HypMix: Hyperbolic Interpolative Data Augmentation

Ramit Sawhney[†‡*], **Megh Thakkar**[§*], **Shivam Agarwal**[‡], **Di Jin**[★], **Diyi Yang**[△], **Lucie Flek**[‡]
[†]ShareChat
[‡]Conversational AI and Social Analytics (CAISA) Lab, University of Marburg
[§]BITS, Pilani
[★]Amazon Alexa AI
[△]Georgia Institute of Technology
ramitsawhney@sharechat.co, lucie.flek@uni-marburg.de

Abstract

Interpolation-based regularisation methods for data augmentation have proven to be effective for various tasks and modalities. These methods involve performing mathematical operations over the raw input samples or their latent states representations - vectors that often possess complex hierarchical geometries. However, these operations are performed in the Euclidean space, simplifying these representations, which may lead to distorted and noisy interpolations. We propose HypMix, a novel model-, data-, and modality-agnostic interpolative data augmentation technique operating in the hyperbolic space, which captures the complex geometry of input and hidden state hierarchies better than its contemporaries. We evaluate HypMix on benchmark and low resource datasets across speech, text, and vision modalities, showing that HypMix consistently outperforms state-of-the-art data augmentation techniques. In addition, we demonstrate the use of HypMix in semi-supervised settings. We further probe into the adversarial robustness and qualitative inferences we draw from HypMix that elucidate the efficacy of the Riemannian hyperbolic manifolds for interpolation-based data augmentation.

1 Introduction

Deep learning methods have improved the state-of-the-art in a wide range of tasks. Yet, when only limited training data is available, they are prone to overfitting (Zou and Gu, 2019). Numerous data augmentation techniques have been proposed, which involve performing operations such as cropping or rotation (Lecun et al., 1998), or paraphrasing (Kumar et al., 2019) individual examples. However, these methods are modality- or dataset-dependent and require domain expertise. Compared to such alteration-based methods, interpolation-based approaches such as Mixup (Zhang et al., 2018) have shown improved performance and generalizability across different modalities. Mixup generates virtual training samples from convex combinations of individual inputs and labels to expand the training distribution. Performing Mixup over the latent representations of inputs has led to further improvements, as the hidden states of deep neural networks carry more information than raw input samples, (Verma et al., 2019a; Chen et al., 2020a). However, most data augmentation methods can only utilize existing labeled data.

Semi-supervised learning methods, on the other hand, can leverage unlabeled data for training. Several semi-supervised methods use interpolation based regularization methods over unlabeled samples to predict soft labels, and combine them with existing labeled samples to increase the overall training data (Verma et al., 2019b; Chen et al., 2020b). Semi-supervised methods use consistency based regularization training (Miyato et al., 2019) which makes the model predictions robust to perturbations on unlabeled samples. However, current semi-supervised learning methods do not generalize across modalities or datasets.

Existing data-augmentation and semi-supervised learning methods operate in the Euclidean space, which is a simplified representative geometry. Representations across modalities inherently possess properties that the Euclidean space is incapable of modeling, and can be better expressed using the more general hyperbolic space (Ganea et al., 2018). The interference of sound waves is hyperbolic in nature, which generates hyperboloid waveforms

[*] Equal contribution.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 9858–9868
November 7–11, 2021. ©2021 Association for Computational Linguistics

(Khan and Panigrahi, 2016). Natural language text exhibits hierarchical structure in a variety of respects and embeddings are more expressive when represented in the hyperbolic space (Dhingra et al., 2018). Data augmentation using Möbius operations over images has shown more diversification and generalization compared to Euclidean operations (Zhou et al., 2021). Performing interpolative operations over representations having complex geometry in the hyperbolic space can lead to more suitable representations for model training.

Building on prior research in limited data and data augmentation studies, and the hyperbolic characteristics of speech, text, and vision, we propose HYPMIX[1]: a model, data, and modality agnostic interpolative regularization method operating in the hyperbolic space. We further extend HYPMIX to semi-supervised settings, which is especially effective in extremely low resource environments. We probe the effectiveness of HYPMIX through extensive experiments over three different tasks for supervised and semi-supervised settings on benchmark and low resource datasets across speech, text, and vision in different languages with varying class label distribution. HYPMIX outperforms current state-of-the-art modality and task specific data augmentation methods across all the datasets for both supervised and semi-supervised conditions.

Our contributions can be summarized as:

- We propose HYPMIX, a novel model, data, and modality agnostic interpolative regularization based data augmentation method functioning in the hyperbolic space.

- We devise a novel Möbius Gyromidpoint Label Estimation (MGLE) method to predict soft labels for unlabeled data, and extend HYPMIX to a hyperbolic semi-supervised learning method.

- HYPMIX outperforms several strong baselines and Euclidean counterparts across speech, text, and vision across benchmark and low-resource datasets, including semi-supervised settings for Urdu and Arabic tasks.

- We further probe the effectiveness of HYPMIX in comparison to existing methods through layer-wise ablation studies and adversarial robustness.

2 Background and Related Work

Data Augmentation enables use of limited training data, with approaches involving modifying the individual training instances, such as cropping (Simonyan and Zisserman, 2015) or paraphrasing (Wei and Zou, 2019; Kumar et al., 2019). Mixup techniques (Zhu et al., 2019) perform interpolation among input samples and have proven to perform better than modifying individual instances as it incorporates the prior knowledge that linear interpolations of feature vectors should lead to linear interpolations of the associated targets. Recent works (Jindal et al., 2020a; Verma et al., 2019a) perform Mixup operations over hidden state representation of input samples instead of the inputs, as high-level representations are often low-dimensional and carry more useful information of input samples as compared to raw inputs. Latent interpolation methods have not been generalized across modalities and operate in the simplified Euclidean space which is unable to capture the complex characteristics possessed by latent state representations.

