APo-VAE: Text Generation in Hyperbolic Space

Shuyang Dai[1*] **Zhe Gan**[2] **Yu Cheng**[2] **Chenyang Tao**[1] **Lawrence Carin**[1] **Jingjing Liu**[2]
[1]Duke University [2]Microsoft Corporation
{shuyang.dai, chenyang.tao, lcarin}@duke.edu
{zhe.gan, yu.cheng, jingjl}@microsoft.com

Abstract

Natural language often exhibits inherent hierarchical structure ingrained with complex syntax and semantics. However, most state-of-the-art deep generative models learn embeddings only in Euclidean vector space, without accounting for this structural property of language. We investigate text generation in a hyperbolic latent space to learn continuous hierarchical representations. An Adversarial Poincaré Variational Autoencoder (APo-VAE) is presented, where both the prior and variational posterior of latent variables are defined over a Poincaré ball via wrapped normal distributions. By adopting the primal-dual formulation of Kullback-Leibler divergence, an adversarial learning procedure is introduced to empower robust model training. Extensive experiments in language modeling, unaligned style transfer, and dialog-response generation demonstrate the effectiveness of the proposed APo-VAE model over VAEs in Euclidean latent space, thanks to its superb capabilities in capturing latent language hierarchies in hyperbolic space.

1 Introduction

The Variational Autoencoder (VAE) (Kingma and Welling, 2013; Rezende et al., 2014) is a generative model widely applied to language-generation tasks, which propagates latent codes drawn from a simple prior to manifest data samples through a decoder. The generative model is augmented by an inference network, which feeds observed data samples through an encoder to yield a distribution on the corresponding latent codes. Since natural language often manifests a latent hierarchical structure, it is desirable for the latent code in a VAE to reflect such inherent language structure, so that the generated text can be more natural and expressive. An example of language structure is illustrated in Figure 1, where sentences are organized into a tree structure.

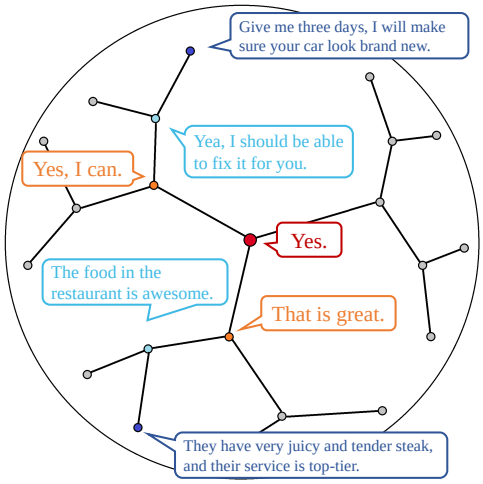

Figure 1: Illustration of the latent hierarchy in natural language. Each tree node is a latent code of its corresponding sentence.

The root node corresponds to simple sentences (*e.g.*, "*Yes*"), while nodes on outer leaves represent sentences with more complex syntactic structure and richer, more specific semantic meaning (*e.g.*, "*The food in the restaurant is awesome*")[1].

In existing VAE-based generative models, such structures are not *explicitly* considered. The latent code often employs a simple Gaussian prior, and the posterior is approximated as a Gaussian with diagonal covariance matrix. Such embeddings assume Euclidean structure, which is inadequate in capturing geometric structure illustrated in Figure 1. While some variants have been proposed to enrich the prior distributions (Xu and Durrett, 2018; Wang et al., 2019; Shi et al., 2019), there is no evidence that structural information in language can be recovered effectively by the model.

Hyperbolic geometry has recently emerged as an effective method for representation learning from data with hierarchical structure (Mathieu et al.,

[*]Work was done when the author interned at Microsoft.

[1]Another possible way to organize sentences is a hierarchy of topics, *e.g.*, a parent node can be a sentence on "*sports*", while its children are sentences on "*basketball*" and "*skiing*".

2019; Nickel and Kiela, 2017). Informally, hyperbolic space can be considered as a continuous map of trees. For example, a Poincaré disk (a hyperbolic space with two dimensions) can represent any tree with arbitrary low distortion (De Sa et al., 2018; Sarkar, 2011). In Euclidean space, however, it is difficult to learn such structural representation even with infinite dimensions (Linial et al., 1995).

Motivated by these observations, we propose Adversarial Poincaré Variational Autoencoder (APo-VAE), a text embedding and generation model based on hyperbolic representations, where the latent code is encouraged to capture the underlying tree-like structure in language. Such latent structure provides more control of the generated sentences, $i.e.$, an increase of sentence complexity and diversity can be achieved along some trajectory from a root to its children. In practice, we define both the prior and the variational posterior of the latent code over a Poincaré ball, via the use of a wrapped normal distribution (Nagano et al., 2019). To obtain more stable model training and learn more flexible representation of the latent code, we exploit the primal-dual formulation of Kullback-Leibler (KL) divergence (Dai et al., 2018) based on the Fenchel duality (Rockafellar et al., 1966), to adversarially optimize the variational bound. Unlike the primal form that relies on Monte Carlo approximation (Mathieu et al., 2019), our dual formulation bypasses the need for tractable posterior likelihoods via the introduction of an auxiliary dual function.

We apply the proposed approach to language modeling, unaligned style transfer and dialog-response generation. For language modeling, in order to enhance the distribution complexity of the prior, we use an additional "variational mixture of posteriors" prior (VampPrior) design (Tomczak and Welling, 2018) for the wrapped normal distribution. Specifically, VampPrior uses a mixture distribution with components from variational posteriors, coupling the parameters of the prior and variational posterior. For unaligned style transfer, we add a sentiment classifier to our model, and disentangle content and sentiment information by using adversarial training (Zhao et al., 2017a). For dialog-response generation, a conditional model variant of APo-VAE is designed to take into account the dialog context.

