HYPERBOLIC RESIDUAL CONNECTIONS FOR STABILITY AT SCALE

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) have become highly capable at modeling text, largely due to systematic gains from increasing model size and training scale. More recently, LLMs with hyperbolic embedding spaces and operations have shown strong performance by modeling natural language on manifolds that better reflect its latent hierarchical structure. However, hyperbolic models have not been scaled to the same extent as their large pre-trained Euclidean counterparts, in part due to difficulties maintaining training stability as model size increases. In this work, we identify two instabilities of existing hyperbolic residual connections: requiring exceedingly large layer embedding norms to change embedding directions and not being robust to larger layer outputs. We propose a new hyperbolic residual connection, TangentAttnRes, that addresses this instability through two features: (1) aggregation in tangent space, which addresses the first instability by reducing the norm required to make directional changes to the residual vector; and (2) attention-based aggregation, which addresses the second instability by selectively weighting past layer outputs to dampen the impact of unstable layers. We apply our method to multiple hyperbolic LLMs and language Transformers across diverse benchmarks, demonstrating that it improves performance over the existing residual connections. This is especially the case as we scale model parameters, where TangentAttnRes shows continually improved performance while the same scaling causes models with prior methods to show negligible gains or even collapse to near-random performance. Our findings demonstrate that TangentAttnRes is crucial for achieving further performance gains as hyperbolic LLMs scale, presenting a path toward models that can compete at the frontier. [1]

1 INTRODUCTION

Transformer-based large language models (LLMs) improve with scale (Kaplan et al., 2020; Hoffmann et al., 2022), and current models are trained at billions or even trillions of parameters (Brown et al., 2020; Grattafiori et al., 2024). This has been instrumental to their rapid performance gains in recent years and led to the rise of generally capable LLMs (Achiam et al., 2023). Thus far, however, all of these large models have been trained with flat Euclidean embedding spaces, despite the fact that natural language is inherently hierarchical and can be more faithfully embedded in geometries with a more negative curvature (Grover et al., 2026).

This geometric mismatch has led to the creation of hyperbolic LLMs (Chen et al., 2024; He et al., 2025a), which have shown strong performance at smaller scales. Nevertheless, these models have not been scaled to the same extent as their Euclidean counterparts and have yet to achieve comparable general capabilities, partly due to the difficulty of sustaining performance gains as hyperbolic LLMs scale. For example, we observe in Figure 1 that scaling the parameters of the HELM-D hyperbolic LLM (He et al., 2025a) causes training loss to plateau early while loss improves consistently for the variant with fewer parameters. We find similar performance collapses at scale in other popular hyperbolic LLMs (Chen et al., 2024; Madhu et al., 2026) as well.

We trace this collapse at scale to training instability which arises from the hyperbolic residual connection and its behavior as the embedding dimension increases. Across existing hyperbolic residual connections (He et al., 2025d; Katsman et al., 2023; Chami et al., 2019), including those that are

[1] We make our code available at https://anonymous.4open.science/r/stable-hyperbolic-residuals-565C/

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.

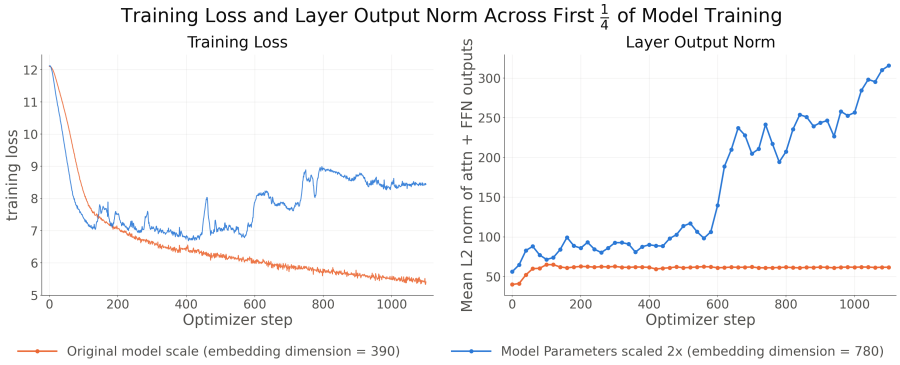


Figure 1: Training cross-entropy loss (left) and mean layer output L2 norm (right) over the first quarter of HELM-D training with the LResNet residual connection at the model's original scale and a 2x scale with embedding dimension, feedforward width, and attention heads doubled. Norms are averaged across attention and feedforward outputs when running inference on the test set of HellaSwag (Zellers et al., 2019). Checkpoints are tested every 20 steps.

implemented in models which collapse at scale, hyperbolic Transformer layer outputs must scale linearly with their layer inputs in order to maintain the same directional change to the residual stream, a stronger requirement than exists for Euclidean space. Existing residual connections are also not robust to unstably large inputs. Specifically, as embedding spaces become higher dimensional, the norms of embeddings grow larger, and this exacerbates both of the struggles with existing methods. We observe in Figure 1 that this leads to rapid norm growth early on in training, and that this growth closely tracks spikes in training loss and an eventual plateau.

To address this shortcoming, we develop a theoretical analysis that identifies the source of norm growth in hyperbolic LLMs as their embedding spaces are scaled and traces it back to the residual connections used in current architectures. We then create TangentAttnRes, a new hyperbolic residual connection that aggregates all previous layer outputs to bound the residual stream norm to that of the largest layer output; this aggregation takes place in the tangent space to avoid the norm growth requirements of residual connections which operate directly on the manifold. These two features allow the method to overcome limitations of existing residual connections that cause rapid norm growth. We test the method's validity by integrating it into a variety of hyperbolic LLMs and find that it consistently outperforms both their original residual connections and other standard alternatives. We additionally implement fused Triton kernels for our method, bringing the runtime of a hyperbolic LLM training step with TangentAttnRes to within 4% of models using much simpler and less expressive baselines. This demonstrates that our approach incurs only modest training overhead, suggesting that the additional computational cost can be manageable for large-scale training.

Our contributions are as follows. We demonstrate multiple theoretical challenges of standard hyperbolic residual connections, and we test their validity empirically on hyperbolic LLMs. We evaluate various hyperbolic LLMs (e.g., He et al., 2025a; Chen et al., 2024; Madhu et al., 2026) with the residual connections in their original architecture as well as other standard hyperbolic residual connections and find that their performance collapses at scale. We then create a new residual connection, TangentAttnRes, which we prove overcomes the theoretical limitations present in previous methods. We find that integrating TangentAttnRes into hyperbolic LLMs improves performance over those that use existing methods, with particularly strong gains as we scale models to sizes where the original architectures suffer from strong performance degradation.

2 Related Work

Hyperbolic Transformers and LLMs. Early hyperbolic networks defined the basic layers, including linear maps and attention, in the Poincaré ball (Ganea et al., 2018; Gulcehre et al., 2019; Shimizu et al., 2020) and the Lorentz model (Nickel & Kiela, 2018; Chen et al., 2022; Mishne et al., 2023; He et al., 2025b). Later works developed hyperbolic Transformers for machine translation (Chen et al., 2022), graphs (Yang et al., 2024c; Cho et al., 2023), and vision (Ermolov et al., 2022; Desai et al., 2023; Fein-Ashley et al., 2024; He et al., 2025c). For language, hyperbolic geometry has been applied to word embeddings (Dhingra et al., 2018; Tifrea et al., 2018; Zhu et al., 2020), in pre-trained

encoders (Chen et al., 2024), in LLM fine-tuning (Yang et al., 2024b; Mandica et al., 2024), and in decoder-only LLMs that operate entirely in hyperbolic space (He et al., 2025a). However, nearly all prior models are trained at a scale far below their Euclidean counterparts. While the numerical limits of hyperbolic representations at large norm are well documented (Mishne et al., 2023; Guo et al., 2022), the question of which architectural component drives embeddings toward those limits as the model grows has not been studied, and it is the question we address.

Hyperbolic Residual Connections. Residual connections (He et al., 2016; Vaswani et al., 2017) keep deep Transformers from collapsing to a low-rank map (Dong et al., 2021). Because early layers can be diluted as later outputs accumulate, Attention Residuals (Team et al., 2026) replace the sum with softmax attention over all preceding layer outputs and report more uniform output magnitudes across depth and performance gains. Hyperbolic adaptations of residuals have been defined through Möbius addition (Ganea et al., 2018; van Spengler et al., 2023; Chami et al., 2019), geodesic steps from the current point (Katsman et al., 2023), addition on the space-like dimension (Bdeir et al., 2024), and a weighted Lorentzian centroid (He et al., 2025d). All of these are sequential and pairwise, and none has been analyzed under increasing embedding dimension. Our work departs from prior works by focusing on how the residual connection determines whether a hyperbolic Transformer can be scaled in parameters, and brings attention over depth to the Lorentz model in a form that also removes the angular cost shown in Proposition 1.

3 PRELIMINARIES

Lorentz model. We work in the Lorentz model of n-dimensional hyperbolic space with constant curvature $K < 0$ (Nickel & Kiela, 2018). A point in this model is written as $x = [x_0, x_s^\top]^\top \in \mathbb{R}^{n+1}$, with time coordinate $x_0 \in \mathbb{R}$ and space-like coordinates $x_s \in \mathbb{R}^n$. The Lorentzian inner product is $\langle x, y\rangle_{\mathcal{L}} = -x_0 y_0 + x_s^\top y_s$, with norm $\|x\|_{\mathcal{L}} = \sqrt{|\langle x, x\rangle_{\mathcal{L}}|}$. The model is a Riemannian manifold (Lee, 2006) which lies on the upper sheet of the hyperboloid $\mathbb{L}^{K,n} = \left\{x \in \mathbb{R}^{n+1} \;\middle|\; \langle x, x\rangle_{\mathcal{L}} = \frac{1}{K},\ x_0 > 0\right\}$, with origin $o = [1/\sqrt{-K}, 0, \ldots, 0]^\top$. The geodesic distance in the model is $d_{\mathbb{H}}(x, y) = \frac{1}{\sqrt{-K}} \cosh^{-1}(K\langle x, y\rangle_{\mathcal{L}})$, and the distance from the origin and Euclidean norm of the space-like coordinates determine one another: $\|x_s\|_2 = \frac{1}{\sqrt{-K}} \sinh\big(\sqrt{-K}\, d_{\mathbb{H}}(o, x)\big)$. Thus, the space-like norm grows exponentially with the distance from the origin.

