An End-to-End Approach for the Verification Problem Through Learned Metric-like Spaces

Anonymous[*]

1 Introduction and Motivation

Learning useful representations from high-dimensional data is one of the main goals of modern machine learning. Neural networks have been shown to be able to efficiently learn such representations without requiring specialized pre-processing of data under analysis. Remarkable results of methods employing neural networks have been published tackling classical challenging problems. However, learning useful lower dimensional features from data is generally a side effect of the solution of a given task, e.g., while learning the decision surface of a classification problem, inner layers of neural networks are shown to make salient cues of input data that are discriminable. Moreover, in an unsupervised setting, bottleneck layers of Autoencoders, learned posteriors of Variational Autoencoders or even the latent layer of Generative Adversarial Networks have all been shown to embed useful properties of input samples that can be leveraged for use in other tasks.

Rather than employing a neural network to solve some task and hope learned features are useful and transferable, approaches such as Siamese Networks (1) have been introduced with the aim at explicitly learning an embedding model that results in a lower dimensional space where samples hold relevant properties, such as class separability. Given class labels, training of the embedding model aims to minimize or maximize some L_p norm of the difference between samples in the embedding space depending on whether they belong to the same class or not. Follow-up work exploited and extended this idea for several applications. In (2), authors proposed a triplet loss alternative to the contrastive scheme proposed for Siamese Networks and showed resulting embeddings achieved high performance when classifiers were trained on top of them. However, as argued in previous literature (3), Euclidean spaces will not in general be suitable metric spaces for representing desired semantic relations from given data. Authors thus introduced embedding computation on a hyperbolic space. Inspired by that, rather than training the embedding model by minimizing/maximizing Euclidean distances between anchor-positive and anchor-negative pairs or hand-designing suitable spaces for a given dataset and desired properties of embeddings, we propose to learn a space by jointly training the mapping model, responsible for mapping a given data sample to the embedding space, along with another model serving as a distance/divergence in the learned space. Both models together, parametrized by neural networks, define a metric-like space within which desired semantic relations such as class separability are represented in terms of distance, which renders inference efficient.

2 Contribution and Results

Consider $\mathcal{M} : \mathbb{R}^D \rightarrow \mathbb{R}^d$ and $\mathcal{D} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow]0, 1[$ are deterministic functions that will be referred to as mapping and distance models, respectively, since they will be parametrized by neural networks. Data samples are such that $x \sim X \in \mathbb{R}^D$, and $z = \mathcal{M}(x)$, with $z \in \mathbb{R}^d$, represent embedded data examples, given that $D \gg d$. Each data example can be further associated to a class label $y \in \{1, ..., k\}$. Moreover, we define anchor/positive pairs of samples such that $x^{ap} = \{x^a, x^p\} : y(x^a) = y(x^p)$, as well as anchor/negative pairs $x^{an} = \{x^a, x^n\} : y(x^a) \neq y(x^n)$. We thus define

[*] Presenting author identifies as Latinx

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

the parameters of $\mathcal{M}$ and $\mathcal{D}$ as θ and ϕ, respectively, which are determined through the minimization of the following loss $\mathcal{L}$:

$$\theta, \phi = \arg\min \mathcal{L} : \mathcal{L} = -\mathbb{E}_{X^{ap}} \log(\mathcal{D}(\mathcal{M}(x^a), \mathcal{M}(x^p))) - \mathbb{E}_{X^{an}} \log(1 - \mathcal{D}(\mathcal{M}(x^a), \mathcal{M}(x^n))). \tag{1}$$

The joint training of $\mathcal{M}$ and $\mathcal{D}$ resembles the setting of a Generative Adversarial Network, with the difference that both models minimize the same loss rather than playing a min-max game. As such, this approach can be shown to be equivalent to maximizing an approximate divergence (Jensen-Shannon divergence in our case) between the joint distributions of anchor/positive and anchor/negative pairs on the embedding space under assumptions on $\mathcal{D}$. A proof of concept experiment as presented in Figure 1 shows discriminable embeddings computed by $\mathcal{M}$ on the test partition of MNIST directly on $\mathbb{R}^2$; no dimensionality reduction is applied to generate the plot.