Experiments also show that the proposed model addresses *posterior collapse* (Bowman et al., 2016), a major obstacle preventing efficient learning of a VAE on text data. In posterior collapse, the encoder learns an approximate posterior similar to the prior, and the decoder tends to ignore the latent code for generation. Experiments show that our proposed model can effectively avoid posterior collapse. We hypothesize that this is due to the use of a more informative prior in hyperbolic space that enhances the complexity of the latent representation, which aligns well with previous work (Tomczak and Welling, 2018; Wang et al., 2019) that advocates a better prior design.

Our main contributions are summarized as follows. (i) We present Adversarial Poincaré Variational Autoencoder (APo-VAE), a novel approach to text embedding and generation based on hyperbolic latent representations. (ii) In addition to the use of a wrapped normal distribution, an adversarial learning procedure and a VampPrior design are incorporated for robust model training. (iii) Experiments on language modeling, unaligned style transfer, and dialog-response generation benchmarks demonstrate the superiority of the proposed approach compared to Euclidean VAEs, as it benefits from capturing informative latent hierarchies in natural language.

2 Preliminaries

2.1 Variational Autoencoder

Let $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ be a dataset of sentences, where each $\boldsymbol{x}_i = [x_{i,1}, ..., x_{i,T_i}]$ is a sequence of tokens of length T_i. Our goal is to learn $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ that best models the observed sentences so that the expected log-likelihood is maximized, *i.e.*, $\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{N}\sum_i \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$.

The variational autoencoder (VAE) considers a latent-variable model $p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})$ to represent sentences, with an auxilary encoder that draws samples of latent code $\boldsymbol{z}$ from the conditional density $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, known as the approximate posterior. Given a latent code $\boldsymbol{z}$, the decoder samples a sentence from the conditional density $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) = \prod_t p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z})$, where the "decoding" pass takes an auto-regressive form. Together with prior $p(\boldsymbol{z})$, the model is given by the joint $p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$. The VAE leverages the approximate posterior to derive an *evidence lower bound* (ELBO) to the (intractable) marginal log-likelihood $\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \log \int p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})\, \mathrm{d}\boldsymbol{z}$:

$$\mathcal{L}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right], \quad (1)$$

where $(\boldsymbol{\theta}, \boldsymbol{\phi})$ are jointly optimized during training, and the gap is given by the decomposition

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \mathcal{L}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) + \mathbb{D}_{\mathrm{KL}}(p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}) \parallel q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})), \tag{2}$$

where $\mathbb{D}_{\mathrm{KL}}$ denotes Kullback-Leibler divergence. Alternatively, the ELBO can be written as:

$$\mathcal{L}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{\boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] - \mathbb{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z})), \tag{3}$$

where the first conditional likelihood and second KL terms respectively characterize reconstruction and generalization capabilities. Intuitively, a good model is expected to strike a balance between good reconstruction and generalization. In most cases, both the prior and variational posterior are assumed to be Gaussian for computational convenience. However, such over-simplified assumptions may not be ideal for capturing the intrinsic characteristics of data that have unique geometrical structure, such as natural language.

2.2 Hyperbolic Space

Riemannian manifolds can provide a more powerful and meaningful embedding space for complex data with highly non-Euclidean structure, that cannot be effectively captured in a vectorial form (*e.g.*, social networks, biology and computer graphics). Of particular interest is the hyperbolic space (Ganea et al., 2018), where (i) the relatively simple geometry allows tractable computations, and (ii) the exponential growth of distance in finite dimensions naturally embeds rich hierarchical structure in a compact form.

Riemannian Geometry. An n-dimensional Riemannian manifold $\mathcal{M}^n$ is a set of points locally similar to a linear space $\mathbb{R}^n$. At each point $\boldsymbol{x}$ of the manifold $\mathcal{M}^n$, we can define a real vector space $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n$ that is tangent to $\boldsymbol{x}$, along with an associated *metric tensor* $g_{\boldsymbol{x}}(\cdot, \cdot) : \mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n \times \mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n \to \mathbb{R}$ which is an inner product on $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n$. Intuitively, a Riemannian manifold behaves like a vector space only in its infinitesimal neighborhood, allowing the generalization of common notation like angle, straight line and distance to a smooth manifold. For each tangent space $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n$, there exists a specific one-to-one map $\exp_{\boldsymbol{x}}(\boldsymbol{v}) : \mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n \to \mathcal{M}^n$ from an ϵ-ball at the origin of $\mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n$ to a neighborhood of $\boldsymbol{x}$ on $\mathcal{M}^n$, called the *exponential map*. We refer to the inverse of an exponential map as the *logarithm map*, denoted $\log_{\boldsymbol{x}}(\boldsymbol{y}) : \mathcal{M}^n \to \mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n$. In addition, a parallel transport $P_{\boldsymbol{x} \to \boldsymbol{x}'} : \mathcal{T}_{\boldsymbol{x}}\mathcal{M}^n \to \mathcal{T}_{\boldsymbol{x}'}\mathcal{M}^n$ intuitively transports tangent vectors along a "straight" line between $\boldsymbol{x}$ and $\boldsymbol{x}'$, so that they remain "parallel." This is the basic machinery that allows us to generalize distributions and computations in the hyperbolic space, as detailed in later sections.