Tangent space, exponential and logarithmic maps. The tangent space of the manifold at $x \in \mathbb{L}^{K,n}$ is $\mathcal{T}_x\mathbb{L}^{K,n} = \{v \in \mathbb{R}^{n+1} \mid \langle x, v\rangle_{\mathcal{L}} = 0\}$. The exponential map $\exp_x^K : \mathcal{T}_x\mathbb{L}^{K,n} \to \mathbb{L}^{K,n}$ and the logarithmic map $\log_x^K : \mathbb{L}^{K,n} \to \mathcal{T}_x\mathbb{L}^{K,n}$ provide a way to map between the manifold and the tangent space at any point and are defined as

$$\exp_x^K(v) = \cosh\Big(\sqrt{-K}\|v\|_x\Big)\, x + \frac{\sinh\big(\sqrt{-K}\|v\|_x\big)}{\sqrt{-K}\|v\|_x}\, v, \tag{1}$$

$$\log_x^K(y) = \frac{\cosh^{-1}(\beta)}{\sqrt{\beta^2 - 1}}\,(y - \beta x)\,, \qquad \beta = K\langle x, y\rangle_{\mathcal{L}}, \tag{2}$$

with $\|\log_x^K(y)\|_x = d_{\mathbb{H}}(x, y)$. At the origin, tangent vectors have the form $v = [0, v_s^\top]^\top$ with $\|v\|_o = \|v_s\|_2$, so $\mathcal{T}_o\mathbb{L}^{K,n}$ is isometric to $\mathbb{R}^n$. For $x, y \in \mathbb{L}^{K,n} \setminus \{o\}$, the angle at the origin $\angle_{\mathbb{H}}(x, o, y)$ is the angle between $\log_o^K(x)$ and $\log_o^K(y)$ in $\mathcal{T}_o\mathbb{L}^{K,n}$.

Lorentzian centroid. For points $x_1, \ldots, x_m \in \mathbb{L}^{K,n}$ and weights $c_1, \ldots, c_m \geq 0$ that are not all zero, the Lorentzian centroid (Law et al., 2019) is

$$\mathrm{LorentzianCentroid}\big(c_{1:m}, x_{1:m}\big) = \frac{\sum_{i=1}^m c_i x_i}{\sqrt{-K}\,\big\|\sum_{i=1}^m c_i x_i\big\|_{\mathcal{L}}}. \tag{3}$$

We define prior hyperbolic residual connections in Appendix A for comparison with our methods.

4 METHODS

In this section, we analyze the instability of existing methods both theoretically and empirically, demonstrating that existing methods face catastrophic collapse when applied to large-scale models. We then propose TangentAttnRes and demonstrate that our method theoretically and empirically resolves these instabilities. Proofs of all propositions are in Appendix D.

4.1 Instabilities of Prior Hyperbolic Residual Connections

Here we identify two aspects in which existing hyperbolic residual connection methods lead to model collapse as a result of the unique instabilities in hyperbolic geometry. We then propose two techniques targeted at resolving these instabilities.

4.1.1 Instability 1: Uncontrolled Norm Growth

Compared to simple Euclidean addition, existing hyperbolic residual connections suffer from potentially amplified norm growth in their layer outputs in order to modify the residual vector:

Proposition 1. *Let f be a hyperbolic layer, $\oplus$ denote Möbius, space-like addition, or LResNet with equal weights, $x \in \mathbb{L}^{-1,n}$ a layer input with hyperbolic distance to origin r, and $\theta = \angle_{\mathbb{H}}(x, o, x \oplus f(x)) \in (0, \frac{\pi}{2})$. Then, $d_{\mathbb{H}}(o, f(x)) \geq \operatorname{arsinh}(\sinh r \sin\theta) \sim r + \log\sin(\theta)$ as $r \to \infty$. For Euclidean angles and layers, the bound becomes $\|f(x)\|_2 \geq r\sin(\theta) \sim r\theta$ as $\theta \to 0$.*

Proposition 1 shows that, when making small perturbation to the residual vector x using existing hyperbolic residual connections, the layer output $f(x)$ must have a norm that scales linearly with the hyperbolic norm of the residual vector. Furthermore, this issue compounds exponentially as the scaling effect operates at each layer. On the other hand, in the Euclidean case, the required norm is dampened by the small angle. This shows us that, for small θ and large r, ***hyperbolic Transformer layers must output much larger embeddings than their Euclidean counterparts*** in order to have the same angular impact to the residual stream. This is particularly a problem as the embedding dimension scales. In most Transformer architectures, the norm of embeddings naturally scale with their dimension d at a rate of $\sqrt{d}$ (Yang et al., 2024a). As a result, existing hyperbolic residual connections could cause greater instability than their Euclidean counterparts, especially for models at scale, as the compounding large norm required to make small angular changes could lead to norm explosion.

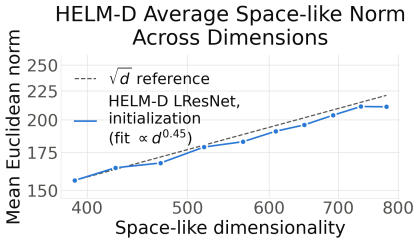


Figure 2: Mean Euclidean norm of the residual stream in HELM-D with the LResNet residual connection at initialization as the space-like dimensionality d increases. Each point is computed on a newly initialized model by running inference on the test set of the HellaSwag dataset (Zellers et al., 2019) and taking the mean L2 norm of the space-like components of the residual stream over all tokens. The fitted exponent is constructed via a least-squares linear regression of $\log$ embedding norm on $\log d$. The intercept for the $\sqrt{d}$ reference line is anchored by the mean Euclidean norm of the model with lowest embedding dimension.

Empirical Validation of Instability 1. To empirically validate the instabilities of existing hyperbolic residual connection methods and the implications of Proposition 1, we train two variants of a dense hyperbolic LLM, HELM-D (He et al., 2025a) at different model sizes using LResNet as the residual connection. As shown in Figure 1, scaling the model's hidden embedding dimensions from 390 to 780 causes training to collapse, with an early loss plateau near the initialized state and norm explosion in the layer outputs. In line with our theoretical hypothesis, Figure 2 additionally shows that the residual stream from scaled models has increasingly large magnitudes, which demonstrates the compounding effect of requiring larger and larger layer outputs. The norm explosion seen in the scaled model is a source of model instability and could be partially responsible for training collapse.

Resolving Instability 1. To address instability 1, we define a tangent space based technique that adapts the tangent space aggregation method from Chami et al. (2019) as residual connection:

$$\text{TangentRes}(x, f(x)) = \exp_o^K\left(\log_o^K(x) + \log_o^K(f(x))\right). \tag{4}$$

TangentRes offers a desirable property over existing hyperbolic residual connection methods: it recovers the same bound as Euclidean addition on the required norm for $f(x)$ to modify x.

Proposition 2. *Let f be a hyperbolic Transformer layer, $\oplus_T$ TangentRes, $x \in \mathbb{L}^{-1,n}$ a layer input with hyperbolic distance to origin r, and $\theta = \angle_{\mathbb{H}}(x, o, x \oplus_T f(x)) \in (0, \frac{\pi}{2})$. Then $d_{\mathbb{H}}(o, f(x)) \geq r\sin(\theta) \sim r\theta$ as $\theta \to 0$.*

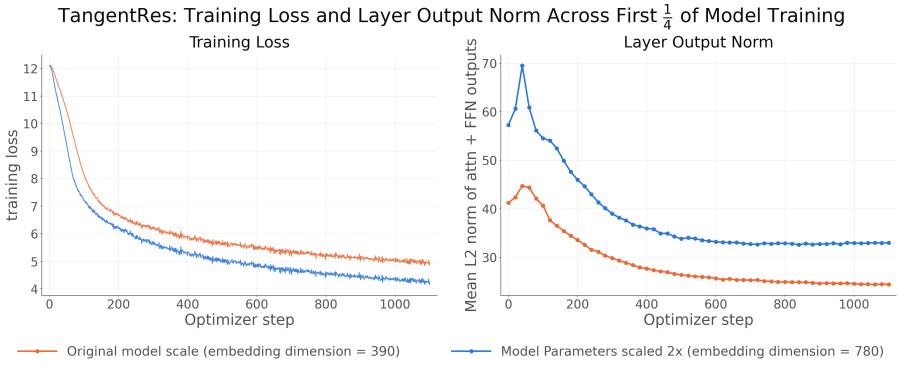


Figure 3: Training cross-entropy loss (left) and mean layer output L2 norm (right) over the first quarter of HELM-D training with the TangentRes residual connection at the model's original scale and a 2x scale with embedding dimension, feedforward width, and attention heads doubled. Norms are averaged across attention and feedforward outputs when running inference on the test set of HellaSwag (Zellers et al., 2019). Checkpoints are tested every 20 steps.

