Motivation:
Use more integer computations to avoid float arithmetic. Consider solving the same problem in the Euclidean plane. Construct a constant-error representation by using the integer-lattice square tiling.

Lorentz hyperboloid model is defined as the Riemannian manifold $(\mathbb{L}^d, g_L)$, where $\mathbb{L}^d$ and associated distance function are given as $\mathbb{L}^d = \{ x \in \mathbb{R}^d : x^t x = -1, x_0 > 0 \}$, $d_L(x, y) = \arccosh(-x^t y)$.

Worst case representation error using floating point arithmetic (with error bounded by $\epsilon_{fl}$) is $d_h = \arccosh(1 + \epsilon_{fl}/(2\log(2/d_h) - 1))$, which becomes $d_h = 4\epsilon_{fl} + \log(\epsilon_{fl}) + o(\epsilon_{fl}^2 \exp(-4\epsilon_{fl}))$ if $d_h = O(\log(\epsilon_{fl}))$, where $d_h$ is the hyperbolic distance to origin.

Strategy:
Propose new models in various dimensions based on integer tilings, which represent any point in the space with provably bounded numerical error, further more, distance and gradient computation error is independent of how far the points are from the origin, hence solves the NaN problem.

$L$-tiling model is defined as the Riemannian manifold $(\mathbb{T}_L^n, g_L)$, where $g_L = g_{fl}$ and $\mathbb{T}_L^n = \{ (g, x) \in \mathbb{F} : x^t g x = -1 \}$, $d_L((g, x), (g, y)) = \arccosh(-x^t y^t g(x,y) g y)$.

$G$ is a specially constructed cocompact Fuchsian group that can be represented with integers. Representation error in $L$-tiling model is bounded by $d_{h_L} = \sqrt{3\epsilon_{fl} + 15\epsilon_{fl}/4 + o(\epsilon_{fl})}$, where $\epsilon_{fl}$ is the machine error.

$H$-tiling model is defined as the Riemannian manifold $(\mathbb{T}_H^n, g_H)$, where $\mathbb{T}_H^n = \{ (j, k, x) \in \mathbb{Z} \times (2^{n-1} \times \{0\}) \times \mathbb{S} \}$, $g_H(j, k, x) = \frac{\epsilon_{fl}}{(2^n x_0)^2}$

the associated distance function on $\mathbb{T}_H^n$ is then given as $d_H((j, k, x), (j, k, y)) = \arccosh(1 + \frac{1}{2^n} \frac{\left[2^{2n} z_1 - 2^{2n} z_2 + 2^{n+1} h_1 - 2^{n+1} h_2\right]^2}{2^{2n} z_1^2 + 2^{2n} z_2^2 + 2^{n+1} h_1^2 + 2^{n+1} h_2^2})$.

Representation error in $H$-tiling model is bounded by $d_{h_H} = \sqrt{(n+3)\epsilon_{fl}/2 + (n+3)\epsilon_{fl}/4 + o(\epsilon_{fl})}$, where $\epsilon_{fl}$ is the machine error.

Empirical Results:
1. Compress embeddings: tiling-based models enable us to store the resulting embeddings with fewer bits, offer algorithms to significantly compress hyperbolic embeddings for real-world datasets. For WordNet Nouns, the $\mathbb{L}$-$t$-tiling model can represent the hyperbolic embedding with only (6.07+1.07) MB, which is $2\%$ of the original 372 MB, while it will cost at least 81 MB for any reasonably accurate baseline model.

2. Learn embeddings: tiling-based models enable us to compute more efficiently in integers and learn high-precision embeddings without using BigFloats. Offer algorithms to learn embeddings for real-world datasets. Report some results on learning, which dataset, dimension.

3. We start by evaluating all 2-dimensional embeddings on the Mammals dataset. the $\mathbb{L}$-$t$-tiling model achieves a $8.8\%$ MAP improvement on Mammals compared to Lorentz model.

4. Tiling-based models generally perform better than baseline models in various dimensions, tiling-based models perform particularly better than baseline models for the largest WordNet Nouns dataset, which further validates that numerical issue happens when the embeddings are far from the origin and affects the embedding performances.