Poincaré Ball Model. Hyperbolic geometry is one type of non-Euclidean geometry with a constant negative curvature. As a classical example of hyperbolic space, an n-dimensional Poincaré ball, with curvature parameter $c \geq 0$ (*i.e.*, radius $\frac{1}{\sqrt{c}}$), can be denoted as $\mathbb{B}_c^n := \{\boldsymbol{z} \in \mathbb{R}^n \,|\, c\|\boldsymbol{z}\|^2 < 1\}$ with its metric tensor given by $g_{\boldsymbol{z}}^c = \lambda_{\boldsymbol{z}}^2 g^E$, where $\lambda_{\boldsymbol{z}} = \frac{2}{1 - c\|\boldsymbol{z}\|^2}$ and g^E denotes the regular Euclidean metric tensor. Intuitively, as $\boldsymbol{z}$ moves closer to the boundary $\frac{1}{\sqrt{c}}$, the hyperbolic distance between $\boldsymbol{z}$ and a nearby $\boldsymbol{z}'$ diverges at a rate of $\frac{1}{1 - c\|\boldsymbol{z}\|^2} \to \infty$. This implies significant representation capacity, as very dissimilar objects can be encoded on a compact domain. Note that as $c \to 0$, the model recovers the Euclidean space $\mathbb{R}^n$, $i.e.$, the lack of hierarchy. In comparison, a larger c implies a stronger hierarchical organization.[2]

Mathematical Operations. We review the closed-form mathematical operations that enable differentiable training for hyperbolic space models, namely the hyperbolic algebra (vector addition) and tangent space computations (exponential/logarithm map and parallel transport). The hyperbolic algebra is formulated under the framework of *gyrovector spaces* (Ungar, 2008), with the addition of two points $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{B}_c^n$ given by the *Möbius addition*:

$$\boldsymbol{z} \oplus_c \boldsymbol{z}' := \frac{(1 + 2c\langle \boldsymbol{z}, \boldsymbol{z}' \rangle + c\|\boldsymbol{z}'\|^2)\boldsymbol{z} + (1 - c\|\boldsymbol{z}\|^2)\boldsymbol{z}'}{1 + 2c\langle \boldsymbol{z}, \boldsymbol{z}' \rangle + c^2\|\boldsymbol{z}\|^2\|\boldsymbol{z}'\|^2}. \tag{4}$$

For any point $\boldsymbol{\mu} \in \mathbb{B}_c^n$, the exponential map and the logarithmic map are given for $\boldsymbol{u} \neq \boldsymbol{0}$ and $\boldsymbol{y} \neq \boldsymbol{\mu}$ by

$$\exp_{\boldsymbol{\mu}}^c(\boldsymbol{u}) := \boldsymbol{\mu} \oplus_c (\tanh(\sqrt{c}\frac{\lambda_{\boldsymbol{\mu}}^c \|\boldsymbol{u}\|}{2})\frac{\boldsymbol{u}}{\sqrt{c}\|\boldsymbol{u}\|}),$$
$$\log_{\boldsymbol{\mu}}^c(\boldsymbol{y}) := \frac{2}{\sqrt{c}\lambda_{\boldsymbol{\mu}}^c} \tanh^{-1}(\sqrt{c}\|\boldsymbol{\kappa}_{\boldsymbol{\mu},\boldsymbol{y}}\|)\frac{\boldsymbol{\kappa}_{\boldsymbol{\mu},\boldsymbol{y}}}{\|\boldsymbol{\kappa}_{\boldsymbol{\mu},\boldsymbol{y}}\|}, \tag{5}$$

[2]The fact that APo-VAE outperforms standard VAE evidences the existence of the hierarchical organization in NLP data.

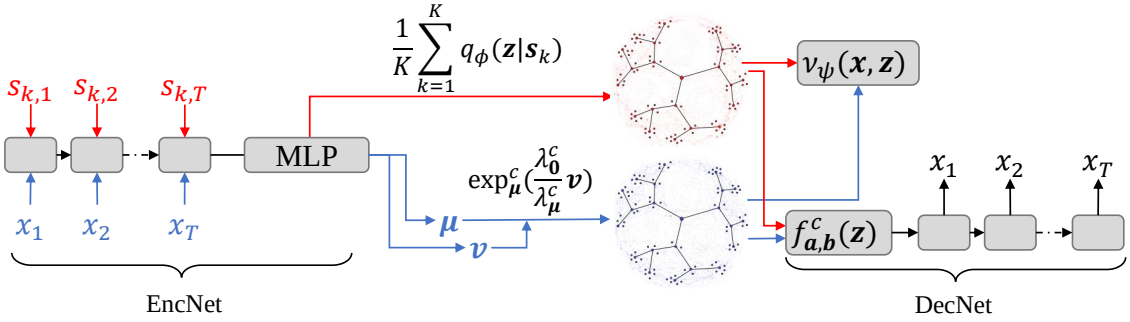

Figure 2: Model framework of the proposed APo-VAE (red is the prior and blue is the posterior). $\boldsymbol{x} = [x_1, ..., x_T]$ is text sequential data, and $\boldsymbol{s}_k = [s_{k,1}, ..., s_{k,T}]$ is the pseudo-input. The posterior (blue) is obtained by (7), and VampPrior (red) is achieved by (12). $\nu_\psi(\boldsymbol{x}, \boldsymbol{z})$ is the dual function.

where $\boldsymbol{\kappa}_{\boldsymbol{\mu},\boldsymbol{y}} := (-\boldsymbol{\mu}) \oplus_c \boldsymbol{y}$. Note that the Poincaré ball model is *geodesically complete* in the sense that $\exp_{\boldsymbol{\mu}}^c$ is well-defined on the full tangent space $\mathcal{T}_{\boldsymbol{\mu}}\mathbb{B}_c^n$. The parallel transport map from a vector $\boldsymbol{v} \in \mathcal{T}_{\boldsymbol{0}}\mathbb{B}_c^n$ to another tangent space $\mathcal{T}_{\boldsymbol{\mu}}\mathbb{B}_c^n$ is given by

$$P_{\boldsymbol{0}\to\boldsymbol{\mu}}^c(\boldsymbol{v}) = \log_{\boldsymbol{\mu}}^c(\boldsymbol{\mu} \oplus_c \exp_{\boldsymbol{0}}^c(\boldsymbol{v})) = \frac{\lambda_{\boldsymbol{0}}^c}{\lambda_{\boldsymbol{\mu}}^c}\boldsymbol{v}. \quad (6)$$

3 Adversarial Poincaré VAE

We first introduce our hyperbolic encoder and decoder, and how to apply reparametrization. We then provide detailed descriptions on model implementation, explaining how the primal-dual form of KL divergence can help stabilize training. Finally, we describe how to adopt VampPrior (Tomczak and Welling, 2018) to enhance performance. A summary of our model scheme is provided in Figure 2.