Proposition 2 shows that TangentRes matches the Euclidean bound, which resolves the issue of instability 1. Indeed, in Figure 3, the scaled hyperbolic LLM using TangentRes does not experience the same norm explosion faced by prior residual connections in Figure 1.

4.1.2 Instability 2: Susceptibility to Large Norm Inputs

While TangentRes resolves instability 1, it faces an additional instability together with prior hyperbolic residual connection methods: susceptibility to inputs with exceedingly large ambient norm. In particular, LResNet and space-like addition both assign global weights to their inputs, which does not effectively dampen inputs with extremely large norm that could collapse the model. As hyperbolic operations are more likely to be unstable than their Euclidean counterparts (He et al., 2025b), these global weights could potentially amplify the worst-case instability of hyperbolic operations such as hyperbolic FFNs. Moreover, as a result of the use of exponential maps, Möbius addition and TangentRes suffer from instability that results from large norm inputs even further.

Proposition 3. *Let $x \in \mathbb{L}^{K,n}$ be a residual vector, $f(x)$ be a layer output, and let y be either $y = \textit{TangentRes}(x, f(x))$ or $y = x \oplus_M f(x)$. Then, $\|y\|_2 = O(\|x\|_2 \|f(x)\|_2)$, where $\|\cdot\|_2$ denotes the ambient norm. The Euclidean case is, instead, $\|x + f(x)\|_2 \leq \|x\|_2 + \|f(x)\|_2$.*

Proposition 3 shows that when x and $f(x)$ have large norms, the output of TangentRes and Möbius addition can have a much larger norm than Euclidean addition, as $\|x\|_2\|f(x)\|_2 \gg \|x\|_2 + \|f(x)\|_2$. As a result, existing hyperbolic residual connection methods are more susceptible to instability caused by an input with extremely large norm than Euclidean models.

Resolving Instability 2. To resolve instability 2, we define an attention-based mechanism for hyperbolic residual connection, building on top of attention residuals introduced in Team et al. (2026). Let $h_0, h_1, \ldots, h_l$ be outputs of the first l layers of the hyperbolic network. We define the attention weight of h_i for residual connection at layer l to be $\alpha_{i \to l} = \frac{\exp(s(w_l, h_i))}{\sum_{j=0}^{l} \exp(s(w_l, h_j))}$, where $w_l \in \mathbb{R}^{n+1}$ is a weight vector and $s(\cdot, \cdot)$ is a scoring function. We then consider the attention extension of a hyperbolic residual connection, where a general hyperbolic residual connection $\oplus$ is aggregated across the set of hyperbolic vectors $\{f(\alpha_{i \to l} g(h_i))\}_{i=0}^{l}$. Using this formulation, we may define *LAttnRes*, the attention extension of LResNet, by taking $f = g = Id$ as follows

$$\text{LAttnRes}\big(h_{0:l}\big) = \text{LorentzianCentroid}\big(\alpha_{0 \to l : l \to l}, h_{0:l}\big), \tag{5}$$

where $s(w_l, h_i) = w_l^\top h_i$. We may define the attention extension of space-like addition by taking g as the projection to space-like dimensions and f as the inverse. In the attention extensions, $\alpha_{i \to l}$ acts as dampening controls for inputs with large norms, while aggregating over a list enables the operations to relieve norm pressure by spreading it across multiple vectors. Indeed, the aggregation of layer outputs in LAttnRes bounds the norm of its output below that of the largest past layer.

Proposition 4. *For $h_0, \ldots, h_l \in \mathbb{L}^{-1,n}$, $d_{\mathbb{H}}(o, \text{LAttnRes}(h_{0:l})) \leq \max_{0 \leq i \leq l} d_{\mathbb{H}}(o, h_i)$.*

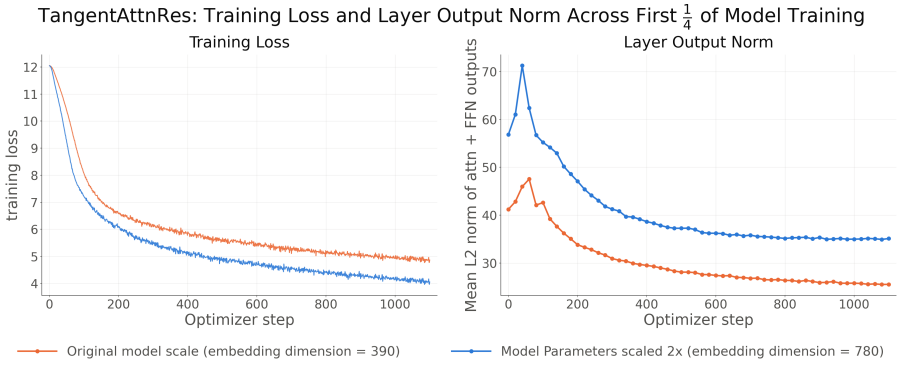


Figure 5: Training cross-entropy loss (left) and mean layer output L2 norm (right) over the first quarter of HELM-D training with the TangentAttnRes residual connection at the model's original scale and a 2x scale with embedding dimension, feedforward width, and attention heads doubled. Norms are averaged across attention and feedforward outputs when running inference on the test set of HellaSwag (Zellers et al., 2019). Checkpoints are tested every 20 steps.

As a result, the attention extension provides a stronger stability guarantee than existing hyperbolic residual connection methods and TangentRes for robustness to unstable layers.

Empirical Validation for Instability 2. In Figure 4, we show that the learned attention weights in LAttnRes apply increasingly more dampening effects as the input norm grows. In contrast and also shown in Figure 4, LResNet assigns global weights to the inputs and thus risk amplifying the input norm instead of dampening it. Thus, LAttnRes is significantly more effective in stabilizing the residual connection against an unstable layer that produces outputs with extremely large norm.

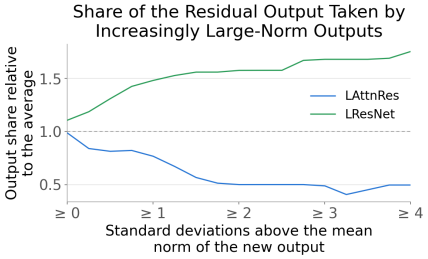


Figure 4: Share of the output vector from the residual connection in 1x HELM-D models taken by high-norm layer outputs. The share of a layer output x is $\|\omega x_s\|/(\|\omega x_s\| + \|p\|)$, where ω is its residual weight and p is the space-like component of the weighted prior residual stream. Tokens are grouped by standard deviations above the mean space-like L2 norm of the new output on HellaSwag (Zellers et al., 2019). The mean share for each standard deviation threshold group is normalized by the mean across all tokens, and the median taken across residual connections is plotted.

4.2 TangentAttnRes: Simultaneously Resolving Both Instabilities

While TangentRes and LAttnRes address instabilities 1 and 2 respectively, neither resolves both instabilities: TangentRes suffers from being highly sensitive to inputs from unstable layers (Proposition 3) while LAttnRes suffers from amplified norm growth when modifying the residual vector (see Appendix D.5). In order to resolve both instabilities simultaneously, we propose TangentAttnRes, an attention extension of TangentRes that retains the desirable theoretical properties of both TangentRes and LAttnRes. Formally, we let $f = \exp_o^K, g = \log_o^K, s(w_l, h_i) = w_l^\top \log_o^K(h_i)$ in the attention extension, where the full operation is given as

$$v_i = \log_o^K(h_i), \quad \widetilde{\alpha}_{i\to l} = \frac{\exp(w_l^\top v_i)}{\sum_{j=0}^{l}\exp(w_l^\top v_j)},$$
$$\mathrm{TangentAttnRes}(h_{0:l}) = \exp_o^K\left(\sum_{i=0}^{l}\widetilde{\alpha}_{i\to l} v_i\right). \tag{6}$$

By utilizing the attention mechanism but computing attention and aggregation in the tangent space, we are able to achieve the theoretical guarantees of both Propositions 2 and 4:

Proposition 5. *Let h_0 be the input embedding, $h_1, \ldots, h_l$ ($l \geq 1$) be the outputs of the first l hyperbolic layers, and $z = \mathrm{TangentAttnRes}(h_{0:l})$. Then, $d_{\mathbb{H}}(o, z) \leq \max_{0\leq i\leq l} d_{\mathbb{H}}(o, h_i)$. Letting $r = d_{\mathbb{H}}(o, h_l)$ and $\theta = \angle_{\mathbb{H}}(h_l, o, z) \in (0, \pi/2)$, it follows $\sum_{i=0}^{l-1}\widetilde{\alpha}_{i\to l}\, d_{\mathbb{H}}(o, h_i) \geq \widetilde{\alpha}_{l\to l}\, r\sin(\theta)$, thus $\max_{0\leq i\leq l-1} d_{\mathbb{H}}(o, h_i) \geq \frac{\widetilde{\alpha}_{l\to l}}{1-\widetilde{\alpha}_{l\to l}} r\sin(\theta)$, where $r\sin(\theta) \sim r\theta$ as $\theta \to 0$.*

Table 1: Token-level perplexity on four held-out corpora (lower is better) for HELM-D with each residual connection in place and for a Euclidean DeepSeek-V3 control with and without attention residuals, at the base configuration and with embedding dimension, feedforward width and attention heads doubled. Perplexity is the exponential of the mean per-token negative log-likelihood. At each model size, the best result for each corpus is in bold and the second best is underlined. †Training failed to converge. Cells marked – are models whose training loss was non-finite.