Semi-supervised Learning methods leverage unlabeled data which is typically available in larger quantities (Clark et al., 2018). Consistency regularization methods for semi-supervised learning predict soft labels for unlabeled data and train models on different permutations of labeled and unlabeled data (Verma et al., 2019b; Chen et al., 2020a). Chen et al. (2020b) uses a label guessing strategy on different augmentations of unlabeled data and combines it with labeled data for training models. However, these methods perform label prediction for unlabeled data using Euclidean operations.

Hyperbolic Learning has proven to be effective in representing information where relations among data points possess hierarchical and tree-like nature (Aldecoa et al., 2015). Learning in the hyperbolic space has been applied to various natural language processing (Dhingra et al., 2018; Gulcehre et al., 2019; Tay et al., 2018), and computer vision tasks (Khrulkov et al., 2020; Peng et al., 2020) as well as graph (Chami et al., 2019), sequence (Tay et al., 2018), and financial (Sawhney et al., 2021) learning. However, the ability of the hyperbolic space to model complex representations while performing interpolative operations across modalities is unexplored.

[1] Our code is available at: `https://github.com/caisa-lab/hypmix-emnlp`.

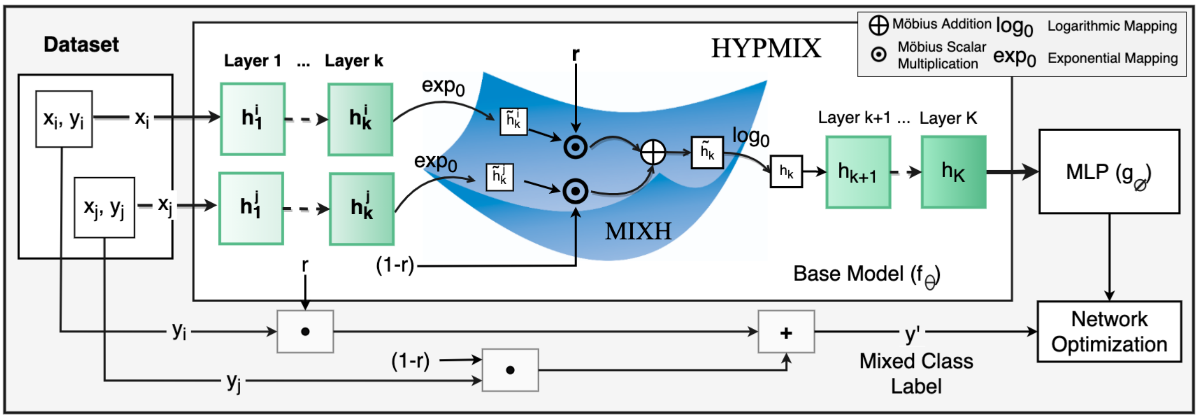

Figure 1: Overview of HypMix and MixH applied at layer k over hidden representations of input x_i and x_j. We perform the forward pass for the inputs upto layer k, and use the mixed representation for the continued pass.

3 Methodology: HypMix

We first formulate the task and introduce the hyperbolic space (Ganea et al., 2018) and Mixup (Zhang et al., 2018) (§3.1). Using the hyperbolic operations, we then introduce Mixup in the hyperbolic space (§3.2), and extend it to operate on the hidden state representations of neural networks. We call the resulting approach HypMix. An overview of the steps is presented in Figure 1. We formulate HypMix for both supervised (§3.3) and semi-supervised (§3.4) methods. We test HypMix on classification tasks across speech, text, and vision.

3.1 Preliminaries

Hyperbolic Space is a non-Euclidean geometry with constant negative curvature. We use the Poincaré ball model of the hyperbolic space (Ganea et al., 2018), defined as $(\mathcal{B}, g_x^{\mathcal{B}})$, where the manifold $\mathcal{B} = \{x \in \mathbb{R}^n : ||x|| < 1\}$, endowed with the Riemannian metric $g_x^{\mathcal{B}} = \lambda_x^2 g^E$, where the conformal factor $\lambda_x = \frac{2}{1-||x||^2}$ and $g^E = \text{diag}[1, .., 1]$ is the Euclidean metric tensor. We denote the tangent space centered at point $\boldsymbol{x}$ as $\mathcal{T}_{\boldsymbol{x}}\mathcal{B}$. We use the Möbius gyrovector space to generalize standard mathematical operations to the hyperbolic space: Möbius Addition, $\oplus$ for a pair of points $x, y \in \mathcal{B}$,

$$x \oplus y := \frac{(1 + 2\langle x, y\rangle + ||y||^2)x + (1 - ||x||^2)y}{1 + 2\langle x, y\rangle + ||x||^2||y||^2} \qquad (1)$$

where, $\langle ., . \rangle$ denotes the Euclidean inner product given by $\langle x, y\rangle = x_0y_0 + x_1y_1 + \ldots x_{n-1}y_{n-1}$, and $|| \cdot ||$ denotes the norm given by $||x|| = \sqrt{\langle x, x\rangle}$.

We define the exponential and logarithmic maps to project vectors between the Euclidean and hyperbolic space respectively.