3.1 Flexible Wrapped Distribution Encoder

We begin by generalizing the standard normal distribution to a Poincaré ball (Ganea et al., 2018). While there are a few competing definitions of the hyperbolic normal, we choose the wrapped normal as our prior and variational posterior, largely due to its flexibility for more expressive generalization. A wrapped normal distribution $\mathcal{N}_{\mathbb{B}_c^n}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is defined as follows: (i) sample vector $\boldsymbol{v}$ from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, (ii) parallel transport $\boldsymbol{v}$ to $\boldsymbol{u} := P_{\boldsymbol{0}\to\boldsymbol{\mu}}^c(\boldsymbol{v})$, and (iii) using exponential map to project $\boldsymbol{u}$ back to $\boldsymbol{z} := \exp_{\boldsymbol{\mu}}^c(\boldsymbol{u})$. Putting these together, a latent sample has the following reparametrizable form:

$$\boldsymbol{z} = \exp_{\boldsymbol{\mu}}^c\left(\frac{\lambda_{\boldsymbol{0}}^c}{\lambda_{\boldsymbol{\mu}}^c}\boldsymbol{v}\right), \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}). \quad (7)$$

For approximate posteriors, $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ depends on $\boldsymbol{x}$. We further generalize the (restrictive) hyperbolic wrapped normal by acknowledging that under the implicit VAE (Fang et al., 2019) framework, one does not need the approximate posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ to be analytically tractable. This allows us to replace the tangent space sampling step $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ in (7) with a more flexible implicit distribution from which we draw samples as $\boldsymbol{v} := G(\boldsymbol{x}, \boldsymbol{\xi}; \boldsymbol{\phi}_1)$ for $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Note that now $\boldsymbol{\mu} := F(\boldsymbol{x}; \boldsymbol{\phi}_2)$ can be regarded as a deterministic displacement vector that anchors embeddings to the correct semantic neighborhood, allowing the stochastic $\boldsymbol{v}$ to only focus on modeling the local uncertainty of the semantic embedding. The synergy between the deterministic and stochastic parts enables efficient representation learning relative to existing alternatives. For simplicity, we denote the encoder neural network as $\text{EncNet}_{\boldsymbol{\phi}}$, which contains G and F, with parameters $\boldsymbol{\phi} = \{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2\}$.

3.2 Poincaré Decoder

To build a geometry-aware decoder for a hyperbolic latent code, we follow Ganea et al. (2018), and use a generalized linear function analogously defined in the hyperbolic space. A Euclidean linear function takes the form $f(\boldsymbol{z}) = \langle \boldsymbol{a}, \boldsymbol{z} - \boldsymbol{b} \rangle = \text{sign}(\langle \boldsymbol{a}, \boldsymbol{z} - \boldsymbol{b} \rangle) \|\boldsymbol{a}\| d^E(\boldsymbol{z}, H_{\boldsymbol{a},\boldsymbol{b}})$, where $\boldsymbol{a}$ is the coefficient, $\boldsymbol{b}$ is the intercept, $H_{\boldsymbol{a},\boldsymbol{b}}$ is a hyperplane passing through $\boldsymbol{b}$ with $\boldsymbol{a}$ as the normal direction, and $d^E(\boldsymbol{z}, H)$ is the distance between $\boldsymbol{z}$ and hyperplane H. The counterpart in Poincaré ball analogously writes

$$f_{\boldsymbol{a},\boldsymbol{b}}^c(\boldsymbol{z}) = \text{sign}(\langle \boldsymbol{a}, \log_{\boldsymbol{b}}^c(\boldsymbol{z}) \rangle_{\boldsymbol{b}}) \|\boldsymbol{a}\|_{\boldsymbol{b}} \, d_c^{\mathbb{B}}(\boldsymbol{z}, H_{\boldsymbol{a},\boldsymbol{b}}^c), \quad (8)$$

where $H_{\boldsymbol{a},\boldsymbol{b}}^c = \{\boldsymbol{z} \in \mathbb{B}_c^n | \langle \boldsymbol{a}, \log_{\boldsymbol{b}}^c(\boldsymbol{z}) \rangle_{\boldsymbol{b}} = 0\}$, and $d_c^{\mathbb{B}}(\boldsymbol{z}, H_{\boldsymbol{a},\boldsymbol{b}}^c) = \frac{1}{\sqrt{c}} \sinh^{-1}\left(\frac{2\sqrt{c}|\langle \boldsymbol{\kappa}_{\boldsymbol{b},\boldsymbol{z}}, \boldsymbol{a} \rangle|}{(1-c\|\boldsymbol{\kappa}_{\boldsymbol{b},\boldsymbol{z}}\|^2)\|\boldsymbol{a}\|}\right)$ are the the gyroplane and the distance between $\boldsymbol{z}$ and the gyroplane, respectively. Specifically, we use the hyperbolic linear function in (8) to extract fea-

tures from the Poincaré embedding $\boldsymbol{z}$. The feature $f^c_{\boldsymbol{a},\boldsymbol{b}}(\boldsymbol{z})$ will be the input to the RNN decoder. We denote the combined network of $f^c_{\boldsymbol{a},\boldsymbol{b}}$ and the RNN decoder as $\text{DecNet}_{\boldsymbol{\theta}}$, where parameters $\boldsymbol{\theta}$ contain $\boldsymbol{a}$ and $\boldsymbol{b}$.