Model	Geometry	Params	WikiText	LAMBADA	C4	OpenWebText
DeepSeek-V3 1x	Euclidean	124M	254.1	322.4	260.8	204.8
DeepSeek-V3 2x	Euclidean	298M	126.3	223.1	157.2	122.1
DeepSeek-V3-AttnRes 1x	Euclidean	124M	239.2	<u>280.6</u>	230.9	181.4
DeepSeek-V3-AttnRes 2x	Euclidean	298M	114.0	202.7	147.0	114.4
Möbius addition 1x	Hyperbolic	115M	–	–	–	–
Möbius addition 2x	Hyperbolic	259M	–	–	–	–
Space-like addition 1x	Hyperbolic	115M	191.4	324.0	218.7	171.1
Space-like addition 2x†	Hyperbolic	259M	3,932.2	4,240.6	4,062.5	3,636.8
LResNet 1x	Hyperbolic	115M	350.3	365.4	311.0	242.9
LResNet 2x†	Hyperbolic	259M	4,482.1	5,558.2	5,089.8	4,559.8
TangentRes 1x	Hyperbolic	115M	<u>138.3</u>	300.4	<u>183.9</u>	<u>143.9</u>
TangentRes 2x	Hyperbolic	259M	72.4	<u>193.0</u>	114.4	88.0
LAttnRes 1x	Hyperbolic	115M	153.3	331.5	214.0	169.7
LAttnRes 2x	Hyperbolic	259M	83.3	204.6	133.7	103.6
TangentAttnRes 1x	Hyperbolic	115M	**105.7**	**252.5**	**163.5**	**128.9**
TangentAttnRes 2x	Hyperbolic	259M	**59.6**	**164.3**	**104.5**	**79.6**

Proposition 5 shows that TangentAttnRes recovers stability guarantees against both instabilities 1 and 2. We validate this empirically in Figure 5 and Figure 6. The former shows that TangentAttnRes resolves the norm explosion collapse of prior methods and enables stable training. The latter shows that TangentAttnRes successfully downweights extremely large embeddings while TangentRes amplifies the output norm, demonstrating that TangentAttnRes is robust to unstable layer outputs.

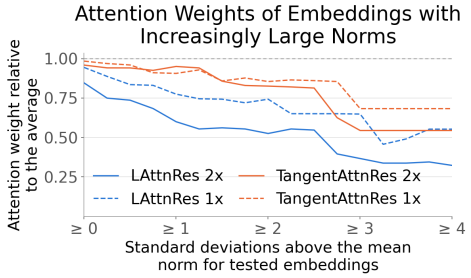


Figure 6: Median normalized attention weights for high-norm outputs from HELM-D models trained with LAttnRes and TangentAttnRes at 1x (dashed) and 2x (solid) scales on HellaSwag (Zellers et al., 2019). Weights are normalized by their mean across tokens for embeddings from the same layer output on the same residual connection in the forward pass. Values below 1 indicate the embeddings are downweighted. Embeddings are grouped by standard deviations above the mean L2 norm of the full hyperbolic vector.

5 Experiments

We demonstrate the efficacy of our methods by substituting them into three different hyperbolic Transformers: HELM-D, a dense hyperbolic LLM from He et al. (2025a), hyperbolic BERT, an encoder-only LLM from Chen et al. (2024), and HyTE-FH from Madhu et al. (2026), a text encoder which we evaluate following its stage 1 masked language modeling (MLM) training.

HELM-D. We train HELM-D on English Wikipedia (Wikimedia Foundation, 2026) with a base configuration of 390 embedding dimensionality, 1,560 feedforward width, 6 attention heads, and an LResNet residual connection to build a 115M parameter model in line with the setup from He et al. (2025a). We then double the number of parameters for each of these model components to build a 259M parameter model and train on this configuration to test the model's ability to scale. At both sizes, we substitute in TangentRes, LAttnRes, and TangentAttnRes and train the model with each residual connection. We retrain the original Euclidean DeepSeek-V3 baseline (DeepSeek-AI, 2024) from He et al. (2025a) and a new configuration DeepSeek-V3-AttnRes, which is identical to DeepSeek-V3 but with attention residuals from Team et al. (2026) implemented as the residual connection, at both scales to establish our methods' performance in comparison to Euclidean models with analogous model sizes and residual connections.

We evaluate each model using held-out perplexity to get a more sensitive measure of LLM performance since all models perform close to a random baseline on standard downstream benchmarks. We evaluate on four held-out corpora: WikiText-103 (Merity et al., 2017) (with all documents held-

Table 2: GLUE results across tasks for hyperbolic BERT with each residual connection, at the base configuration (1x) and with the hidden dimension and feedforward width doubled (2x) and quadrupled (4x). Each cell shows results across three random seeds at the model's best performing learning rate. **Avg** is the mean over the eight tasks. At each model size, the best and second best results for each task are bolded and underlined, respectively. †Training failed to converge.

Model	Params	CoLA	SST-2	MRPC	STS-B	QQP	QNLI	RTE	MNLI	Avg
Space-like 1x	111M	31.8±5.0	87.8±0.1	**86.7±0.7**	85.2±0.2	87.7±0.1	86.8±0.2	63.2±2.4	78.4±0.1	75.94
LAttnRes 1x	111M	34.2±2.2	87.3±0.4	85.5±1.2	84.3±0.1	86.6±1.0	86.7±0.6	64.0±1.7	75.5±1.9	75.53
TangentRes 1x	111M	**37.7±1.1**	**88.1±0.4**	79.9±2.2	**86.6±0.3**	**87.8±0.1**	**87.2±0.3**	64.1±2.2	**78.7±0.0**	76.28
TangentAttnRes 1x	111M	34.5±4.2	87.9±0.6	85.4±0.9	85.4±0.2	87.6±0.3	86.8±0.5	**66.2±0.2**	78.1±0.3	**76.50**
Space-like 2x	395M	21.0±3.2	87.4±0.6	86.7±0.4	84.0±0.4	87.2±0.1	86.6±0.2	64.6±0.8	77.7±0.5	74.40
LAttnRes 2x	395M	32.1±1.3	88.4±0.4	**87.3±0.1**	84.7±0.1	87.9±0.0	**87.7±0.0**	64.3±1.8	78.5±0.1	76.33
TangentRes 2x	395M	33.7±3.0	88.5±0.4	83.6±1.5	**85.9±0.0**	87.9±0.2	86.7±0.5	64.3±1.1	78.4±0.6	76.12
TangentAttnRes 2x	395M	**37.3±0.7**	**88.6±0.1**	86.6±0.7	**85.9±0.3**	**88.2±0.2**	87.4±0.5	**65.0±1.8**	**79.3±0.0**	**77.28**
Space-like 4x†	1.483B	0.0±0.0	50.9±0.0	74.8±0.0	3.6±0.0	31.6±0.0	50.5±0.0	52.7±0.0	35.4±0.0	37.45
LAttnRes 4x	1.483B	35.7±3.0	89.0±0.3	**86.4±0.4**	85.6±0.3	88.2±0.1	**87.7±0.1**	64.4±1.0	79.1±0.0	76.99
TangentRes 4x	1.483B	35.7±1.6	**89.2±0.5**	79.2±4.4	86.3±0.2	88.2±0.0	87.3±0.2	63.7±0.9	78.7±0.2	76.03
TangentAttnRes 4x	1.483B	**35.8±1.3**	89.0±0.4	85.9±0.5	**87.0±0.2**	**88.3±0.1**	87.6±0.1	**66.4±0.0**	**79.3±0.1**	**77.42**

out from the Wikipedia data we train on), the OpenAI variant of LAMBADA (Paperno et al., 2016; Radford et al., 2019), C4 (Raffel et al., 2020), and OpenWebText (Gokaslan et al., 2019) and report the results in Table 1. We observe that models with all prior hyperbolic residual connections have a large drop in performance when scaled while models with any of our proposed residual connections continue to improve in performance as the model is scaled. We find that at both scales TangentAttnRes performs best and has the lowest perplexity out of all configurations on all four corpora for the scaled model. We also observe that the Euclidean models do not have the same issue of performance collapse when scaled and that they continue to improve. Even when given the expressivity of attention residuals, the Euclidean model improves performance slightly but none of these models' performance surpasses that of the hyperbolic models with our methods implemented.

Hyperbolic BERT. We train a 111M parameter hyperbolic BERT model from Chen et al. (2024), configured with a hidden dimension of 768 and feedforward width of 3072 and train on English Wikipedia (Wikimedia Foundation, 2026) [2]. We then double and quadruple the hidden dimension and feedforward width to build 395M and 1.483B parameter models and train these models as well. We train each model with the architecture's original residual connection, a space-like addition followed by a learned scale, as well as with TangentRes, LAttnRes, and TangentAttnRes.

We evaluate the trained models on the GLUE benchmark suite (Wang et al., 2018), a diverse set of tasks which the model is fine-tuned for and then evaluated upon. We perform a fine-tuning sweep over five learning rates and three random seeds, select the learning rate with the best performance for each model, and calculate the model benchmark performance as the average performance over the fine-tuned models trained across the three random seeds. We report the results for each task and an average across all GLUE tasks in Table 2. We find again that the architecture's original residual connection fails to perform at scale, first dropping slightly at 395M parameters followed by a full performance collapse at 1.483B. Meanwhile, our residual connections maintain strong performance at each parameter count and, except for TangentRes, monotonically improve over each of the scalings. This may be an indicator that TangentRes is not quite as stable at scale as LAttnRes and TangentAttnRes or that these methods are able to use their attention weights more effectively at higher scales. We also find that TangentAttnRes is the strongest performing method at every model size, indicating its efficacy in a text encoder model in addition to a decoder-only LLM.

HyTE-FH. Lastly, we use the fully hyperbolic text encoder from Madhu et al. (2026) and run its stage 1 masked language modeling training on the 2023 Wikipedia dump (Wikimedia Foundation, 2026). The base model has an embedding dimension of 756 and feedforward width of 3024 for 157M total parameters. We double these parameters to additionally train at an embedding dimension of 1512 and feedforward width of 6048, reaching 535M total parameters. Based upon the success of TangentAttnRes across the previous two experiments, we compare the original model, which uses LResNet, against one with TangentAttnRes.