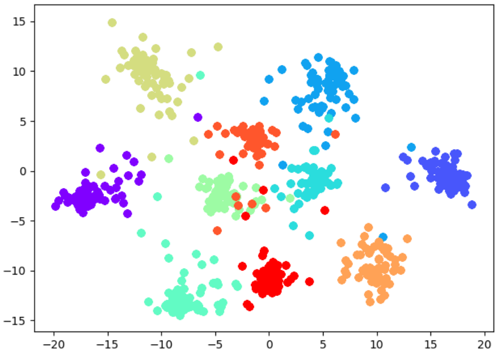

Figure 1: MNIST embeddings on a 2-dimensional space. Each color represents samples corresponding to a digit from 0 to 9. Held out data was employed.

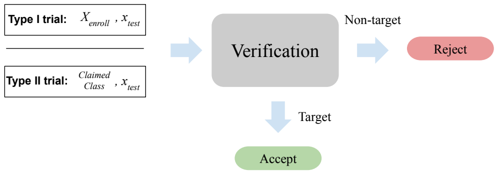
Type I trial: X_{enroll} , x_{test}
Type II trial: $\begin{smallmatrix}Claimed\\Class\end{smallmatrix}$, x_{test}
Verification
Non-target
Reject
Target
Accept

Figure 2: Illustration of the verification problem. The goal is to classify trials as either target or non-target meaning that claimed and test classes match or not, respectively. Classes appearing at time might differ from those used during training.

We evaluate the proposed approach on the *verification problem*, illustrated in Figure 2, which corresponds to, given a trial containing two examples, deciding whether such examples belong to the same class or not. The evaluation on the verification problem can be performed under the closed or open set conditions, i.e. the label set at test time matches that of training data on the closed set case, while new classes appear on test data for the open set setting. We evaluate the proposed approach on both cases and in both image and speech tasks. Results in terms of equal error rate (EER) are reported in Tables 1 for Cifar-10 (closed-set) and Mini-Imagenet (open-set), and in Table 2 for language identification on the OLR language identification task (closed-set) (4) as well as on the VoxCeleb speaker verification task (open-set) (5) in which case models are trained on VoxCeleb-2 train partition and evaluated on the entire VoxCeleb-1 train data. In all cases, a list of trials is created by pairing all examples in the test data. EER is a threshold independent performance metric for binary classification consisting of the value of the false acceptance rate at the threshold in which it matches the false rejection rate.

Evaluation is performed using the cosine similarity, a common scoring strategy within the verification literature, as well as directly using the output of $\mathcal{D}$ as a score which we refer to as **E2E** in the tables. In all considered cases, using the learned divergence as a score outperformed the cosine distance.

		EER (%)
Cifar 10	Cosine	5.02
	E2E	**3.39**
Mini-Imagenet	Cosine	28.79
	E2E	**28.48**

Table 1: Verification performance on the verification task on object classification in terms of EER (the lower the better).

		EER (%)
OLR	Cosine	4.74
	E2E	**3.70**
VoxCeleb	Cosine	7.22
	E2E	**5.87**

Table 2: Verification performance on the verification task on voice biometrics tasks in terms of EER (the lower the better)

References

[1] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proceeedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2006, pp. 1735–1742.

[2] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *International Workshop on Similarity-Based Pattern Recognition*. Springer, 2015, pp. 84–92.

[3] M. Nickel and D. Kiela, "Learning continuous hierarchies in the lorentz model of hyperbolic geometry," *arXiv preprint arXiv:1806.03417*, 2018.

[4] Z. Tang, D. Wang, Y. Chen, and Q. Chen, "Ap17-olr challenge: Data, plan, and baseline," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017*. IEEE, 2017, pp. 749–753.

[5] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.