Exponential Mapping[2] maps the tangent vector $\boldsymbol{v}$ to the point $\exp_{\boldsymbol{x}}(\boldsymbol{v})$ on the Poincaré ball,

$$\exp_{\boldsymbol{x}}(\boldsymbol{v}) := \boldsymbol{x} \oplus \left(\tanh\left(\frac{\lambda_{\boldsymbol{x}}||\boldsymbol{v}||}{2}\right)\frac{\boldsymbol{v}}{||\boldsymbol{v}||}\right) \qquad (2)$$

Logarithmic Mapping maps a point $\boldsymbol{y} \in \mathcal{B}$ to a point $\log_{\boldsymbol{x}}(\boldsymbol{y})$ on the tangent space at $\boldsymbol{x}$,

$$\log_{\boldsymbol{x}}(\boldsymbol{y}) := \frac{2}{\lambda_{\boldsymbol{x}}}\text{arctanh}\left(||-\boldsymbol{x} \oplus \boldsymbol{y}||\right)\frac{-\boldsymbol{x} \oplus \boldsymbol{y}}{||-\boldsymbol{x} \oplus \boldsymbol{y}||} \qquad (3)$$

For exponential and logarithmic mapping, we choose the tangent space center $\boldsymbol{x} = 0$ and use $\exp_{\mathbf{0}}(\cdot)$ and $\log_{\mathbf{0}}(\cdot)$.
Möbius Scalar Multiplication $\odot$ multiplies matrix $\boldsymbol{x} \in \mathcal{B}$ with scalar $r \in \mathcal{B}$,

$$r \odot \boldsymbol{x} = \tan\left(r \tan^{-1}(\|\boldsymbol{x}\|)\right)\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \qquad (4)$$

Mobius Gyromidpoint M_g calculates the hyperbolic weighted addition for gyrovectors $\{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}$ and weights $\{\alpha_1, \ldots, \alpha_n\}$,

$$\begin{aligned} M_g(\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}, \alpha_1, \alpha_2, \ldots, \alpha_n) = \\ (\boldsymbol{x_1} \odot \alpha_1) \oplus (\boldsymbol{x_2} \odot \alpha_2) \ldots \oplus (\boldsymbol{x_n} \odot \alpha_n) \end{aligned} \qquad (5)$$

Mixup (Zhang et al., 2018) involves training a neural network on convex combinations of a pair of instances and their labels. For two labeled data points (x_i, y_i) and (x_j, y_j), *mixup* uses linear interpolation with mixing ratio r to generate the synthetic sample x' and corresponding mixed label y',

$$\begin{aligned} x' &= \text{mix}(x_i, x_j) = r \cdot x_i + (1 - r) \cdot x_j \\ y' &= \text{mix}(y_i, y_j) = r \cdot y_i + (1 - r) \cdot y_j \end{aligned} \qquad (6)$$

By leveraging the hyperbolic operations and Mixup, we define Mixup in the hyperbolic space.

[2] We use the implementation by geoopt: https://geoopt.readthedocs.io/

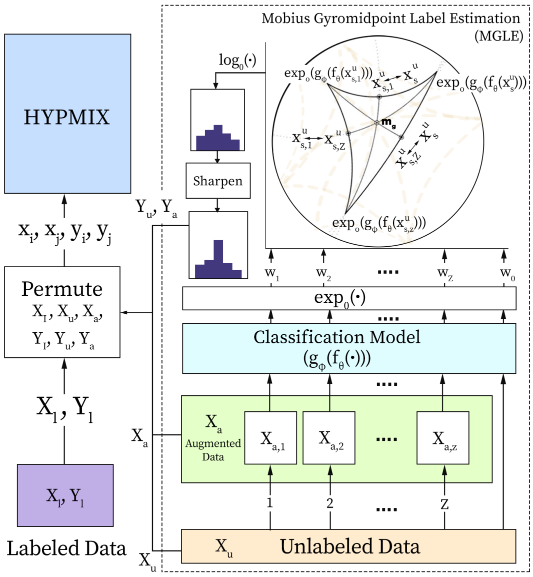
Mobius Gyromidpoint Label Estimation (MGLE)
HYPMIX
x_i, x_j, y_i, y_j
Permute
$X_l, X_u, X_a, Y_l, Y_u, Y_a$
X_l, Y_l
X_l, Y_l
Labeled Data
$\log_0(\cdot)$
Sharpen
Y_u, Y_a
X_a
X_u
$\exp_o(g_\phi(f_\theta(x^u_{s,1})))$
$\exp_o(g_\phi(f_\theta(x^u_s)))$
$\exp_o(g_\phi(f_\theta(x^u_{s,z})))$
$X^u_{s,1} \leftrightarrow X^u_s$
$X^u_{s,1} \leftrightarrow X^u_{s,Z}$
$X^u_{s,Z} \leftrightarrow X^u_s$
m_g
w_1
w_2
....
w_Z
w_0
$\exp_0(\cdot)$
Classification Model
$(g_\phi(f_\theta(\cdot)))$
X_a
Augmented Data
$X_{a,1}$
$X_{a,2}$
$X_{a,z}$
1
2
Z
X_u
Unlabeled Data

Figure 2: An overview of MGLE and hyperbolic semi-supervised learning with HypMix.

3.2 Formulating Mixup in Hyperbolic Space

For inputs possessing complex geometrical properties, performing mathematical operations in the Euclidean spaces often lead to vectorial distortions which can be stabilized by using the hyperbolic space (Ganea et al., 2018). To minimize these distortions, we formulate MixH, Mixup in the hyperbolic space by leveraging hyperbolic operations as building blocks. First, we replace Euclidean arithmetic addition ($+$) and scalar product ($\cdot$) with their Möbius counterparts: addition ($\oplus$) and scalar multiplication ($\odot$) respectively. We then transform inputs to the hyperbolic space using the exponential mapping $\exp_{\mathbf{0}}(\cdot)$, perform Mixup to generate convex combinations of pairs of inputs x_i, x_j, and map them back to the Euclidean space using the logarithmic mapping $\log_{\mathbf{0}}(\cdot)$. Formally,

$$\text{MixH}(x_i, x_j) = \log_{\mathbf{0}}(r \odot \exp_{\mathbf{0}}(x_i) \oplus (1-r) \odot \exp_{\mathbf{0}}(x_j)) \tag{7}$$

We now extend MixH as a generalizable interpolative regularizer over hidden state representations across neural network layers.

