Selective order data injection.

Does the order of training examples matter?

Abstract Idea: The idea is to present maximally different examples in rapid succession. In this way we hope to establish boundary conditions around the problem space. The intuition for why this works mathematically is….

Review of Other Associated Work:

Within the work of Yoshua Bengio we see the abstract notion that concepts ought to be introduced in order of increasing complexity. Regrettably, this work suggests a less high-level methodology is best. Here I propose that optimally, the examples ought to be fed in with respect to maximal difficulty as determined by an adversarial learner at each intermediate step.

Applications:

Running an adversarial learner each step of the training may prove to be impractical for most applications, however, apart from the obvious insights yielded into deep learning networks in general we can imagine a scenario where we have a set of samples determined at some selected step size. Often a per epoch/s basis likely provides a rough approximation of such. A general optimization for local optimizations of a such an algorithm can be computed with an L2 distance.

Please consult the following git repository for further info:

Implemented with TensorFlow v2.0. All experiments were performed on a Dell XPS 15 using an 8th generation i7 and a 1050ti m series and were computed using.

A grid search was completed across the top 5 performing adversarial learners on the ImageNet database.

While we do not see a performance in eventual learning convergence.

Explanation:

The inherent linearity of a higher dimensional space implies the definition of the problem space operates with superior accuracy when adversarial examples are deliberately injected into an intelligent system.

Let us establish an experiment whereby we have an extremely limited dataset and there exists a necessity to achieve performance on all example classes within the dataset which matches or exceeds at least 75%. There are many ways this can be accomplished, but an additional methodology can now be put forward in the selective training on adversarial demonstrated examples.

Furthermore, there exist scenarios where the performance in edge cases is the most important a model may encounter. This may include creating production systems which are resilient to

We can imagine production scenarios where it is highly important to establish high performance in edge cases, under such circumstances it is essential that approximate performance be established before ideal performance.

Future Work:

Future work in this domain should include operating on large scale datasets. Additionally, I would like to put forward the open question on how this dynamic affects deep double descent.

Effort may also be productively put forward on inquiring how such a network can be integrated into a large-scale simulation and integration of synthetic data. The principle notion here is the utilization of pair of networks each has access to some amount of good data, data which adheres to some performance accuracy defined to be adequate to fully justify the performance of the model.

This will be the topic of my next report.

Idealized network performance via synthetic approximation of data to some point.

If we inject adversarial examples into a network where these synthetic examples are the result of the generation of a generative adversarial network, we can create a model which only evaluates its performance based upon the few natural examples. In this manner we can constructively improve performance of the network even while the network expands.

1. We begin with natural adversarial cases
2. We loosen the adversarial grip on the network even as we increase the amount of synthetic allowable data.
3. The network being trained biases the level of insight of new training examples via their relevance on the new data. This allows the network to generate synthetic minds, which are graded based on the performance relative to the naturally occurring data. That is, gans, with evolutions. We need a way to quantify their performance. Augmented Gans with harsh self-grading.

ISI

Iron Sharpens Iron

References:

<http://www.machinelearning.org/archive/icml2009/papers/119.pdf> Curriculum Learning

<https://arxiv.org/pdf/1704.03453.pdf> The space of transferable adversarial examples.