3.3 Implementing APo-VAE

While it is straightforward to compute the ELBO (3) via Monte Carlo estimates using the explicit wrapped normal density (Mathieu et al., 2019), we empirically observe that: (i) the normal assumption restricts the expressiveness of the model, and (ii) the wrapped normal likelihood makes the training unstable. Therefore, we appeal to a primal-dual view of VAE training to overcome such difficulties (Rockafellar et al., 1966; Dai et al., 2018; Fang et al., 2019). Specifically, the KL term in (3) can be reformulated as:

$$\mathbb{D}_{\text{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z})) = \max_{\boldsymbol{\psi}} \left\{\mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z}) - \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\exp\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})\right\}, \tag{9}$$

where $\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})$ is the (auxiliary) dual function (*i.e.*, a neural network) with parameters $\boldsymbol{\psi}$. The primal-dual view of the KL term enhances the approximation ability, while also being tractable computationally. Meanwhile, since the density function in the original KL term in (3) is replaced by the dual function $\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})$, we can avoid direct computation with respect to the probability density function of the wrapped normal distribution.

To train our proposed APo-VAE with the primal-dual form of the VAE objective, we follow the training schemes of coupled variational Bayes (CVB) (Dai et al., 2018) and implicit VAE (Fang et al., 2019), which optimize the objective adversarially. Specifically, we update $\boldsymbol{\psi}$ in the dual function $\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})$ to maximize:

$$\mathcal{L}_1 = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{X}}[\,\mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z}) - \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\exp\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})\,], \tag{10}$$

where $\mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{X}}[\cdot]$ denotes the expectation over empirical distribution on observations. Accordingly, parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are updated to maximize:

$$\mathcal{L}_2 = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{X}}\mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\,\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) - \nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})\,]. \tag{11}$$

Note that the term $\mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{X}}\mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\nu_{\boldsymbol{\psi}}(\boldsymbol{x},\boldsymbol{z})$ is maximized in (10) while it is minimized in (11), *i.e.*, adversarial learning. In other words, one can consider the dual function as a discriminative network that distinguishes between the prior $\boldsymbol{z}\sim p(\boldsymbol{z})$ and the variational posterior $\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, both of which are paired with the input data $\boldsymbol{x}\sim\boldsymbol{X}$.

Algorithm 1 Training procedure of APo-VAE.

Input: Data samples $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$, Poincaré curvature c, and number of pseudo-input K.
Initialize $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\boldsymbol{\psi}$, and $\boldsymbol{\delta}$.
for $iter$ from 1 to max_iter **do**
  Sample a mini-batch $\{\boldsymbol{x}_m\}_{m=1}^M$ from $\boldsymbol{X}$ of size M.
  # Sampling in the Hyperbolic Space.
  Obtain $\boldsymbol{\mu}_m$ and $\boldsymbol{v}_m$ from $\text{EncNet}_{\boldsymbol{\phi}}(\boldsymbol{x}_m)$.
  Move $\boldsymbol{v}_m$ to $\boldsymbol{u}_m = P^c_{\boldsymbol{0}\to\boldsymbol{\mu}_m}(\boldsymbol{v}_m)$ by (6).
  Map $\boldsymbol{u}_m$ to $\boldsymbol{z}_m = \exp^c_{\boldsymbol{\mu}_m}(\boldsymbol{u}_m)$ by (5).
  # Update the dual function and the pseudo-input.
  Sample $\tilde{\boldsymbol{z}}_m$ by (12).
  Update $\boldsymbol{\psi}$ and $\boldsymbol{\delta}$ by gradient ascent on (10)
  # Update the encoder and decoder networks.
  Update $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ by gradient ascent on (11).
end for

3.4 Data-driven Prior

While the use of a standard normal prior is a simple choice in Euclidean space, we argue that it induces bias in the hyperbolic setup. This is because natural sentences have specific meaning, and it is unrealistic to have the bulk of mass concentrated in the center (this is for low dimension; for high dimensions, it will concentrate near the surface of a sphere, which may partly explain why cosine similarity works favorably compared with Euclidean distance for NLP applications).

To reduce the induced bias from a pre-fixed prior, we adopt the VampPrior framework (Tomczak and Welling, 2018), which is a mixture of variational posteriors conditioned on learnable pseudo-data points. Specifically, we consider the prior as a learnable distribution given by

$$p_{\boldsymbol{\delta}}(\boldsymbol{z}) = \frac{1}{K}\textstyle\sum_{k=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{s}_k), \tag{12}$$

where $q_{\boldsymbol{\phi}}$ is the learned approximate posterior, and we call the parameter $\boldsymbol{\delta} := \{\boldsymbol{s}_k\}_{k=1}^K$ pseudo inputs. Intuitively, $p_{\boldsymbol{\delta}}(\boldsymbol{z})$ seeks to match the aggregated posterior (Makhzani et al., 2015): $q(\boldsymbol{z}) = \frac{1}{N}\sum_{i=1}^N q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}_i)$ in a cost-efficient manner via parameterizing the pseudo inputs. By replacing the

prior distribution $p(\boldsymbol{z})$ in (10) with $p_{\boldsymbol{\delta}}(\boldsymbol{z})$, we complete the final objective of the proposed APo-VAE. The detailed training procedure is summarized in Algorithm 1.

4 Related Work

VAE for Text Generation. Many VAE models have been proposed for text generation, most of which focus on solving the issue of posterior collapse. The most popular strategy is to alter the training dynamics, keeping the encoder away from bad local optima. For example, variants of KL annealing (Bowman et al., 2016; Zhao et al., 2018; Fu et al., 2019) dynamically adjust the weight on the KL penalty term as training progresses. Lagging VAE (He et al., 2019) aggressively optimizes the encoder before each decoder update, to overcome the imbalanced training issue between the encoder and decoder. Alternative strategies have also been proposed based on competing theories or heuristics. δ-VAE (Razavi et al., 2019) tackles this issue by enforcing a minimum KL divergence between the posterior and the prior. Yang et al. (2017) blames mode-collapse on the auto-regressive design of the decoder and advocates alternative architectures. A semi-amortized inference network is considered by Kim et al. (2018) to bridge the amortization gap between log-likelihood and the ELBO.