3.3 HypMix: Interpolative MixH

Previous works (Chen et al., 2020b; Jindal et al., 2020b) applying interpolation based regularization in the latent space of neural networks operate in the Euclidean space, which cannot capture the complex geometries of hidden state vectors (Tifrea et al., 2019). To better model the fine-grained information present in latent representations using the hyperbolic space, we extend MixH to the hidden representation space. Let $f_\theta(\cdot)$ denote *any* general base model with parameters θ having N layers. $f_{\theta,n}(\cdot)$ denotes the n-th layer of the model and h_n is the hidden space vector at layer n for $n \in [1, N]$ and h_0 denotes the input vector.

We introduce HypMix as hyperbolic interpolation at a layer $k \sim [1, N]$, for which we first calculate the latent representations separately for the inputs for layers before the k-th layer. For input samples x_i, x_j, we let h^i_n, h^j_n denote their respective hidden state representations at layer n of $f_\theta(\cdot)$,

$$\begin{aligned} h^i_n &= f_{\theta,n}(h^i_{n-1}), \quad & n \in [1, k] \\ h^j_n &= f_{\theta,n}(h^j_{n-1}), \quad & n \in [1, k] \end{aligned} \tag{8}$$

We then apply MixH over the individual hidden state representations h^i_k, h^j_k from layer k as:

$$\begin{aligned} h_k &= \text{MixH}(h^i_k, h^j_k) \\ &= \log_{\mathbf{0}}(r \odot \exp_{\mathbf{0}}(h^i_k) \oplus (1-r) \odot \exp_{\mathbf{0}}(h^j_k)) \end{aligned} \tag{9}$$

The mixed hidden representation h_k is used as the input for the continuing forward pass,

$$h_n = f_{\theta,n}(h_{n-1}); \quad n \in [k+1, N] \tag{10}$$

We define $\text{HypMix}(f_\theta(\cdot), r, k)$ for a layer k and mixing ratio r to obtain the final hidden layer representation h_N as,

$$h_N = \text{HypMix}(x_i, x_j, f_\theta(\cdot), r, k) \tag{11}$$

Supervised Network Optimization For classification, we apply a perceptron $g_\phi(\cdot)$ with parameters ϕ to calculate the class logits from the final hidden state output h_N. We optimize the model using KL-divergence loss (KL) to bring the model output distribution closer to the mixed label distribution. We minimize the loss L between the mixed label y' and logits obtained from HypMix,

$$L = \text{KL}(\text{mix}(y_i, y_j) || g_\phi(\text{HypMix}(x_i, x_j, f_\theta(\cdot), r, k))) \tag{12}$$

3.4 Hyperbolic Semi-supervised Learning

Semi-supervised training methods leverage unlabeled data to improve the training for limited or low resource settings (Verma et al., 2019b). We extend HypMix to effectively utilize p labeled data points, $X_l = \{x^l_1, x^l_2, \ldots, x^l_p\}$ and q unlabeled data points,

$X_u = \{x_1^u, x_2^u, \ldots, x_q^u\}$ using a semi-supervised training strategy in the hyperbolic space (Figure 2).

We first use existing data augmentation techniques across different modalities to increase the unlabeled training data X_u. For an unlabeled sample x_s^u, we generate Z augmented samples using different augmentation methods such as backtranslation (Edunov et al., 2018) and combine them to generate unlabeled augmented sets, $X_a = \{X_{a,1}, X_{a,2}, \ldots, X_{a,Z} | X_{a,z} = \{x_{1,z}^u, x_{2,z}^u, \ldots, x_{q,z}^u\}, z \in [1, Z]\}$.

Möbius Gyromidpoint Label Estimation (MGLE) predicts soft logits for unlabeled and augmented data in the hyperbolic space, allowing us to combine the unlabeled data with limited training data using HypMix for training. For an unlabeled sample x_s^u and corresponding augmented samples $\{x_{s,1}^u, x_{s,2}^u, \ldots, x_{s,Z}^u\}$, we compute the Möbius Gyromidpoint M_g of the hyperbolic mapped outputs $\{g_\phi(f_\theta(x_s^u)), g_\phi(f_\theta(x_{s,1}^u)), \ldots, g_\phi(f_\theta(x_{s,Z}^u))\}$ with fixed weights $\{w_o, w_1, \ldots, w_Z | w_i \in (0.5, 1.5)\}$, where weight w_o is applied to the original unlabeled sample. The weights control the contribution of different augmentation techniques based on their augmentation quality. We map the predicted output logits to the Euclidean space using $\log_{\mathbf{0}}(\cdot)$ to predict the soft logits y_s^u,

$$y_s^u = \log_{\mathbf{0}}(M_g(\exp_{\mathbf{0}}(g_\phi(f_\theta(x_s^u))), \exp_{\mathbf{0}}(g_\phi(f_\theta(x_{s,1}^u))), \ldots, \exp_{\mathbf{0}}(g_\phi(f_\theta(x_{s,Z}^u))), w_o, w_1, \ldots, w_Z)) \tag{13}$$

We sharpen the output y_s^u with a hyperparameter temperature T, to prevent it from being too uniform if the model predictions are random,

$$y_s^u = \frac{(y_s^u)^{\frac{1}{T}}}{||(y_s^u)^{\frac{1}{T}}||_1} \tag{14}$$

where $||\cdot||_1$ is the l_1-norm of the vector.