[2]Note that we do not train on the BookCorpus dataset from the original repository as it is no longer available to download from its original source.

Table 3: GLUE results for HyTE-FH with LResNet and TangentAttnRes implemented as the residual connection, at the original configuration (1x) and with the hidden dimension and feedforward width doubled (2x). Results are presented identically to Table 2. Best result at each model size is bolded.

Model	Params	CoLA	SST-2	MRPC	STS-B	QQP	QNLI	RTE	MNLI	Avg
LResNet 1x	157M	6.0±2.3	81.1±1.3	78.4±1.0	55.2±2.6	85.2±0.2	87.6±1.0	53.8±1.6	82.2±1.0	66.18
TangentAttnRes 1x	157M	**53.3±0.9**	**92.9±0.3**	**81.4±0.8**	**84.3±0.5**	**89.3±0.1**	**90.8±0.3**	**55.5±2.7**	**84.7±0.3**	**79.04**
LResNet 2x	535M	0.0±0.0	50.3±0.9	55.1±27.8	2.0±0.7	40.7±6.5	50.2±0.5	49.1±2.6	34.2±1.7	35.21
TangentAttnRes 2x	535M	**47.3±7.0**	**93.6±0.4**	**86.5±0.9**	**85.7±0.1**	**90.0±0.0**	**91.9±0.2**	**53.1±3.1**	**86.6±0.1**	**79.33**

We evaluate the models again on GLUE to assess their broad capabilities and find that TangentAttnRes significantly outperforms LResNet at both model sizes. We observe that the performance of the LResNet model drops significantly once scaled while TangentAttnRes sees a performance improvement. The ***persistence pattern*** of TangentAttnRes and baseline methods as parameters are scaled across three hyperbolic LLMs indicates that the success of TangentAttnRes over standard residual connections is not an artifact of any individual training run or model but is instead a general solution to training instability at scale for these models.

Table 4: Runtime of each residual connection using our fused kernels where applicable, both at the operator level and on a full HELM-D $2\times$ training step. Ratio columns are relative to LResNet. Medians are taken over 500 iterations, with the training step measured after 300 warm-up steps.

Residual	Operator (ms)	Ratio	Train Step (ms)	Ratio
TangentRes (fused)	**0.296**	$0.37\times$	**316.24**	$0.984\times$
Space-like addition	0.337	$0.43\times$	316.00	$0.983\times$
LResNet	0.790	$1.00\times$	321.32	$1.000\times$
LAttnRes (fused)	**1.222**	$1.55\times$	**324.84**	$1.011\times$
TangentAttnRes (fused)	**1.243**	$1.57\times$	**333.32**	$1.037\times$
Möbius addition	1.909	$2.42\times$	333.31	$1.037\times$

5.1 Runtime Analysis of Methods

LAttnRes and TangentAttnRes read N prior layer outputs at each residual connection, while the LResNet method only reads two vectors, raising the concern of training time. To address this, we implement fused Triton kernels for LAttnRes, TangentAttnRes, and TangentRes, and compare their cost against standard hyperbolic residual baselines – LResNet, space-like addition, and Möbius addition – at the operator level (the residual connection following the final layer, which has the highest cost) and on a full HELM-D $2\times$ training step. Further methodology details are in Appendix C. Table 4 reports the results. At the operator level, we find our kernels to range from $0.37\times$ to $1.57\times$ the cost of LResNet, and on a full training step, all three of our methods land within 1.1–3.7%. These results highlight that our proposed residual connections have a negligible effect on training time overhead once implemented efficiently and do not demonstrate an issue for efficiently training hyperbolic LLMs at large scales.

6 Conclusion

In this work, we propose TangentAttnRes, a tangent space attention weighted aggregation that can be integrated into hyperbolic LLMs for improved performance and stability at scale. We establish the theoretical limitations of previous methods and demonstrate empirically that models which use them fail to remain stable as their parameters are scaled. We show that TangentAttnRes overcomes these limitations and outperforms standard residual connections when integrated into models both at their original and larger model sizes, with particularly strong gains at increased scales where the original architectures grow unstable and see sharp declines in performance. We envision TangentAttnRes as a key component to scaling hyperbolic LLMs and bringing their performance towards that of today's largest pre-trained language models.

AI use statement

All research ideas presented in this work are original and human-made. LLMs were used for writing code for experiments, table and plot generation, and document formatting, for which code was reviewed by humans. LLMs were used to check for spelling and grammatical errors, and we made all edits for these errors. We used LLMs for drafting the paper and to identify relevant literature.

We reviewed all LLM-assisted text, including suggested references. We used LLMs to aid in our mathematical understanding when formulating mathematical claims or developing proofs, but all written mathematical text is our own and we have verified LLM mathematical claims for correctness. We have not used generative AI tools to generate synthetic datasets, develop theoretical models, design experiments, form hypotheses or clean datasets. Qualitative data analysis and translation are inapplicable to this work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We refrained from using the BookCorpus dataset to train a model in one of our experiments despite it originally being used in the original work due to the removal of the original dataset by the author and the complex ethical history with the dataset. This is noted in a footnote.

REPRODUCIBILITY STATEMENT

We make our code available at https://anonymous.4open.science/r/stable-hyperbolic-residuals-565C/ and use only publicly available data for training and evaluation of models. All datasets used are cited throughout the paper. We train models using publicly available codebases that can be found in the referenced work.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ahmad Bdeir, Kristian Schwethelm, and Niels Landwehr. Fully hyperbolic convolutional neural networks for computer vision. In *ICLR*, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully hyperbolic neural networks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5672–5686, 2022.

Weize Chen, Xu Han, Yankai Lin, Kaichen He, Ruobing Xie, Jie Zhou, and Zhiyuan Liu. Hyperbolic pre-trained language model. *IEEE TASLP*, 32, 2024.

Sungjun Cho, Seunghyuk Cho, Sungwoo Park, Hankook Lee, Honglak Lee, and Moontae Lee. Curve your attention: Mixed-curvature transformers for graph representation learning. *arXiv preprint arXiv:2309.04082*, 2023.

DeepSeek-AI. Deepseek-v3 technical report, 2024. URL https://arxiv.org/abs/2412.19437.

Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *ICML*, pp. 7694–7731. PMLR, 2023.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. Embedding text in hyperbolic spaces. In Goran Glavaš, Swapna Somasundaran, Martin Riedl, and Eduard Hovy (eds.), *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pp. 59–69, New Orleans, Louisiana, USA, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-1708. URL https://aclanthology.org/W18-1708/.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International conference on machine learning*, pp. 2793–2803. PMLR, 2021.

Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *CVPR*, pp. 7409–7419, 2022.

Jacob Fein-Ashley, Ethan Feng, and Minh Pham. Hvt: A comprehensive vision framework for learning in non-euclidean space. *arxiv*, 2024. URL https://arxiv.org/abs/2409.16897.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. In *NeurIPS*, pp. 5345–5355, 2018.

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

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv:2407.21783*, 2024.

Karish Grover, Hanqing Zeng, Yinglong Xia, Christos Faloutsos, and Geoffrey J. Gordon. Text has curvature. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=TLrE49BPC4.

Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, et al. Hyperbolic attention networks. In *ICLR*, 2019.

Yunhui Guo, Xudong Wang, Yubei Chen, and Stella X Yu. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *CVPR*, pp. 11–20, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Neil He, Rishabh Anand, Hiren Madhu, Ali Maatouk, Smita Krishnaswamy, Leandros Tassiulas, Menglin Yang, and Rex Ying. HELM: Hyperbolic large language models via mixture-of-curvature experts. *arXiv preprint arXiv:2505.24722*, 2025a.

Neil He, Jiahong Liu, Buze Zhang, Ngoc Bui, Ali Maatouk, Irwin King, Menglin Yang, Melanie Weber, and Rex Ying. Position: Beyond euclidean – foundation models should embrace non-euclidean geometries. In *The Fourth Learning on Graphs Conference*, 2025b. URL https://openreview.net/forum?id=WoK4o90lln.

Neil He, Menglin Yang, and Rex Ying. Hypercore: The core framework for building hyperbolic foundation models with comprehensive modules. *arXiv preprint arXiv:2504.08912*, 2025c.

Neil He, Menglin Yang, and Rex Ying. Lorentzian residual neural networks. In *KDD*, 2025d.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022. URL https://arxiv.org/abs/2203.15556.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL https://arxiv.org/abs/2001.08361.

Isay Katsman, Eric Chen, Sidhanth Holalkere, Anna Asch, Aaron Lou, Ser Nam Lim, and Christopher M De Sa. Riemannian residual neural networks. *Advances in neural information processing systems*, 36:63502–63514, 2023.

Marc Law, Renjie Liao, Jake Snell, and Richard Zemel. Lorentzian distance learning for hyperbolic representations. In *ICML*, pp. 3672–3681. PMLR, 2019.

John M Lee. *Riemannian manifolds: An introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

Hiren Madhu, Ngoc Bui, Ali Maatouk, Leandros Tassiulas, Smita Krishnaswamy, Menglin Yang, Sukanta Ganguly, Kiran Srinivasan, and Rex Ying. HypRAG: Hyperbolic dense retrieval for retrieval augmented generation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=w6FXg6d6My.

