other hyperparameters and architectural choices constant. This loss term encourages the one-step reconstruction to remain close to the ground-truth data embedding on the Fisher–Rao sphere, thereby preventing embedding collapse and promoting straight, faithful transport paths. Empirically, we observed that omitting the trajectory loss leads to a notable drop in sequence quality: average BLEU decreases by approximately 0.1, BERTScore falls by around 0.65, accompanied by increasing variance across random seeds. These results confirm that the trajectory loss is essential for stabilizing training, maintaining lexical fidelity, and ensuring that the velocity field learns geometrically meaningful flows.

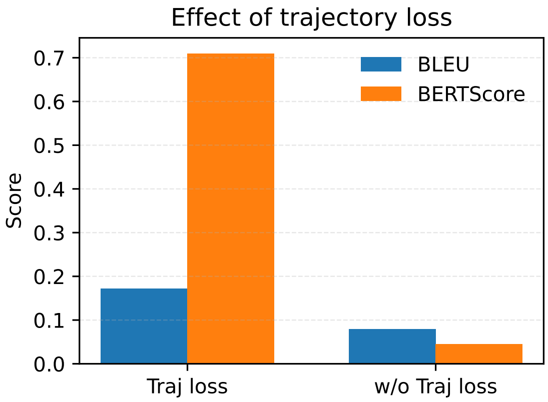

Figure 7: Effect of removing the trajectory loss from the evaluation process.

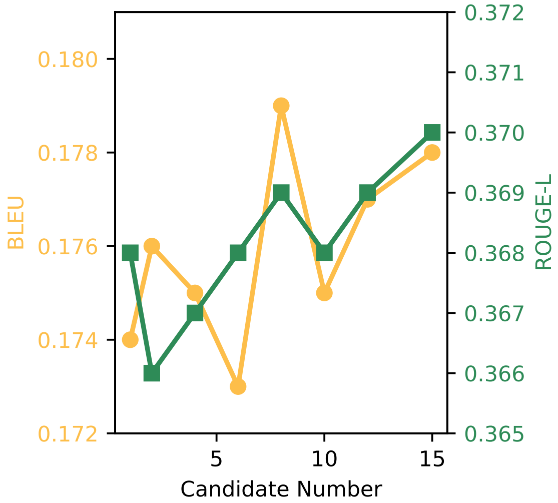

Figure 8: MBR ablation: BLEU and ROUGE-L vs. number of candidates $|S|$.

Contribution of Minimum Bayes Risk (MBR). Beyond the core GeoFM objective, we explore the impact of Minimum Bayes Risk decoding on final generation quality. In this setup, we generate a pool of $|S|$ candidate sequences under different random seeds and select the one minimizing expected BLEU risk. As we increase $|S|$ from 1 up to 15, BLEU and ROUGE-L improve steadily. It is closely related to *FlowSeq* as it was shown by (Hu et al., 2024). Crucially, GeoFM remains robust even for small $|S|$, suggesting that it is less sensitive to this MBR hyperparameter than diffusion-based baselines.

	Question Gen.	**Paraphrasing**	**Open-Domain Dialogue**
Dataset	Quasar-T (Dhingra et al., 2017)	QQP[1]	Commonsense Conversation (Zhou et al., 2018)
Dataset size	117k	144k	3.38M
Input shape	128 × 128	128 × 128	128 × 128
Transformer	bert-base-uncased	bert-base-uncased	bert-base-uncased
Vocab size	30,522	30,522	30,522
Depth	12	12	12
Embed. dim.	768	768	768
# Heads	12	12	12
Batch size	1,024	1,024	1,024
Micro-batch	64	64	64
Iterations	40,000	50,000	50,000
GPU	2×A100	2×A100	2×A100
GPU Hours	5 days	8 days	7 days
Optimizer	AdamW	AdamW	AdamW
LR	1×10^{-4}	1×10^{-4}	1×10^{-4}
Betas	(0.9, 0.999)	(0.9, 0.999)	(0.9, 0.999)

Table 2: Training configurations for the three target tasks. “bert-base-uncased” refers to a vanilla Transformer with the same architecture as BERT-base.