

進捗報告

1 今週やったこと

- FastAutoAugment 動作確認
- TDGA の理解, テンプレ準備

2 FastAutoAugment 動作確認

Github の FastAutoAugment の PyTorch 実装 [1] の Docker 環境での動作確認をした。だいたいの実装を追えたので来週以降は AutoAugment のベイズ最適化の部分を GA(SGA) に差し替えることをまずは試す。表 1 にレポの人が上げてくれていたシードで回して精度確認ができた validation accuracy (Acc @1) の遷移を示す。赤線が AutoAugment なしのベースラインを表す。また、学習時間は子モデルの学習に 1 時間弱かかるので 5 分割学習 + 最終学習でだいたい GPU 時間で 5 時間程度かかる。

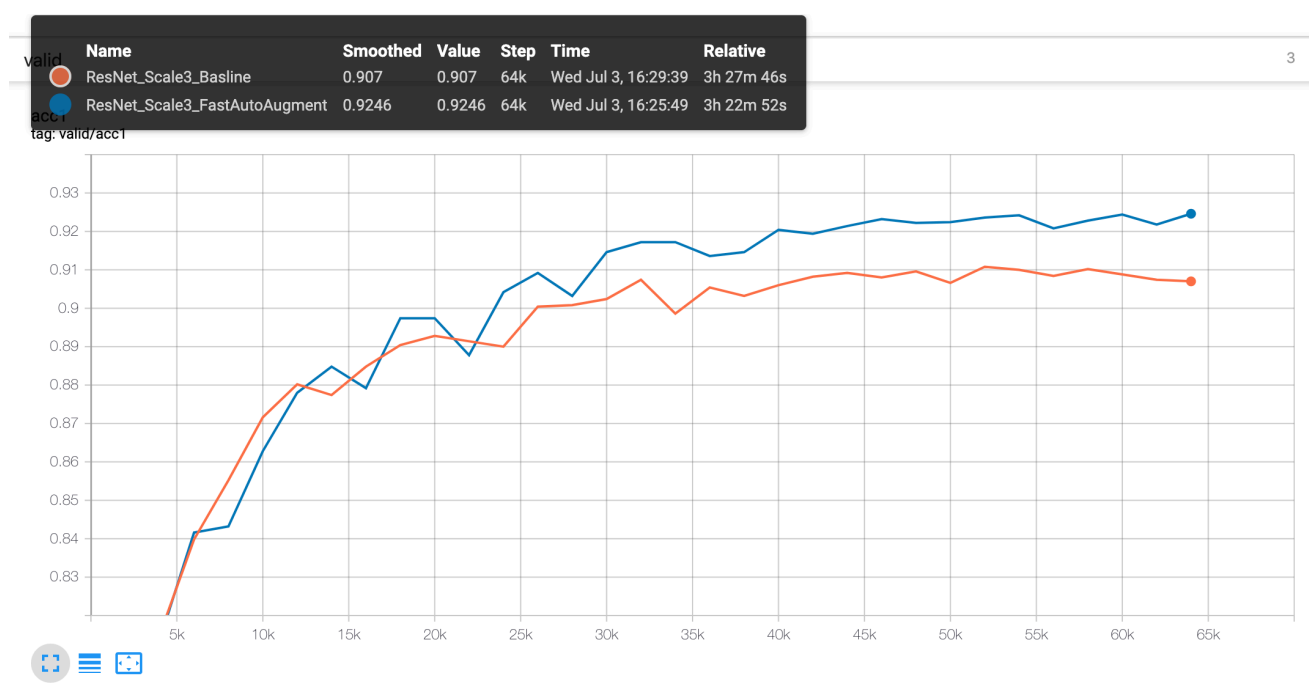


図 1: FastAutoAugment Acc@1

3 TDGA

<https://github.com/1g-hub/TDGA> にテンプレートレポジトリを作っておきました。最低限 pip で依存関係解消してインストールできるようにしたいと思ってます。ベースを deap でその中の selection 操作の 1 つとして拡張する形で作ろうと構想しているので、ほとんど書き方は deap のままで大丈夫になるはず。

4 来週のタスク

JSAL, TDGA 実装をすすめる .

参考文献

- [1] JunYeopLee/fast-autoaugment-efficientnet-pytorch. [https://github.com/JunYeopLee/
fast-autoaugment-efficientnet-pytorch.](https://github.com/JunYeopLee/fast-autoaugment-efficientnet-pytorch)