Paolo Mandica, Luca Franco, Konstantinos Kallidromitis, Suzanne Petryk, and Fabio Galasso. Hyperbolic learning withnbsp;multimodal large language models. In *Computer Vision – ECCV 2024 Workshops: Milan, Italy, September 29–October 4, 2024, Proceedings, Part XVII*, pp. 382–398, Berlin, Heidelberg, 2024. Springer-Verlag. ISBN 978-3-031-91584-0. doi: 10.1007/978-3-031-91585-7_23. URL https://doi.org/10.1007/978-3-031-91585-7_23.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Gal Mishne, Zhengchao Wan, Yusu Wang, and Sheng Yang. The numerical stability of hyperbolic representation learning. In *International Conference on Machine Learning*, pp. 24925–24949. PMLR, 2023.

Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *ICML*, pp. 3779–3788, 2018.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc-Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1525–1534, 2016.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. Hyperbolic neural networks++. In *ICLR*, 2020.

Kimi Team, Guangyu Chen, Yu Zhang, Jianlin Su, Weixin Xu, Siyuan Pan, Yaoyu Wang, Yucheng Wang, Guanduo Chen, Bohong Yin, Yutian Chen, Junjie Yan, Ming Wei, Y. Zhang, Fanqing Meng, Chao Hong, Xiaotong Xie, Shaowei Liu, Enzhe Lu, Yunpeng Tai, Yanru Chen, Xin Men, Haiqing Guo, Y. Charles, Haoyu Lu, Lin Sui, Jinguo Zhu, Zaida Zhou, Weiran He, Weixiao Huang, Xinran Xu, Yuzhi Wang, Guokun Lai, Yulun Du, Yuxin Wu, Zhilin Yang, and Xinyu Zhou. Attention residuals, 2026. URL https://arxiv.org/abs/2603.15031.

Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincaré glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*, 2018.

Max van Spengler, Erwin Berkhout, and Pascal Mettes. Poincaré resnet. *CVPR*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pp. 5998–6008, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355, 2018.

Wikimedia Foundation. Wikimedia downloads. `https://dumps.wikimedia.org`, 2026. Accessed: 2026-04-29.

Greg Yang, James B. Simon, and Jeremy Bernstein. A spectral condition for feature learning, 2024a. URL `https://arxiv.org/abs/2310.17813`.

Menglin Yang, Aosong Feng, Bo Xiong, Jihong Liu, Irwin King, and Rex Ying. Hyperbolic fine-tuning for large language models. *ICML LLM Cognition Workshop*, 2024b.

Menglin Yang, Harshit Verma, Delvin Ce Zhang, Jiahong Liu, Irwin King, and Rex Ying. Hypformer: Exploring efficient transformer fully in hyperbolic space. In *KDD*, pp. 3770–3781, 2024c.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Yudong Zhu, Di Zhou, Jinghui Xiao, Xin Jiang, Xiao Chen, and Qun Liu. Hypertext: Endowing fasttext with hyperbolic geometry. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1166–1171, 2020.

A Previous Hyperbolic Residual Connections

Here we define existing methods for residual connections on the Lorentz manifold.

Hyperbolic residual connections. Let f_l denote the l-th layer of a hyperbolic Transformer, where we count each attention and feedforward operation as a separate layer. Let $x^{(l)} \in \mathbb{L}^{K,n}$ denote the residual stream after layer l, let $h_l = f_l(x^{(l-1)})$ denote the output of layer l, and let $h_0 = x^{(0)}$ denote the initial embedding. Existing residual connections are binary operations $\oplus : \mathbb{L}^{K,n} \times \mathbb{L}^{K,n} \to \mathbb{L}^{K,n}$ applied once per layer,

$$x^{(l)} = x^{(l-1)} \oplus f_l\big(x^{(l-1)}\big), \tag{7}$$

so each connection merges the stream accumulated over all previous layers with one new layer output. We consider three choices of $\oplus$, written for a layer input x and output $f(x)$.

Parallel transport. Parallel transport moves a tangent vector $v \in \mathcal{T}_x\mathbb{L}^{K,n}$ along the geodesic from x to y

$$P_{x\to y}(v) = v + \frac{\langle y, v\rangle_{\mathcal{L}}}{-1/K - \langle x, y\rangle_{\mathcal{L}}}\,(x+y) \in \mathcal{T}_y\mathbb{L}^{K,n}. \tag{8}$$

Möbius addition transports the layer output to the tangent space of the input and follows the resulting geodesic (Ganea et al., 2018; Chami et al., 2019)

$$x \oplus_M f(x) = \exp_x^K\Big(P_{o\to x}\big(\log_o^K(f(x))\big)\Big). \tag{9}$$

This is the Lorentz-model form of Möbius addition in the Poincaré ball (Ganea et al., 2018).

Space-like addition adds the space-like coordinates and recomputes the time coordinate so that the result lies on the manifold (Bdeir et al., 2024)

$$x \oplus_S f(x) = \left[\sqrt{\|x_s + f(x)_s\|_2^2 - 1/K},\ \big(x_s + f(x)_s\big)^\top\right]^\top. \tag{10}$$

LResNet (He et al., 2025d) takes a weighted Lorentzian centroid of the input and the output,

$$x \oplus_{\mathcal{L}} f(x) = \frac{\omega_1 x + \omega_2 f(x)}{\sqrt{-K}\,\|\omega_1 x + \omega_2 f(x)\|_{\mathcal{L}}}, \tag{11}$$

with learnable weights $\omega_1, \omega_2 > 0$, which is Equation 3 with $m = 2$.

B Ablation of Learned Attention Weights in LAttnRes and TangentAttnRes

We additionally ablate the learnability of attention weights from LAttnRes and TangentAttnRes to assess whether they are crucial for model performance and stability. We train on HELM-D across both model sizes from the main experiment and report the results in Table A1. We find that learned weights improve performance for both residual connections and are essential for maintaining stability with LAttnRes. This underscores the importance of robustness to large layer outputs which attention weights provide, as shown in Figure 6, in stabilizing models with LAttnRes integrated. The comparatively modest drop in performance for TangentAttnRes supports our hypothesis that the tangent space aggregation does in fact make training more stable than the standard Lorentzian centroid operation.

C Runtime Analysis: Environment and Methodology

Hardware and software. All measurements were performed on an NVIDIA H200 GPU using PyTorch 2.7.0 with CUDA 12.8 and Triton 3.3.0. Both operator-level and end-to-end training measurements use FP32.

Timing methodology. We use CUDA events with 100 warmup iterations and 500 timed iterations, and report median measurements.

Table A1: Token-level perplexity on four held-out corpora (lower is better) for HELM-D with LAttnRes and TangentAttnRes using learned attention weights and uniform attention weights ($\alpha_{i\to l} = 1/(l+1)$), at the base configuration and with embedding dimension, feedforward width and attention heads doubled. †Training failed to converge.

Model	Weights	Params	WikiText	LAMBADA	C4	OpenWebText
LAttnRes 1x	Learned	115M	153.3	331.5	214.0	169.7
LAttnRes 1x†	Uniform	115M	3,956.7	4,210.1	4,132.2	3,657.9
TangentAttnRes 1x	Learned	115M	105.7	252.5	163.5	128.9
TangentAttnRes 1x	Uniform	115M	135.7	298.7	185.2	144.9
LAttnRes 2x	Learned	259M	83.3	204.6	133.7	103.6
LAttnRes 2x†	Uniform	259M	4,473.7	5,718.6	5,294.1	4,811.8
TangentAttnRes 2x	Learned	259M	59.6	164.3	104.5	79.6
TangentAttnRes 2x	Uniform	259M	76.2	192.8	119.8	91.3

Comparison to unfused PyTorch implementations. Without our fused kernels, TangentRes, TangentAttnRes, and LAttnRes are $2.9\times$, $3.2\times$, and $2.6\times$ more runtime intensive than an LResNet operator, respectively. Our kernels reduce these to $0.37\times$, $1.57\times$, and $1.55\times$. We find the parameter-free TangentRes benefits most from fusion, which we believe is because it does not need to read from the N-vector stack required by our attention-based residual implementations.

D PROOFS

Let $K < 0$ and $k = \sqrt{-K}$. Throughout all our proofs, we write $\mathbb{L}^{K,n} = \{x \in \mathbb{R}^{n+1} : \langle x, x\rangle_{\mathcal{L}} = 1/K,\ x_0 > 0\}$, with $\langle x, y\rangle_{\mathcal{L}} = -x_0 y_0 + \sum_{j=1}^{n} x_j y_j$, and split each $x \in \mathbb{L}^{K,n}$ as $x = (x_0, x_s)$ into its time-like component x_0 and its space-like components $x_s \in \mathbb{R}^n$. Additionally, the origin is $o = (1/k, 0, \ldots, 0)$, the Euclidean norm on $\mathbb{R}^n$ is $\|\cdot\|$, and the ambient Euclidean norm on $\mathbb{R}^{n+1}$ is $\|\cdot\|_2$.

We also utilize the following standard facts about the Lorentz model.

(F1) **Distance to the origin.** $d_{\mathbb{H}}(x, y) = \frac{1}{k}\operatorname{arcosh}\big(K\langle x, y\rangle_{\mathcal{L}}\big)$. Since $K\langle o, p\rangle_{\mathcal{L}} = k\,p_0$, a point $p \in \mathbb{L}^{K,n}$ with $d_{\mathbb{H}}(o, p) = \delta$ satisfies $p_0 = \cosh(k\delta)/k$ and $\|p_s\| = \sinh(k\delta)/k$. For $K = -1$, it follows that $p_0 = \cosh\delta$ and $\|p_s\| = \sinh\delta$.

(F2) **Tangent space at the origin.** A tangent vector v at o must satisfy $\langle o, v\rangle_{\mathcal{L}} = 0$, which forces $v_0 = 0$. So $T_o\mathbb{L}^{K,n} = \{(0, w) : w \in \mathbb{R}^n\}$. For two such vectors, their Lorentzian inner product is the dot product of their space-like coordinates: $\langle (0, w), (0, w')\rangle_{\mathcal{L}} = w^\top w'$. We therefore treat $T_o\mathbb{L}^{K,n}$ as $\mathbb{R}^n$.

