AmiableNet: Learning to Teach with Good Natured Adversarial Examples

Eli Andrew*
Department of Computer Science
Stanford University
eyandrew@stanford.edu

1 Project Proposal

We propose to further AI teaching methods by training a teacher model to modify training examples such that a given image classifier model learns faster and or performs better. Most AI teaching methods have focused on training data selection and loss function creation. While our methods will also deal with training data, they will be focused on the modification of a given example rather than the selection of a particular example. This is an interesting problem because it aims to apply the approach of adversarial examples in a novel way, and looks to expand AI teaching methods to modifying training examples which would potentially help with creating better training data even from sub-par training datasets. Our dataset will be the CIFAR-10 dataset. We plan on potentially reducing the number of classes in the dataset to a number less than 10 and potentially as low as 1. This is because our goal is to improve training speed and accuracy of a classifier, and if multiple classes end up hindering that goal they will be removed. We propose using an existing model architecture for CIFAR-10, and re-training it with our teacher model. Our approach will be similar to the approach taken when generating adversarial examples that read loss information of the network and try and modify the input image to fool the network into mis-classifying it (ex. classify a cat as a dog). However, our approach will not be adversarial but rather amiable, as it will modify the input image such that the student model is more likely to correctly classify the input image. Our intention is that the teacher model will modify training data such that it slowly introduces complexity to the student model, which will hopefully lead to faster training and or better test results. Our first performance metric: test-set error at fixed training iteration intervals, is meant to measure how quickly the student model is able to learn. The second performance metric: final test-set error (with unbounded training iterations), is meant to measure any non-training time specific performance gains between the model with and without a teacher. Both performance metrics will be graphed and compared between the student model without a teacher model (same as it's old architecture) and with a teacher model, to see if there are any performance gains in test-set error.

References

[1] Yang Fen, Fei Tian, Tao Qin. Xiang-Yang Li, and Tie-Yan Liu. Learning to teach. In *International Conference on Learning Representations*, 2018.

[2] Lijun Wu, Fei Tian, Yingee Xia, Yang Fan, Tao Qin, Jianhuang Lai, Tie-Yan Liu. Learning to Teach with Dynamic Loss Functions. In *Neural Information Processing Systems*, 2018.

[3] Mengye Ren, Wenyuan Zeng, Bin Yang, Raquel Urtasun. Learning to Reweight Examples for Robust Deep Learning. In *International Conference on Machine Learning*, 2018.

*e.andrew@salesforce.com

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)