Gradients as Features for Deep Representation Learning

Fangzhou Mu, Yingyu Liang, Yin Li

University of Wisconsin-Madison
ICLR 2020 (poster)

Code repository
https://github.com/fmu2/gradfeat20
Project webpage
http://pages.cs.wisc.edu/~fmu/gradfeat20
Representation Learning

Unsupervised Learning

BiGAN (Dumolin et al., 2016; Donahue et al., 2016)
VAE (Kingma & Welling, 2014)
...

Self-supervised learning

Jigsaw (Noroozi & Favaro, 2016)
RotNet (Gidaris et al., 2018)
...

Transfer learning

ImageNet (Deng et al., 2009)
...

\[ f_\theta : \mathbb{R}^D \rightarrow \mathbb{R}^d \]
Phase 1: Learning Representations

\[ f_\theta : \mathbb{R}^D \to \mathbb{R}^d \]
Phase 1: Learning Representations

Phase 2: Learning Linear Classifier (Standard approach)
Phase 1: Learning Representations

Phase 2: Learning Linear Classifier (Proposed approach)

Logits = Linear weight × Activation + Gradient feature (w.r.t. parameters) × Linear weight
Our model subsumes the standard logistic classifier.

Logits = Linear weight \times Activation + Gradient feature (w.r.t. parameters) \times Linear weight
Our model provides a local linear approximation to fine-tuning.

**Key insight**: Wide neural networks evolve as linear models under gradient descent. (*Lee et al.*, NeurIPS 2019)

*(More details in Section 3.2 of our paper)*
Our model is fast at training and inference time.

**Key insight:** Embed the evaluation of Jacobian-vector product in forward pass.

*(More details in Section 3.3 of our paper)*
Results

![Graph showing classification accuracy for BiGAN (CIFAR-10), Jigsaw (VOC07), and ImageNet (VOC07) datasets with baseline, proposed, and fine-tuning methods.](More details in Section 4 of our paper)
Summary

• A novel linear model that leverages both network activation and per-sample parameter gradients as features for representation learning.

• An interpretation of our method as linear approximation of fine-tuning.

• A scalable algorithm for training and inference of our method.

• Strong results of our method in various settings.

Thank you!

Code repository
https://github.com/fmu2/gradfeat20
Project webpage
http://pages.cs.wisc.edu/~fmu/gradfeat20