(F3) **Exponential and logarithmic maps at the origin.** For $v \in T_o\mathbb{L}^{K,n} \setminus \{0\}$ and $p \in \mathbb{L}^{K,n} \setminus \{o\}$,

$$\exp_o^K(v) = \Big(\tfrac{1}{k}\cosh(k\|v\|),\ \tfrac{1}{k}\sinh(k\|v\|)\tfrac{v}{\|v\|}\Big), \qquad \log_o^K(p) = d_{\mathbb{H}}(o, p)\,\frac{p_s}{\|p_s\|},$$

with $\exp_o^K(0) = o$ and $\log_o^K(o) = 0$. These maps are mutually inverse bijections between $T_o\mathbb{L}^{K,n}$ and $\mathbb{L}^{K,n}$, and $\|\log_o^K(p)\| = d_{\mathbb{H}}(o, p)$.

(F4) **Angles at the origin.** $\angle_{\mathbb{H}}(x, o, y)$ is the angle at o between the geodesics from o to x and to y, i.e., the angle between $\log_o^K(x)$ and $\log_o^K(y)$ in $T_o\mathbb{L}^{K,n}$. By (F3), $\log_o^K(x)$ is a positive multiple of x_s, so $\angle_{\mathbb{H}}(x, o, y)$ is the Euclidean angle between x_s and y_s.

(F5) **Möbius addition is a translation.** On $\mathbb{L}^{K,n}$, Möbius addition is $x \oplus_M y = \exp_x^K\big(\mathrm{PT}_{o\to x}^K(\log_o^K(y))\big)$, which coincides with Möbius addition in the Poincaré ball under the standard isometry (Ganea et al., 2018). Since parallel transport is a linear isometry between tangent spaces and $d_{\mathbb{H}}\big(x, \exp_x^K(v)\big) = \|v\|_{\mathcal{L}}$, we have $d_{\mathbb{H}}(x, x \oplus_M y) = d_{\mathbb{H}}(o, y)$.

Finally, we use the following fact from Euclidean geometry.

Lemma D.1. *Let $a, c \in \mathbb{R}^n$ with $a \neq 0$ and $a + c \neq 0$, and let $\varphi \in [0, \pi]$ be the angle between a and $a + c$. Then $\|c\| \geq \|a\|\sin\varphi$. Conversely, if $\varphi < \pi/2$ and e is a unit vector at angle φ from a, then $c = \|a\|\cos\varphi\, e - a$ makes the angle between a and $a + c$ equal to φ and satisfies $\|c\| = \|a\|\sin\varphi$.*

Proof. Let $\hat{e} = (a+c)/\|a+c\|$, so that $\langle a, \hat{e}\rangle = \|a\|\cos\varphi$. Since $a+c$ lies on the line $\mathbb{R}\hat{e}$,

$$\|c\|^2 = \|(a+c)-a\|^2 \geq \min_{t\in\mathbb{R}} \|t\hat{e}-a\|^2 = \|a\|^2 - \langle a,\hat{e}\rangle^2 = \|a\|^2\sin^2\varphi.$$

To prove the converse, notice that $a + c = \|a\|\cos\varphi\, e$ is a positive multiple of e, since $\cos\varphi > 0$ for $\varphi < \pi/2$. So the angle between a and $a+c$ is φ. Since $\langle a, e\rangle = \|a\|\cos\varphi$, we get $\|c\|^2 = \|a\|^2 - 2\|a\|\cos\varphi\,\langle a,e\rangle + \|a\|^2\cos^2\varphi = \|a\|^2\sin^2\varphi$, as desired. □

D.1 Proof of Proposition 1

Proposition 1. *Let f be a hyperbolic layer, $\oplus$ denote Möbius, space-like addition, or LResNet with equal weights, $x \in \mathbb{L}^{-1,n}$ a layer input with hyperbolic distance to origin r, and $\theta = \angle_{\mathbb{H}}(x, o, x \oplus f(x)) \in (0, \frac{\pi}{2})$. Then, $d_{\mathbb{H}}(o, f(x)) \geq \operatorname{arsinh}(\sinh r \sin\theta) \sim r + \log\sin(\theta)$ as $r \to \infty$. For Euclidean angles and layers, the bound becomes $\|f(x)\|_2 \geq r\sin(\theta) \sim r\theta$ as $\theta \to 0$.*

Proof. Here $K = -1$, so by (F1), every $p \in \mathbb{L}^{-1,n}$ satisfies $\|p_s\| = \sinh d_{\mathbb{H}}(o,p)$. Now let $y = f(x)$, $z = x \oplus y$ and $\rho = d_{\mathbb{H}}(o,y)$. Since arsinh is increasing, it suffices to show $\sinh\rho \geq \sinh r\sin\theta$.

Space-like addition. In this case, $z = \big(\sqrt{1+\|x_s+y_s\|^2},\ x_s+y_s\big)$, so $z_s = x_s + y_s$ and, by (F4), θ is the angle between x_s and $x_s + y_s$. Lemma D.1 with $a = x_s$ and $c = y_s$ then gives $\sinh\rho = \|y_s\| \geq \|x_s\|\sin\theta = \sinh r\sin\theta$.

LResNet. In this case, using weights $w_x, w_y > 0$ and $q = w_x x + w_y y$, LResNet gives $z = q/\sqrt{-\langle q,q\rangle_{\mathcal{L}}}$ (He et al., 2025d), meaning z_s is a positive multiple of $w_x x_s + w_y y_s$ and θ is the angle between $w_x x_s$ and $w_x x_s + w_y y_s$. Lemma D.1 with $a = w_x x_s$ and $c = w_y y_s$ then gives $\sinh\rho \geq \frac{w_x}{w_y}\sinh r\sin\theta$, which is our claim when $w_x = w_y$.

Möbius addition. By (F5), $d_{\mathbb{H}}(x,z) = d_{\mathbb{H}}(o,y) = \rho$, so it suffices to provide a lower bound on the distance between x and z. Let $t = d_{\mathbb{H}}(o,z)$, $\hat{x} = x_s/\|x_s\|$, and $e = z_s/\|z_s\|$. By (F1) and (F4), $x = (\cosh r,\ \sinh r\,\hat{x})$, $z = (\cosh t,\ \sinh t\, e)$ and $\langle \hat{x}, e\rangle = \cos\theta$, so

$$\cosh\rho = -\langle x,z\rangle_{\mathcal{L}} = A\cosh t - B\sinh t, \qquad A = \cosh r, \quad B = \sinh r\cos\theta.$$

Then, since $A > |B|$, we can write $A = C\cosh t_0$ and $B = C\sinh t_0$ with $C = \sqrt{A^2 - B^2}$ and $t_0 = \operatorname{artanh}(B/A)$, so that $A\cosh t - B\sinh t = C\cosh(t - t_0) \geq C$. Therefore

$$\cosh^2\rho \geq A^2 - B^2 = \cosh^2 r - \sinh^2 r\cos^2\theta = 1 + \sinh^2 r\sin^2\theta,$$

giving $\sinh\rho \geq \sinh r\sin\theta$.

Asymptotics. As $u \to \infty$, $\operatorname{arsinh}(u) = \log\big(u + \sqrt{u^2+1}\big) = \log(2u) + o(1)$. Setting $u = \sinh r\sin\theta$ and using $2\sinh r = e^r(1 - e^{-2r})$ gives $\operatorname{arsinh}(\sinh r\sin\theta) = r + \log\sin\theta + o(1)$ as $r \to \infty$, as desired.

Euclidean case. Let $x \in \mathbb{R}^n$ with $\|x\|_2 = r$, and let θ be the angle between x and $x + f(x)$. Lemma D.1 with $a = x$ and $c = f(x)$ gives $\|f(x)\|_2 \geq r\sin\theta$. Finally, $r\sin\theta/(r\theta) = \sin\theta/\theta \to 1$ as $\theta \to 0$. □

D.2 Proof of Proposition 2

Proposition 2. *Let f be a hyperbolic Transformer layer, $\oplus_T$ TangentRes, $x \in \mathbb{L}^{-1,n}$ a layer input with hyperbolic distance to origin r, and $\theta = \angle_{\mathbb{H}}(x, o, x \oplus_T f(x)) \in (0, \frac{\pi}{2})$. Then $d_{\mathbb{H}}(o, f(x)) \geq r\sin(\theta) \sim r\theta$ as $\theta \to 0$.*

Proof. Let $u = \log_o^K(x)$ and $v = \log_o^K(f(x))$. Then $\|u\| = r$ and $\|v\| = d_{\mathbb{H}}(o, f(x))$ by (F3). Since $\log_o^K$ inverts $\exp_o^K$, it follows that $\log_o^K\big(x \oplus_T f(x)\big) = u + v$, so by (F4), θ is the angle between u and $u+v$ in $T_o\mathbb{L}^{K,n} \cong \mathbb{R}^n$. Lemma D.1 with $a = u$ and $c = v$ then gives

$$d_{\mathbb{H}}(o, f(x)) = \|v\| \geq \|u\|\sin\theta = r\sin\theta,$$

and $r\sin\theta \sim r\theta$ as $\theta \to 0$ because $\sin\theta/\theta \to 1$. □

D.3 Proof of Proposition 3

Proposition 3. *Let $x \in \mathbb{L}^{K,n}$ be a residual vector, $f(x)$ be a layer output, and let y be either $y = \mathit{TangentRes}(x, f(x))$ or $y = x \oplus_M f(x)$. Then, $\|y\|_2 = O(\|x\|_2 \|f(x)\|_2)$, where $\|\cdot\|_2$ denotes the ambient norm. The Euclidean case is, instead, $\|x + f(x)\|_2 \le \|x\|_2 + \|f(x)\|_2$.*

Proof. Let $a = k\, d_{\mathbb{H}}(o, x)$ and $b = k\, d_{\mathbb{H}}(o, f(x))$. By (F1), every $p \in \mathbb{L}^{K,n}$ satisfies $\|p_s\| < p_0$, so

$$p_0 \le \|p\|_2 \le \sqrt{2}\, p_0, \qquad \text{where } p_0 = \tfrac{1}{k} \cosh\big(k\, d_{\mathbb{H}}(o, p)\big). \tag{12}$$

Specifically, $\cosh a = k\, x_0$ and $\cosh b = k\, f(x)_0$.

