MLO Semester Project Proposal – Spring 2019

Understanding the adaptive training of neural nets across DL tasks and datasets

Tao LIN*

1 Introduction

There[2] has been tremendous progress in first-order optimization algorithms for training deep neural networks. One of the most dominant algorithms is Stochastic gradient descent (SGD) [RM51], which performs well across many applications in spite of its simplicity. However, SGD scales the gradient uniformly in all directions, which may lead to poor performance when the training data are sparse as well as limited training speed. Recent work has proposed a variety of adaptive methods that scale the gradient by square roots of some form of the average of the squared values of past gradients, e.g., Adagrad [DHS11], Adadelta [Zei12], RMSprop [TH12] and Adam [KB14].

Adam, in particular, has become the default algorithm leveraged across many deep learning frameworks due to its rapid training speed [WRS+17]. Despite their popularity, the generalization ability and out-of-sample behavior of these adaptive methods are likely worse than their non-adaptive counterparts. Adaptive methods often display faster progress in the initial portion of the training, but their performance quickly plateaus on the unseen data (development/test set) [WRS+17]. Indeed, the optimizer is chosen as SGD in several recent state-of-the-art works in natural language processing and computer vision [MKS17, LH16], wherein these instances SGD does perform better than adaptive methods.

In the meanwhile, the research community tries to analyze the convergence of a class of adaptive algorithms for non-convex optimization [CLSH18, ZTY+18, ZSJ+18, ZS18] or propose novel learning schemes [CYY+18, ZRS+18, CG18, Ano19]. This project aims to better study the theoretical and empirical properties of these adaptive algorithms.

2 Your Tasks

The routine of the project could follow:

1. (1 – 2 weeks) Try to understand different optimization methods, i.e., SGD with momentum, AdaGrad, Adadelta, RMSprop, Adam, and AmsGrad[3], as well as other recent proposed adaptive algorithms.

2. (3 – 4 weeks) Understanding the convergence of these algorithms, i.e., unifying the convergence of different algorithms.

*Machine Learning and Optimization Laboratory (MLO), EPFL

[2]have borrowed some sentences from [Ano19]

[3][LH17, RKK18] have recently proposed to address the problem with the Adam update rule.

3. (3 – 4 weeks) “Implement” these algorithms in PyTorch and benchmark them on top of ResNet and LSTM for some simple datasets.

4. (3 – 4 weeks) Try to empirically understand the property of these adaptive algorithms, e.g., the impact of large-batch training, the generalization gap.

3 Requirements

1. Experience with Machine Learning and Deep Learning,

2. Familiar with PyTorch,

3. Passionate about the topic.

References

[Ano19] Anonymous. Adaptive gradient methods with dynamic bound of learning rate. In *Submitted to International Conference on Learning Representations*, 2019. under review.

[CG18] Jinghui Chen and Quanquan Gu. Closing the generalization gap of adaptive gradient methods in training deep neural networks. *arXiv preprint arXiv:1806.06763*, 2018.

[CLSH18] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. *arXiv preprint arXiv:1808.02941*, 2018.

[CYY+18] Zaiyi Chen, Tianbao Yang, Jinfeng Yi, Bowen Zhou, and Enhong Chen. Universal stagewise learning for non-convex problems with convergence on averaged solutions. *arXiv preprint arXiv:1808.06296*, 2018.

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[KB14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[LH16] Ilya Loshchilov and Frank Hutter. Sgdr: stochastic gradient descent with restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[LH17] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 2017.

[MKS17] Stephen Merity, Nitish Shirish Keskar, and Richard Socher. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*, 2017.

[RKK18] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. 2018.

[RM51] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[TH12] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[WRS+17] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nathan Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. *arXiv preprint arXiv:1705.08292*, 2017.

[Zei12] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[ZRS+18] Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 9814–9824, 2018.

[ZS18] Fangyu Zou and Li Shen. On the convergence of adagrad with momentum for training deep neural networks. *arXiv preprint arXiv:1808.03408*, 2018.

[ZSJ+18] Fangyu Zou, Li Shen, Zequn Jie, Weizhong Zhang, and Wei Liu. A sufficient condition for convergences of adam and rmsprop. *arXiv preprint arXiv:1811.09358*, 2018.

[ZTY+18] Dongruo Zhou, Yiqi Tang, Ziyan Yang, Yuan Cao, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.