Semi-supervised Network Optimization For optimizing the model in semi-supervised settings, we use the training set $X = X_l \cup X_u \cup X_a$ with labels $Y = Y_l \cup Y_u \cup Y_a$, where Y_u is used for both unlabeled and augmented inputs, i.e. $Y_a = Y_u$. We then uniformly sample two elements, $x_i, x_j \sim X$ and the corresponding labels $y_i, y_j \sim Y$, and apply HypMix(x_i, x_j). We optimize the model using KL-divergence loss L over the model outputs and the mixed labels mix(y_i, y_j),

$$L = \mathrm{KL}(\mathrm{mix}(y_i, y_j) || g_\phi(\textsc{HypMix}(x_i, x_j))) \tag{15}$$

	Dataset	Class Labels	# Classes	# Samples
Speech	ESC-10 (2015)	Sound Source	10	400
	US8K (2017)	Sound Source	10	8,732
	Urdu SER (2018)	Emotion	4	400
Text	AG News (2018)	News Topic	4	127,600
	DB Pedia (2012)	Wiki Topic	14	630,000
	Arabic HS (2018)	Hate Speech	2	3,950
Vis.	CIFAR-10 (2009)	Object	10	60,000
	CIFAR-100 (2009)	Object	100	60,000

Table 1: Datasets, tasks, # classes and # samples.

4 Experimental Setup

4.1 Datasets and Preprocessing

We consider benchmark and low-resource datasets across speech, text, and vision spanning a varying number of classes, languages, and class imbalances for a comprehensive evaluation of HypMix. We choose these datasets based on existing works across different task settings and baselines for a fair comparison with HypMix. We also choose datasets with comparatively lower language resources, different structures, and language roots, leading to a more diverse evaluation of HypMix. We summarize dataset statistics in Table 1. We follow the same preprocessing across all datasets as done by previous works (Jindal et al., 2020b), (Chen et al., 2020b), (Verma et al., 2019a).

4.2 Task Setup

We evaluate HypMix on three different settings for an extensive analysis: supervised training with limited training data, semi-supervised training with low resource data, and a fully supervised setup with complete training data.

Speech Following previous works, we use EnvNet-v2 with strong augmentation (Tokozume et al., 2018) as our base architecture $f_\theta(\cdot)$ followed by a fully connected layer $g_\phi(\cdot)$. We modify MixH to account for the auditory perception and amplitude of speech signals (Tokozume et al., 2018). We use Fourier and Inverse Fourier Transform to generate augmented samples. We compare HypMix with the current state-of-the-art method Speechmix (Jindal et al., 2020b) across multiple settings.

Text Following Chen et al. (2020b), we use BERT-base (Devlin et al., 2019) as the backbone architecture $(f_\theta(\cdot))$ for English datasets and BERT-base-arabic (Safaya et al., 2020) for the Arabic dataset. We use a two layer MLP with hidden size 128 as the classifier $(g_\phi(\cdot))$ and generate augmented data using back-translation (Edunov et al.,

Model	**Speech** (Error rate ↓)						**Text** (Error rate ↓)						**Vision** (Error rate ↓)					
	ESC-10			Urdu SER			AG News		DBPedia		Arabic HS		CIFAR-10			CIFAR-100		
#Samples n	5	10	15	5	10	20	10	200	10	200	50	100	10	100	500	10	100	500
Base Model	47.1	32.5	26.2	28.7	23.8	18.2	30.5	12.5	4.8	1.5	42.1	40.6	65.7	34.2	14.9	79.7	41.1	21.2
+ EucMix	33.2	24.2	21.0	18.8	16.5	13.7	25.9	11.9	3.2	1.3	41.1	39.6	64.6	32.3	**14.1**	78.8	40.3	**20.3**
+ HypMix	**30.3**$^\diamond$	**22.2**$^\diamond$	**19.5**$^\diamond$	**16.5**$^\diamond$	**15.2**$^\diamond$	**12.5**$^\diamond$	**21.8**$^\diamond$	**11.8**	**3.0**$^\diamond$	**1.2**	**40.3**$^\diamond$	**38.9**$^\diamond$	**63.7**$^\diamond$	**30.9**$^\diamond$	14.5	**77.9**$^\diamond$	**39.4**$^\diamond$	20.5

Table 2: Performance comparison of HypMix on limited data. EucMix is Euclidean mixup: SpeechMix (Jindal et al., 2020b) for sound, TMix (Chen et al., 2020b) for text and Manifold Mixup (Verma et al., 2019a) for vision. n is the number of labeled training samples per class. Improvements are shown with blue (↓). **Bold** shows the best result. $\diamond$ shows significant ($p < 0.01$) improvement over EucMix methods under Wilcoxon's signed-rank test.

2018). We compare HypMix with TMix and MixText (Chen et al., 2020b) for supervised and semi-supervised training respectively.

Vision Following Verma et al. (2019a), we use PreActResNet18 (He et al., 2016) as the backbone architecture ($f_\theta(\cdot)$) and a linear layer as the classifier ($g_\phi(\cdot)$). We compare HypMix with manifold mixup (Verma et al., 2019a) for different settings.

4.3 Training Setup

Speech We use Nesterov's accelerated gradient (Sutskever et al., 2013) using momentum of 0.9, weight decay of $5e-4$, learning rate of 0.01 and mini-batch size of 64 for 2000 epochs. For ESC-10, we train the model on 5 folds, and for UrbanSound8k, we train the model on 10 folds to report the average error rate. We randomly sample the mixing ratio from a uniform distribution, $r \sim U(0, 1)$. For semi-supervised training, we use 50 unlabeled samples from each class.