Upper bound. For TangentRes, (F3) and the triangle inequality in $T_o\mathbb{L}^{K,n}$ give $d_{\mathbb{H}}(o, y) = \|\log_o^K(x) + \log_o^K(f(x))\| \le d_{\mathbb{H}}(o, x) + d_{\mathbb{H}}(o, f(x))$. For Möbius addition, the triangle inequality in $\mathbb{L}^{K,n}$ and (F5) give $d_{\mathbb{H}}(o, y) \le d_{\mathbb{H}}(o, x) + d_{\mathbb{H}}(x, y) = d_{\mathbb{H}}(o, x) + d_{\mathbb{H}}(o, f(x))$. In both cases, $k\, d_{\mathbb{H}}(o, y) \le a + b$. Since $\cosh$ is increasing on $[0, \infty)$ and $\cosh(a + b) = \cosh a \cosh b + \sinh a \sinh b \le 2 \cosh a \cosh b$,

$$\|y\|_2 \le \sqrt{2}\, y_0 \le \tfrac{\sqrt{2}}{k} \cosh(a + b) \le \tfrac{2\sqrt{2}}{k} \cosh a \cosh b = 2\sqrt{2}\, k\, x_0\, f(x)_0 \le 2\sqrt{2}\, k\, \|x\|_2 \|f(x)\|_2,$$

so $\|y\|_2 = O(\|x\|_2 \|f(x)\|_2)$.

Lower bound. Suppose $\log_o^K(f(x))$ is a nonnegative multiple of $\log_o^K(x)$. For TangentRes, the two tangent vectors point the same way, so $d_{\mathbb{H}}(o, y) = d_{\mathbb{H}}(o, x) + d_{\mathbb{H}}(o, f(x))$. For Möbius addition, $f(x)$ lies on the geodesic ray from o through x. Parallel transport along this geodesic maps its velocity at o to its velocity at x, so $x \oplus_M f(x)$ continues along the same ray, and again $d_{\mathbb{H}}(o, y) = d_{\mathbb{H}}(o, x) + d_{\mathbb{H}}(o, f(x))$. In both cases, $k\, d_{\mathbb{H}}(o, y) = a + b$. Since $\cosh(a + b) \ge \cosh a \cosh b$ for $a, b \ge 0$, equation 12 gives

$$\|y\|_2 \ge y_0 = \tfrac{1}{k} \cosh(a + b) \ge \tfrac{1}{k} \cosh a \cosh b = k\, x_0\, f(x)_0 \ge \tfrac{k}{2} \|x\|_2 \|f(x)\|_2,$$

and $\frac{k}{2} = \frac{\sqrt{-K}}{2}$.

Euclidean case. This follows directly from the triangle inequality. □

D.4 Proof of Proposition 4

Proposition 4. *For $h_0, \ldots, h_l \in \mathbb{L}^{-1,n}$, $d_{\mathbb{H}}(o, \mathrm{LAttnRes}(h_{0:l})) \le \max_{0 \le i \le l} d_{\mathbb{H}}(o, h_i)$.*

Proof. Since all $\alpha_{i \to l}$ come from a softmax, we know $\alpha_{i \to l} > 0$ and $\sum_{i=0}^{l} \alpha_{i \to l} = 1$. Let $q = \sum_{i=0}^{l} \alpha_{i \to l} h_i$. For $K = -1$, the weighted Lorentzian centroid (Law et al., 2019) is $\mu = \mathrm{LAttnRes}(h_{0:l}) = q / \sqrt{-\langle q, q \rangle_{\mathcal{L}}}$. For any $p, p' \in \mathbb{L}^{-1,n}$, we have $-\langle p, p' \rangle_{\mathcal{L}} = \cosh d_{\mathbb{H}}(p, p') \ge 1$, so

$$-\langle q, q \rangle_{\mathcal{L}} = \sum_{i=0}^{l} \sum_{j=0}^{l} \alpha_{i \to l}\, \alpha_{j \to l} \big(-\langle h_i, h_j \rangle_{\mathcal{L}}\big) \ge \Big(\sum_{i=0}^{l} \alpha_{i \to l}\Big)^2 = 1.$$

Thus $\langle q, q \rangle_{\mathcal{L}} < 0$, and $q_0 > 0$ since every $(h_i)_0 > 0$, so μ is well defined and lies in $\mathbb{L}^{-1,n}$. Then by (F1), $\cosh d_{\mathbb{H}}(o, p) = p_0$ for every $p \in \mathbb{L}^{-1,n}$, meaning

$$\cosh d_{\mathbb{H}}(o, \mu) = \mu_0 = \frac{q_0}{\sqrt{-\langle q, q \rangle_{\mathcal{L}}}} \le q_0 = \sum_{i=0}^{l} \alpha_{i \to l} \cosh d_{\mathbb{H}}(o, h_i) \le \max_{0 \le i \le l} \cosh d_{\mathbb{H}}(o, h_i).$$

Finally, since $\cosh$ is increasing on $[0, \infty)$, $d_{\mathbb{H}}(o, \mu) \le \max_{0 \le i \le l} d_{\mathbb{H}}(o, h_i)$, as desired. □

D.5 Proof of Proposition 5

Proposition 5. *Let h_0 be the input embedding, $h_1, \ldots, h_l$ ($l \ge 1$) be the outputs of the first l hyperbolic layers, and $z = \mathrm{TangentAttnRes}(h_{0:l})$. Then, $d_{\mathbb{H}}(o, z) \le \max_{0 \le i \le l} d_{\mathbb{H}}(o, h_i)$. Letting $r = d_{\mathbb{H}}(o, h_l)$ and $\theta = \angle_{\mathbb{H}}(h_l, o, z) \in (0, \pi/2)$, it follows $\sum_{i=0}^{l-1} \widetilde{\alpha}_{i \to l}\, d_{\mathbb{H}}(o, h_i) \ge \widetilde{\alpha}_{l \to l}\, r \sin(\theta)$, thus $\max_{0 \le i \le l-1} d_{\mathbb{H}}(o, h_i) \ge \frac{\widetilde{\alpha}_{l \to l}}{1 - \widetilde{\alpha}_{l \to l}}\, r \sin(\theta)$, where $r \sin(\theta) \sim r\theta$ as $\theta \to 0$.*

Proof. Let $v_i = \log_o^K(h_i)$, so $\|v_i\| = d_{\mathbb{H}}(o, h_i)$ by (F3), and let $m = \sum_{i=0}^{l} \widetilde{\alpha}_{i\to l} v_i$, so $z = \exp_o^K(m)$ and $\log_o^K(z) = m$.

All weights are positive and sum to one, so $d_{\mathbb{H}}(o, z) = \|m\| \le \sum_{i=0}^{l} \widetilde{\alpha}_{i\to l} \|v_i\| \le \max_{0\le i\le l} d_{\mathbb{H}}(o, h_i)$.

Now let $m = u + w$ with $u = \widetilde{\alpha}_{l\to l} v_l$ and $w = \sum_{i=0}^{l-1} \widetilde{\alpha}_{i\to l} v_i$. By (F4), θ is the angle between v_l and m. Since $\widetilde{\alpha}_{l\to l} > 0$, this is also the angle between u and $u + w$. Lemma D.1 with $a = u$ and $c = w$ then gives $\|w\| \ge \|u\| \sin\theta = \widetilde{\alpha}_{l\to l}\, r \sin\theta$. Combined with

$$\|w\| \le \sum_{i=0}^{l-1} \widetilde{\alpha}_{i\to l}\, d_{\mathbb{H}}(o, h_i) \le \big(1 - \widetilde{\alpha}_{l\to l}\big) \max_{0\le i\le l-1} d_{\mathbb{H}}(o, h_i),$$

this proves both inequalities, and $r \sin\theta \sim r\theta$. □

Remark. We note that the angle bound in Proposition 5 does not apply for LAttnRes. Take $K = -1$, let $r = d_{\mathbb{H}}(o, h_l)$, and let $\theta = \angle_{\mathbb{H}}(h_l, o, \mathrm{LAttnRes}(h_{0:l})) \in (0, \pi/2)$. The space-like part of $\mathrm{LAttnRes}(h_{0:l})$ is a positive multiple of $\sum_{i=0}^{l} \alpha_{i\to l}(h_i)_s$, and $\|(h_i)_s\| = \sinh d_{\mathbb{H}}(o, h_i)$ by (F1). Applying Lemma D.1 to the space-like parts (like in the proof above) gives

$$\sinh\Big(\max_{0\le i\le l-1} d_{\mathbb{H}}(o, h_i)\Big) \ge \frac{\alpha_{l\to l}}{1 - \alpha_{l\to l}} \sinh r \sin\theta.$$

This is the bound from Proposition 1, up to the weight factor, meaning LAttnRes still faces the issue of instability 1 while TangentAttnRes does not.