Recent work has also shown that posterior collapse can be ameliorated by using more expressive priors and variational posteriors other than Gaussian. Flow-based VAE is considered in Ziegler and Rush (2019) to enhance the flexibility of prior distributions. A topic-guided prior is proposed in Wang et al. (2019) to achieve more controllable text generation. Fang et al. (2019) explores implicit sample-based representations, without requiring an explicit density form for the approximate posterior. Xu and Durrett (2018) considers replacing the Gaussian with the spherical von Mises-Fisher (vMF) distribution. Compared to these prior arts, our model features structured representation in hyperbolic space, which not only captures latent hierarchies but also combats posterior collapse.

Hyperbolic Space Representation Learning. There has been a recent surge of interest in representation learning in hyperbolic space, largely due to its exceptional effectiveness modeling data with underlying graphical structure (Chamberlain et al., 2017), such as relation nets (Nickel and Kiela, 2017). In the context of NLP, hyperbolic geometry has been considered for word embeddings (Tifrea et al., 2018). A popular vehicle for hyperbolic representation learning is the auto-encoder (AE) framework (Grattarola et al., 2019; Ovinnikov, 2019), where the decoders are built to efficiently exploit the hyperbolic geometry (Ganea et al., 2018). Closest to our APo-VAE are the works of hyperbolic VAEs (Mathieu et al., 2019; Nagano et al., 2019), where wrapped normal distributions have been used. Drawing power from the dual form of the KL, the proposed APo-VAE highlights an implicit posterior and data-driven prior, showing improved training stability.

5 Experiments

We evaluate the proposed model on three tasks: (i) language modeling, (ii) unaligned style transfer, and (iii) dialog-response generation, with quantitative results, human evaluation and qualitative analysis.

5.1 Experimental Setup

Datasets. We use three datasets for language modeling: *Penn Treebank* (PTB) (Marcus et al., 1993), *Yahoo* and *Yelp* corpora (Yang et al., 2017). PTB contains one million words of 1989 Wall Street Journal material annotated in Treebank II style, with 42k sentences of varying lengths. Yahoo and Yelp are much larger datasets, each containing 100k sentences with greater average length.

For unaligned style transfer, we use the Yelp restaurant reviews dataset (Shen et al., 2017), which is obtained by pre-processing the Yelp dataset, $i.e.$, sentences are shortened for more feasible sentence level sentiment analysis. Overall, the dataset includes 350k positive and 250k negative reviews (based on user rating).

Following Gu et al. (2019), we use the Switchboard (Godfrey and Holliman, 1997) dataset for dialogue-response generation. The former contains 2.4k two-sided telephone conversations, manually transcribed and aligned. We split the data into training, validation and test sets following the protocol described in Zhao et al. (2017b).

Evaluation Metrics. To benchmark language modeling performance, we report the ELBO and *Perplexity* (PPL) of APo-VAE and baselines. In order to verify our proposed Apo-VAE is more resistant to posterior collapse, we also report the KL-divergence $\mathbb{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z}))$ and *mutual information* (MI) between $\boldsymbol{z}$ and $\boldsymbol{x}$ (He et al., 2019). The

Model	-ELBO	PPL	KL	MI	AU
	PTB				
VAE	102.6	108.26	1.1	0.8	2
β-VAE	104.5	117.92	7.5	3.1	5
SA-VAE	102.6	107.71	1.2	0.7	2
vMF-VAE	95.8	93.70	2.9	3.2	21
$\mathcal{P}$-VAE	91.4	76.13	4.5	2.9	23
iVAE	87.2	53.44	**12.5**	**12.2**	**32**
APo-VAE	87.2	53.32	8.4	4.8	**32**
APo-VAE+VP	**87.0**	**53.02**	8.9	4.5	**32**
	Yahoo				
VAE	328.6	61.21	0.0	0.0	0
β-VAE	328.7	61.29	6.3	2.8	8
SA-VAE	327.2	60.15	5.2	2.9	10
LAG-VAE	326.7	59.77	5.7	2.9	15
vMF-VAE	318.5	53.92	6.3	3.7	23
$\mathcal{P}$-VAE	313.4	50.57	7.2	3.3	27
iVAE	309.1	47.93	**11.4**	**10.7**	**32**
APo-VAE	286.2	47.00	6.9	4.1	**32**
APo-VAE+VP	**285.6**	**46.61**	8.1	4.9	**32**
	Yelp				
VAE	357.9	40.56	0.0	0.0	0
β-VAE	358.2	40.69	4.2	2.0	4
SA-VAE	357.8	40.51	2.8	1.7	8
LAG-VAE	355.9	39.73	3.8	2.4	11
vMF-VAE	356.2	51.03	4.1	3.9	13
$\mathcal{P}$-VAE	355.4	50.64	4.3	4.8	19
iVAE	348.7	36.88	11.6	**11.0**	**32**
APo-VAE	319.7	34.10	12.1	7.5	**32**
APo-VAE+VP	**316.4**	**32.91**	**12.7**	6.2	**32**

Table 1: Results on PTB, Yahoo, and Yelp datasets. A better language model achieves lower negative ELBO and PPL. Higher KL and MI indicate a better utilization of the latent space.

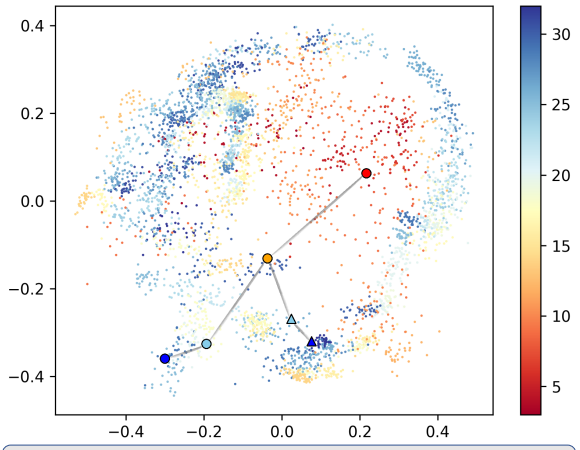

Figure 3: Visualization of the hyperbolic latent space of 5,000 randomly sampled sentences from the test set of PTB. The lengths of the samples are color-coded (red for short ones and blue for longer ones). The six listed sentences are created by modifying data samples.

number of active units (AU) of the latent code is also reported, where the activity of a latent dimension z is measured as $A_{\boldsymbol{z}} = \text{Cov}_{\boldsymbol{x}}\mathbb{E}_{\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[z]$, and defined as active if $A_{\boldsymbol{z}} > 0.01$.