Text We use AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate $1e-5$ for the BERT encoder and $1e-3$ for the MLP. We follow Chen et al. (2020b) to sample the mixing ratio r from a beta distribution based on the number of labeled samples. For semi-supervised setting, we use 1000 unlabeled samples from each class.

Vision We use Nesterov's accelerated gradient (Sutskever et al., 2013) using momentum of 0.9 and learning rate of 0.1, batch size of 100 to train for 2000 epochs. Following Verma et al. (2019a), we sample the mixing ratio $r \sim \text{Beta}(2, 2)$, where Beta denotes the Beta distribution.

5 Results and Analysis

5.1 Supervised Training with Limited Data

We compare HypMix in a limited training data setup with baseline methods in Table 2. We observe that Euclidean mixup techniques (EucMix) improve the performance over base models, indicating the importance of using the latent representation space of neural network architectures to perform interpolative regularization (Verma et al., 2019a). HypMix further improves performance ($p < 0.01$) over Euclidean methods across modalities, validating that the hyperbolic space is able tp better capture the complex geometry of latent representations for different inputs when performing interpolative operations.

HypMix shows maximum improvement when applied on extremely low training data, with samples in order of $n = 10$. This is in line with works (Zhou et al., 2021) which suggest that the variation generated by Möbius operations is very high as compared to Euclidean operations, leading to much more diverse samples from a small training set. This paves a path for better utilization of low resource datasets for downstream tasks across different modalities by leveraging the hyperbolic space. For all modalities, the relative improvement over the baseline architecture reduces with increasing number of labeled samples per class (n). This is in line with works (Verma et al., 2019b; Chen et al., 2020b) observing similar trends, suggesting that with an increase in number of labeled samples, the overall diversity of interpolative representations saturates, leading to lower relative improvements.

Across modalities, we observe maximum improvement when HypMix is applied on speech datasets, since speech waves inherently possess hyperbolic nature (Khan and Panigrahi, 2016), and their interpolative augmentation closely resembles hyperbolic wave interference (Chaturvedi et al., 1998). Improvements due to HypMix on text datasets ties with works stating that text inherently displays tree-like hierarchical characteristics and can be better represented using Riemannian ge-

Model	Urdu SER (↓)			Arabic HS (↓)		
#Samples n	5	10	20	10	50	100
Base Model	28.7	23.8	18.2	45.5	42.1	40.6
Supervised Training Methods						
+ EUCMIX	18.8	16.5	13.7	44.6	41.1	39.6
+ HYPMIX	16.5*	15.2*	12.5*	44.0*	40.3*	38.9*
Semi-supervised Training Methods						
+ EUCMIX	17.5	13.7	8.7	43.1	38.4	36.6
+ HYPMIX	**15.0**$^\diamond$	**12.5**$^\diamond$	**5.0**$^\diamond$	**42.1**$^\diamond$	**37.5**$^\diamond$	**35.7**$^\diamond$

Table 3: Performance comparison of HypMix in semi-supervised settings with state-of-the-art methods in terms of % Error rate. n is the number of labeled training samples per class. Improvements are shown with blue (↓) and poorer performance with red (↑). **Bold** shows the best result. *, $\diamond$ show significant ($p < 0.01$) improvement over Euclidean supervised and semi-supervised methods, respectively, under Wilcoxon's signed-rank test.

ometry (Tifrea et al., 2019). The improvements on vision datasets are in line with works suggesting that performing augmentation operations over images using Möbius operations improves generalization while increasing diversity as compared to simplified Euclidean operations (Zhou et al., 2021). Improvements across different modalities, datasets, and base architectures indicate the modality, data, and model agnostic nature of HYPMIX.

5.2 Semi-Supervised Results: Low-Resource

We probe the effect of using hyperbolic semi-supervised learning (§3.4) for low resource datasets using HYPMIX in Table 3. Using semi-supervised learning shows significant improvements over their supervised counterparts trained with limited data for both Euclidean and hyperbolic (HYPMIX) representations, indicating the importance of using unlabeled and augmented data as additional training data. For both Euclidean and hyperbolic methods, we see larger improvements with increasing the number of labeled samples n, due to the increased number of permutations of labeled-labeled, unlabeled-labeled, and unlabeled-unlabeled samples encountered during training. We observe greater improvements when semi-supervised training is applied in the hyperbolic space with HYPMIX (Figure 3) for both speech and text as compared to EUCMIX, indicating that the hyperbolic space is able to generate less noisy, yet more diverse samples by effectively modeling the complex latent space representations.

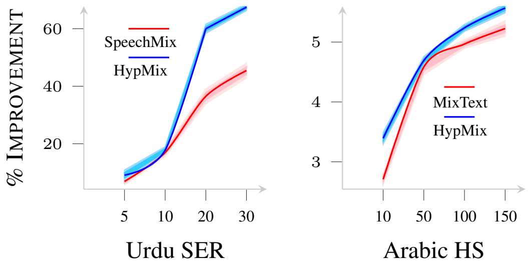

Figure 3: % improvement with semi-supervised learning with different resource settings in terms of labeled samples per class. Red denotes EUCMIX and blue denotes HYPMIX.

Layer Set	EUCMIX(↓)	HYPMIX(↓)
{2, 4, 6, 8}	7.9	**6.3**
{4, 6, 8}	9.2	8.8
{0, 2, 4, 6}	8.5	7.7
{2, 4, 6}	**7.1**	7.5

Table 4: Layer-wise ablation (% Error rate) on ESC-10.

Across modalities, speech datasets that are augmented with simpler methods such as mathematical transforms show larger improvements as compared to text datasets that are augmented with more complicated methods like backtranslation. We attribute this difference to the proximity of augmented unlabeled samples to the original unlabeled data distribution, suggesting that better augmentation methods and controlling the weights for Möbius Gyromidpoint Label Estimation (MGLE, §3.4) based on the augmentation quality is an important factor for the performance of semi-supervised methods.

