Model Vulnerability to Distributional Shifts over Image Transformation Sets

Riccardo Volpi, Vittorio Murino

Abstract. We provide a summary of the pre-print ‘Model Vulnerability to Distributional Shifts over Image Transformation Sets’ (arxiv.org/abs/1903.11900) [22].

1. Introduction

When we devise and deploy a machine learning system, a generally desirable property is its ability to generalize to unknown scenarios. For example, we would like a vision module (e.g., for a self-driving car) to perform well under a broad variety of visual conditions and environments. Albeit, modern learning systems are well known to be vulnerable to the dataset bias issue [6, 2, 1, 19]: when trained on data from some distribution, they will typically learn the peculiar statistics of the training data, and in result will perform more poorly in different settings.

In light of this, two fundamental research directions are (i) training more robust models against the distributional shifts that they might encounter after deployment – domain adaptation [6, 2, 16, 4, 20, 18, 13, 21] and domain generalization [9, 14, 15, 17, 12, 23, 10, 11, 23] are possible directions to face this problem – and (ii) developing tools to understand the vulnerability regions of a model before its deployment. In this work, we propose methods related to the second branch, and further exploit them to devise algorithms associated with the first one.

Focusing on computer vision models, we start from the assumption that, given a set of standard, content-preserving image transformations, we can generate a huge set of possible distributional shifts by concatenating them and applying them to the datapoints at our disposal. We propose a combinatorial optimization problem aimed at detecting the concatenations of different transformations that a given (black-box) model is most vulnerable to, and face it through random search and evolution-based search.

Endowed of these tools, we propose a training procedure where, over iterations, harmful image transformations in the given set are searched, and used as data augmentation rules throughout the training procedure. Models trained with our method, not only are more resistant against the transformations from the provided set, but also better generalize to unseen scenarios. For example, we propose results associated with a semantic segmentation module trained on CamVid [3], showing that it better generalizes to foggy scenarios even though it has never encountered them during training.

2. Problem formulation

Let $\hat{M}(.)$ be a model that takes in input images and provides an output according to the given task. Let $D = \{(x^{(i)}, y^{(i)})\}_{i=0}^{m} \sim P(X, Y)$ be a set of datapoints with their labeling, drawn from some data distribution. Finally, let $\mathcal{T} = \{(\tau^{(i)}, t^{(i)}), i = 1 \ldots N_T, j = 1 \ldots N_t\}$ be a finite set, where each object $t = (\tau^{(i)}, t^{(j)})$ is a data transformation $\tau$ with a related magnitude $l$. The transformations give in output images in the same format as the input ones. We define a concatenation of different transformations as a transformation tuple; $\mathcal{T}_N$ is the set of all the possible transformation tuples that one can obtain by combining objects in $\mathcal{T}$. We define $\mathcal{T}$ with the following transformations: autocorrection (20), sharpness (20), brightness (20), color (20), contrast (20), grayscale conversion (1), R-channel enhancer (30), G-channel enhancer (30), B-channel enhancer (30), solarize (20), where the numbers in parenthesis indicate the number of different magnitude levels (Table 2 in [22]). Armed with this set, we propose the following combinatorial optimization problem

$$\min_{\mathcal{T}^* \in \mathcal{T}_N} f(\mathcal{M}, \mathcal{T}^*, D)$$

where $f$ is a function that measures the accuracy of a model $\mathcal{M}$ provided with set of labelled datapoints $D$, modified through a transformation tuple $T^*$. The $N$–tuples that
Algorithm 1 Robust Training

1: Input: \( D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N} \), init. weights \( \theta_0 \), \( N \)–tuple set \( T_N \), init. data augmentation set \( T_{tr} \), learning rate \( \alpha \), loss \( \ell \).
2: Output: learned weights \( \theta \)
3: Initialize: \( \theta \leftarrow \theta_0 \)
4: for \( j = 1, ..., J \) do
5: Minimize \( \ell \) through \( k \) SGD steps using data sampled from \( D \) modified with transformation tuples sampled from \( T_N \)
6: Find \( T^* \in T_N \) by running RS or ES on a subset of \( D \) and append it to \( T_{tr} \)

induce lower \( f \) values are the ones that a model \( M \) is more vulnerable to, with respect to the chosen metric.

Search algorithms. In order to approach the combinatorial optimization problem 1, we rely on Random Search (RS) and Evolution-based Search (ES). In RS, we test an arbitrary number of transformation tuples from the set \( T_N \) and choose the one that leads to the lowest \( f \) value. In ES, we use standard operations from the genetic algorithm literature (selection, mutation and cross-over) to find harmful transformations more efficiently; for the details, we refer to the original work [22].

3. Training more robust models

We define a training procedure to learn models (ConvNets [8]) robust against image transformations from an arbitrary set as follows: (a) we initialize a transformation set to sample from during training (the “data augmentation set” \( T_{tr} \)) with the identity transformation; (b) we train the network via gradient descent updates, augmenting samples via data augmentation procedures sampled from \( T_{tr} \); (c) we run RS or ES, using appropriate fitness function \( f \) and tuple set \( T_N \), and append the so-found transformation to \( T_{tr} \). We alternate between steps (b) and (c) for the desired number of times. See Algorithm 1 for a detailed view.

4. Experiments

In our original work [22], we have tested our search methods and training procedure on a variety of tasks. We report here the results associated with a semantic scene segmentation task (FC-DenseNet [7] with 103 layers trained on CamVid [3]), and refer to [22] for the other results (as well as for the description of the hyper-parameters). In Algorithm 1, we set the number of transformations concatenated as \( N = 5 \) and use the cross-entropy function between the output of the model and the ground truth labels as loss \( \ell \).

Figure 2 shows the output of a model trained via standard Empirical Risk Minimization (middle row) and the output of our model (last row), when the original input (first column) is perturbed by different image transformations found via ES (first row). These results qualitatively show that models trained through Algorithm 1 are more resistant against image transformations; furthermore, they show that the transformations in \( T_N \) are reasonable approximations of possible visual conditions that a computer vision model might face after deployment. For example, images in the first row, second and third column, can be interpreted as simulations of the light that one could face during dawn or sunset—and in which the baseline model we compare against performs poorly.

Table 1 reports the pixel accuracy values. The columns RS and ES indicate the lowest values obtained by running the two search procedures using models trained via standard Empirical Risk Minimization (ERM) and with our method (first and second row, respectively) They confirm the higher level of robustness of models trained through Algorithm 1, using ES as search algorithm (line 6). The last column shows results obtained by manipulating the images with artificial fog [5]: also in this case, our models show better generalization properties.

5. Conclusions

We propose a combinatorial optimization problem to find harmful distributional shifts for a given model, defined in terms of concatenations of image transformations from an arbitrary set. We show that random search and, in particular, evolution-based search are effective approaches to face this problem. Further, we show that the these search algorithms can be embedded in a training procedure, where harmful transformations are searched and used as data augmentation rules throughout training.
References