To evaluate our model on unaligned style transfer, we consider the transfer accuracy from one sentiment to another, the BLEU scores between original and transferred sentences, the reconstruction perplexity of original sentences, and the reverse perplexity (RPPL) based on a language model from the transferred sentences.

For dialogue-response generation, we adopt the evaluation metrics used in previous studies (Zhao et al., 2017b; Gu et al., 2019), including BLEU (Papineni et al., 2002), BOW (Liu et al., 2016), and *intra/inter-dist* values (Gu et al., 2019). The first two metrics are used to assess the relevance of the generated response, and the third is for diversity evaluation.

Model Implementation. For language modeling, we adopt the LSTM (Hochreiter and Schmidhuber, 1997) for both the encoder and decoder, with dimension of the latent code set to 32. Following Mathieu et al. (2019), the hyper-parameter c is set to 0.7. For unaligned style transfer, we extend our model in the same fashion as Fang et al. (2019). For dialogue-response generation, we modify APo-VAE following the conditional VAE framework (Zhao et al., 2017b). Specifically, an extra input of context embedding $\boldsymbol{s}$ is supplied to the model (*i.e.*, $p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{s}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{s})$). The prior $p(\boldsymbol{z}|\boldsymbol{s})$ is a wrapped normal conditioned on context embedding, learned together with the posterior.

5.2 Experimental Results

Language Modeling. Table 1 shows results on language modeling. We compare APo-VAE with other VAE-based solutions, including β-VAE (Higgins et al., 2017), SA-VAE (Kim et al., 2018), lagging VAE (LAG-VAE) (He et al., 2019), vMF-VAE (Xu and Durrett, 2018), Poincaré VAE ($\mathcal{P}$-VAE) (Mathieu et al., 2019) and iVAE[3] (Fang et al., 2019). On all three datasets, the proposed model achieves lower negative ELBO and PPL than other

[3]We report iVAE_{MI} results in all our experiments.

Model	BLEU			BOW			Intra-dist		Inter-dist	
	R	P	F1	A	E	G	dist-1	dist-2	dist-1	dist-2
CVAE	0.295	0.258	0.275	0.836	0.572	0.846	0.803	0.415	0.112	0.102
CVAE-BOW	0.298	**0.272**	0.284	0.828	0.555	0.840	0.819	0.493	0.107	0.099
CVAE-CO	0.299	0.269	0.283	0.839	0.557	0.855	0.863	0.581	0.111	0.110
DialogWAE	0.394	0.254	0.309	0.897	0.627	0.887	0.713	0.651	0.245	0.413
iVAE	0.427	0.254	0.319	0.930	0.670	0.900	0.828	0.692	0.391	0.668
APo-VAE	**0.438**	0.261	**0.328**	**0.937**	**0.683**	**0.904**	**0.861**	**0.792**	**0.445**	**0.717**

Table 2: Results on SwitchBoard (P: precision, R: recall, A: average, E: extreme, G: greedy). Higher BLEU and BOW Embedding indicate better quality of generated responses. Higher intra/inter-dist means better generation diversity.

Model	ACC	BLEU	PPL	RPPL
ARAE	**95**	32.5	6.8	395
iVAE	92	36.7	6.2	285
APo-VAE	**95**	**37.8**	**6.1**	**273**

Table 3: Unaligned style transfer on the Yelp restaurant reviews dataset.

	vs iVAE			vs DialogWAE		
	win	loss	tie	win	loss	tie
Informativeness	52.8	27.9	19.3	63.7	27.1	19.2
Coherence	41.7	35.5	22.8	41.2	34.4	24.4
Diversity	51.2	26.4	22.4	62.1	25.1	12.8

Table 4: Human evaluation results. Win/loss/tie indicates the percentage of responses generated by APo-VAE being better/worse/equal to the compared model.

models, demonstrating its strong ability to better model sequential text data. Meanwhile, the larger KL term and higher mutual information (between $\boldsymbol{z}$ and $\boldsymbol{x}$) of APo-VAE model indicate its robustness in handling posterior collapse. In addition, the introduction of a data-driven prior (denoted as APo-VAE+VP) further boosts the performance, especially on negative ELBO and PPL.

Visualization. To verify our hypothesis that the proposed model is capable of learning latent tree structure in text data, we visualize the two-dimensional projection of the learned latent code in Figure 3. For visualization, we randomly draw 5k samples from PTB-test, and encode them to the latent space using the APo-VAE encoder. We color-code each sentence based on its length (*i.e.*, blue for long sentences and red for short sentences). Note that only a small portion of data have a length longer than 32 ($< 10\%$), and human inspection verified that most of them contain multiple sub-sentences. We exclude these samples from our analysis.

As shown in Figure 3, longer sentences (dark blue) tend to occupy the outer rim of the Poincaré ball, while the shorter ones (dark red) are concentrated in the inner area. We also select some long sample sentences (dark blue), and manually shorten them to create several variants of different lengths (ranging from 6 to 27), which are related in a hierarchical manner based on human judgement. We visualize their latent codes projected by the trained APo-VAE. The resulting plot is consistent with a hierarchical structure for APo-VAE: as the sentence becomes more specific, the embedding moves outward. We also decode from the neighbours of these latent codes, the outputs (see the Appendix) of which demonstrate a similar hierarchical structure.

Unaligned Style Transfer. Table 3 shows the results on the Yelp restaurant reviews dataset. APo-VAE achieves over 1 point increased BLEU scores than iVAE, capturing a more informative and structured feature space. Comparable performance is achieved for the other evaluation metrics.