5.3 Layer wise Ablation

We experiment with different sets of layers from which we uniformly sample k to perform HYPMIX. We experiment with the best performing layer sets from corresponding previous works (Jindal et al., 2020b; Chen et al., 2020b) for a fair comparison.

Speech Table 4 compares the error rates on the ESC-10 dataset for Speechmix (Jindal et al., 2020b) and HYPMIX. We observe that HYPMIX achieves the best performance when the layer set has layers performing a max-pool operation in EnvNet-v2. These layers capture different features of sound such as frequency response and auditory perception (Tokozume et al., 2018), suggesting that HYPMIX is able to extend the training distribution by modeling various combinations of latent speech vectors representing different auditory features using hyperbolic interpolation.

Layer Set	EucMix(↓)	HypMix(↓)
ϕ	30.5	30.5
$\{7, 9, 12\}$	**25.9**	22.7
$\{6, 7, 9, 12\}$	27.8	**21.8**
$\{6, 7, 9\}$	28.1	24.9

Table 5: Layer-wise ablation (% Error rate) on AG News with $n = 10$ labeled samples per class.

Text We compare different layer sets of BERT-base (Devlin et al., 2019) for performing HypMix for text datasets in Table 5. Layers $\{3, 4, 6, 7, 9, 12\}$ of BERT-base contain the most information about different aspects of natural language (Jawahar et al., 2019). We experiment with different combinations of the layer set $\{3, 4, 6, 7, 9, 12\}$. EucMix achieves the best result when using the set $\{7, 9, 12\}$ for interpolation, layers containing the semantic and syntactic information. HypMix is able to better capture the syntax tree information present in layer 6 (Jawahar et al., 2019) and shows higher improvements when the mixup layer is chosen from $\{6, 7, 9, 12\}$, validating the ability of the hyperbolic space to model hierarchical information better than the Euclidean space (Ganea et al., 2018).

During the layer-wise ablation study, we observe that even though there is intersection between the optimum layer sets of EucMix and HypMix, they are not exactly the same. This leads to interesting questions regarding the representations that Euclidean and hyperbolic spaces capture, and how can the hyperbolic space be further exploited for modeling complex geometries.

5.4 Supervised HypMix with Complete Data

Model	ESC10 (↓)	US8K (↓)	UrduSER (↓)
EnvNet-v2 (2018)	10.9	24.9	10.1
+ BC Learning (2018)	8.6	21.7	8.7
+ SpeechMix (2020b)	7.1	20.8	6.2
+ HypMix-Input	$6.5^{\diamond}$	20.9	$5.0^{\diamond}$
+ HypMix	**6.3**$^{\diamond}$	**20.4**$^{\diamond}$	**2.5**$^{\diamond}$

Table 6: Performance comparison in terms of % Error rate(↓) of HypMix with baselines in supervised settings with full training data. **Bold** shows the best result. $\diamond$ show significant ($p < 0.01$) improvement over previous state-of-the-art method under Wilcoxon's signed-rank test.

We compare the performance of HypMix for three benchmark and low resource speech datasets in Table 6 by applying BC Learning (Tokozume et al., 2018), and Speechmix (Jindal et al., 2020b) over EnvNet-v2 with strong augmentation (Tokozume et al., 2018). We observe that mixup-based approaches, i.e., BC learning and Speechmix improve the performance over the standard learning models, validating the importance of interpolative acoustic mixup based on the auditory perception of input samples. HypMix achieves state-of-the-art performance ($p < 0.01$) across all three datasets, suggesting that the hyperbolic representation better models the latent representation of speech signals and acoustic wave interference, compared to the Euclidean space. We also present the results of HypMix-Input, where we perform HypMix over the raw inputs instead of latent representations. HypMix-Input outperforms SpeechMix for two datasets, suggesting that the hyperbolic input space itself is able to generate diverse synthetic samples as compared to Euclidean methods.

5.5 Robustness to Adversarial Attacks

Method	ESC-10		Urdu SER	
	FGSM	I-FGSM	FGSM	I-FGSM
EucMix	87.92	97.50	68.74	82.50
HypMix	82.75	97.47	66.24	77.50

Table 7: Classification errors on adversarial examples generated using white box FGSM and I-FGSM attacks.

Adversarial attacks provide inputs to models specifically designed to confuse them. We compare the robustness of HypMix and HypMix-Input with BC Learning (Tokozume et al., 2018) and Speechmix (Jindal et al., 2020b) by performing white box adversarial attacks using Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) and Iterative Fast Gradient Sign Method (I-FGSM) (Kurakin et al., 2016) in Table 7. We observe that HypMix is more robust by 6.1% and HypMix-Input is robust by 5.8% compared to their Euclidean counterparts, indicating that the hyperbolic space helps the model generalize better and make it more resistant to adversarial examples.

5.6 Cost of Hyperbolic Operations

HypMix requires additional hyperbolic transformations such as exponential and logarithmic mappings and tangential and hyperbolic tangential operations on-top of EucMix. However, on a GPU, they can be carried out in parallel. Hence, HypMix requires longer time only by a constant factor compared to EucMix, with the individual operations having similar time complexity, which is of the order of the input dimensions of the latent representations of the samples to be mixed. We compare the per iteration time taken by HypMix with EucMix in Table 8.

Method	AGNews	ESC-10
EucMix	0.826	0.862
HypMix	0.869	0.893

Table 8: Time (in seconds) per iteration for EucMix and HypMix.