Dialogue Response Generation. Results on SwitchBoard are summarized in Table 2. Our proposed model generates comparable or better responses than the baseline models in terms of both relevance (BLEU and BOW) and diversity (intra/inter-dist). APo-VAE improves the average recall from 0.427 (by iVAE) to 0.438, while significantly enhancing generation diversity (*e.g.*, from 0.692 to 0.792 for intra-dist-2).

Human Evaluation. We further perform human evaluation via Amazon Mechanical Turk. We asked the turkers to compare generated responses from two models, and assess each model's informativeness, relevance to the dialog context (coherence), and diversity. We use 500 randomly sampled contexts from the test set, each assessed by three judges. In order to evaluate diversity, 5 responses are generated for each dialog context. For quality control, only workers with a lifetime task approval rating greater than 98% were allowed to participate in our study. Table 4 summarizes the human evaluation results. The responses generated by our model

are clearly preferred by the judges compared with other competing methods.

6 Conclusions

We present APo-VAE, a novel model for text generation in hyperbolic space. Our model can learn latent hierarchies in natural language via the use of wrapped normals for the prior. A primal-dual view of KL divergence is adopted for robust model training. Extensive experiments on language modeling, text style transfer, and dialog response generation demonstrate the superiority of the model. For future work, we plan to combine APo-VAE with the currently prevailing large-scale pre-trained language models.

References

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *CoNLL*.

Benjamin Paul Chamberlain, James Clough, and Marc Peter Deisenroth. 2017. Neural embeddings of graphs in hyperbolic space. *arXiv preprint arXiv:1705.10359*.

Bo Dai, Hanjun Dai, Niao He, Weiyang Liu, Zhen Liu, Jianshu Chen, Lin Xiao, and Le Song. 2018. Coupled variational bayes via optimization embedding. In *NeurIPS*.

Christopher De Sa, Albert Gu, Christopher Ré, and Frederic Sala. 2018. Representation tradeoffs for hyperbolic embeddings. *Proceedings of machine learning research*.

Le Fang, Chunyuan Li, Jianfeng Gao, Wen Dong, and Changyou Chen. 2019. Implicit deep latent variable models for text generation. In *EMNLP*.

Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. 2019. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. In *NAACL*.

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

John J Godfrey and Edward Holliman. 1997. Switchboard-1 release 2. *Linguistic Data Consortium, Philadelphia*, 926:927.

Daniele Grattarola, Lorenzo Livi, and Cesare Alippi. 2019. Adversarial autoencoders with constant-curvature latent manifolds. *Applied Soft Computing*, 81:105511.

Xiaodong Gu, Kyunghyun Cho, Jung-Woo Ha, and Sunghun Kim. 2019. Dialogwae: Multimodal response generation with conditional wasserstein auto-encoder. In *ICLR*.

Junxian He, Daniel Spokoyny, Graham Neubig, and Taylor Berg-Kirkpatrick. 2019. Lagging inference networks and posterior collapse in variational autoencoders. In *ICLR*.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2(5):6.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*.

Yoon Kim, Sam Wiseman, Andrew C Miller, David Sontag, and Alexander M Rush. 2018. Semi-amortized variational autoencoders. In *ICML*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Nathan Linial, Eran London, and Yuri Rabinovich. 1995. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. 2015. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*.

Emile Mathieu, Charline Le Lan, Chris J Maddison, Ryota Tomioka, and Yee Whye Teh. 2019. Continuous hierarchical representations with poincaré variational auto-encoders. In *NeurIPS*.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. 2017. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. In *ICML*.

Yoshihiro Nagano, Shoichiro Yamaguchi, Yasuhiro Fujita, and Masanori Koyama. 2019. A wrapped normal distribution on hyperbolic space for gradient-based learning. In *ICML*.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*.

Ivan Ovinnikov. 2019. Poincar\'e wasserstein autoencoder. *arXiv preprint arXiv:1901.01427*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Ali Razavi, Aäron van den Oord, Ben Poole, and Oriol Vinyals. 2019. Preventing posterior collapse with delta-vaes. In *ICLR*.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*.

R Tyrrell Rockafellar et al. 1966. Extension of fenchel'duality theorem for convex functions. *Duke mathematical journal*.

Rik Sarkar. 2011. Low distortion delaunay embedding of trees in hyperbolic plane. In *International Symposium on Graph Drawing*, pages 355–366. Springer.

Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek Abdelzaher. 2020. Controlvae: Controllable variational autoencoder. In *ICML*.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *NeurIPS*.

Wenxian Shi, Hao Zhou, Ning Miao, Shenjian Zhao, and Lei Li. 2019. Fixing gaussian mixture vaes for interpretable text generation. *arXiv preprint arXiv:1906.06719*.

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

Jakub M Tomczak and Max Welling. 2018. Vae with a vampprior. In *AISTATS*.

Abraham Albert Ungar. 2008. A gyrovector space approach to hyperbolic geometry. *Synthesis Lectures on Mathematics and Statistics*, 1(1):1–194.

Wenlin Wang, Zhe Gan, Hongteng Xu, Ruiyi Zhang, Guoyin Wang, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Topic-guided variational autoencoders for text generation. *arXiv preprint arXiv:1903.07137*.

Jiacheng Xu and Greg Durrett. 2018. Spherical latent spaces for stable variational autoencoders. In *EMNLP*.

Zichao Yang, Zhiting Hu, Ruslan Salakhutdinov, and Taylor Berg-Kirkpatrick. 2017. Improved variational autoencoders for text modeling using dilated convolutions. In *ICML*.

Jake Zhao, Yoon Kim, Kelly Zhang, Alexander M Rush, and Yann LeCun. 2017a. Adversarially regularized autoencoders. *arXiv preprint arXiv:1706.04223*.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. 2018. Infovae: Information maximizing variational autoencoders. In *AAAI*.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017b. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*.

Zachary M Ziegler and Alexander M Rush. 2019. Latent normalizing flows for discrete sequences. In *ICML*.