6 Conclusion and Future Work

Drawing inspiration from works showing that speech, text, and vision data inherently possess hyperbolic characteristics and can be better represented in the hyperbolic space, we propose HypMix, a model, data, and modality agnostic interpolative regularization method operating in the hyperbolic space. We devise a Möbius Gyromidpoint Label Estimation (MGLE) technique to predict labels for unlabeled training data and combine it with HypMix to formulate a hyperbolic semi-supervised learning method. HypMix outperforms existing methods for benchmark and low resource datasets across speech, text, and vision in supervised and semi-supervised settings with complete and limited training data. HypMix is also more robust to white-box adversarial attacks compared to Euclidean methods. HypMix being model, data, and modality agnostic can be extended to downstream tasks across modalities and interpolative augmentation for data such as sequences and graphs. As future work, we plan to evaluate HypMix on larger datasets and a variety of tasks such as the GLUE and SuperGLUE benchmarks, and tasks comprising multimodal settings.

7 Acknowledgements

This work has been supported by the German Federal Ministry of Education and Research (BMBF) as a part of the Junior AI Scientists program under the reference 01-S20060. Ramit Sawhney is supported by ShareChat. We thank the anonymous reviewers for their valuable inputs.

References

Nuha Albadi, Maram Kurdi, and Shivakant Mishra. 2018. Are they our brothers? analysis and detection of religious hate speech in the arabic twittersphere. In *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 69–76. ACM.

Rodrigo Aldecoa, Chiara Orsini, and Dmitri Krioukov. 2015. Hyperbolic graph generator. *Computer Physics Communications*, 196:492–496.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 4869–4880.

S Chaturvedi, GJ Milburn, and Zhongxi Zhang. 1998. Interference in hyperbolic space. *Physical Review A*, 57(3):1529.

Jiaao Chen, Zhenghui Wang, Ran Tian, Zichao Yang, and Diyi Yang. 2020a. Local additivity based data augmentation for semi-supervised ner. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1241–1251.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020b. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. 2018. Embedding text in hyperbolic spaces. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on*

Empirical Methods in Natural Language Processing, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Octavian Ganea, Gary Becigneul, and Thomas Hofmann. 2018. Hyperbolic neural networks. In *Advances in Neural Information Processing Systems*.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, and Nando de Freitas. 2019. Hyperbolic attention networks. In *International Conference on Learning Representations*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. In *Computer Vision – ECCV 2016*, pages 630–645, Cham. Springer International Publishing.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Amit Jindal, Arijit Ghosh Chowdhury, Aniket Didolkar, Di Jin, Ramit Sawhney, and Rajiv Ratn Shah. 2020a. Augmenting NLP models using latent feature interpolations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6931–6936, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Amit Jindal, Narayanan Elavathur Ranganatha, Aniket Didolkar, Arijit Ghosh Chowdhury, Di Jin, Ramit Sawhney, and Rajiv Ratn Shah. 2020b. Speechmix—augmenting deep sound recognition using hidden space interpolations. *Proc. Interspeech 2020*, pages 861–865.

Md Nazoor Khan and Simanchala Panigrahi. 2016. *Interference*, page 98–185. Cambridge University Press.

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. 2020. Hyperbolic image embeddings. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

A. Krizhevsky and G. Hinton. 2009. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. 2016. Adversarial examples in the physical world.

Siddique Latif, Adnan Qayyum, Muhammad Usman, and Junaid Qadir. 2018. Cross lingual speech emotion recognition: Urdu vs. western languages. In *2018 International Conference on Frontiers of Information Technology (FIT)*, pages 88–93.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

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

Pablo Mendes, Max Jakob, and Christian Bizer. 2012. DBpedia: A multilingual cross-domain knowledge base. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1813–1817, Istanbul, Turkey. European Language Resources Association (ELRA).

Takeru Miyato, Shin-Ichi Maeda, Masanori Koyama, and Shin Ishii. 2019. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993.

Wei Peng, Jingang Shi, Zhaoqiang Xia, and Guoying Zhao. 2020. Mix dimension in poincaré geometry for 3d skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 1432–1440, New York, NY, USA. Association for Computing Machinery.

Karol J. Piczak. 2015. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, New York, NY, USA. Association for Computing Machinery.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online). International Committee for Computational Linguistics.

Justin Salamon and Juan Pablo Bello. 2017. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3):279–283.

Ramit Sawhney, Shivam Agarwal, Megh Thakkar, Arnav Wadhwa, and Rajiv Ratn Shah. 2021. *Hyperbolic Online Time Stream Modeling*, page 1682–1686. Association for Computing Machinery, New York, NY, USA.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. 2013. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1139–1147, Atlanta, Georgia, USA. PMLR.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Hyperbolic representation learning for fast and efficient neural question answering. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*.

Alexandru Tifrea, Gary Becigneul, and Octavian-Eugen Ganea. 2019. Poincare glove: Hyperbolic word embeddings. In *International Conference on Learning Representations*.

Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Learning from between-class examples for deep sound recognition. In *International Conference on Learning Representations*.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. 2019a. Manifold mixup: Better representations by interpolating hidden states. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6438–6447, Long Beach, California, USA. PMLR.

Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. 2019b. Interpolation consistency training for semi-supervised learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pages 3635–3641. AAAI Press.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*.

Sharon Zhou, Jiequan Zhang, Hang Jiang, Torbjörn Lundh, and Andrew Y Ng. 2021. Data augmentation with mobius transformations. *Machine Learning: Science and Technology*, 2(2):025016.

Yingke Zhu, Tom Ko, and Brian Mak. 2019. Mixup learning strategies for text-independent speaker verification. *Proc. Interspeech 2019*.

Difan Zou and Quanquan Gu. 2019. An improved analysis of training over-parameterized deep neural networks. In *Advances in Neural Information Processing Systems*.