# Lookahead Optimizer: k steps forward, 1 step back

Michael R. Zhang, James Lucas, Geoffrey Hinton, Jimmy Ba

UNIVERSITY OF TORONTO

VECTOR INSTITUTE

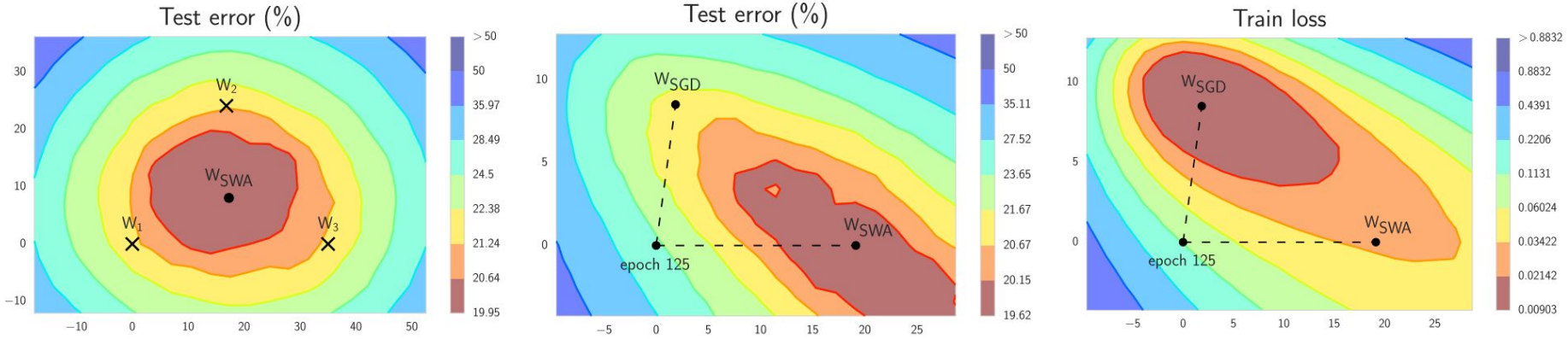
# Related Work / Motivation

# Polyak Averaging

- Proposed by Boris Polyak as a method for acceleration in convex optimization in 1992. Ruppert independently explored this in 1988.
- Taking arithmetic average of weights gives faster convergence in convex optimization
- Weight averaging in neural networks has seen more interest recently

# Stochastic Weight Averaging (2018)

- Create an ensemble by averaging the *weights* of a neural network at different points in training
- Beats existing methods for ensembling in model space




Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, Andrew Gordon Wilson

# Regularized Nonlinear Acceleration (RNA)

- A related, more complicated algorithm that tries to find a point where the gradient is zero.
- It solves a linear system based on the most recent k iterates
- Achieves faster convergence and occasionally better generalization
- Factor of k times more memory and more compute
  - $O(K^2d + K^3)$

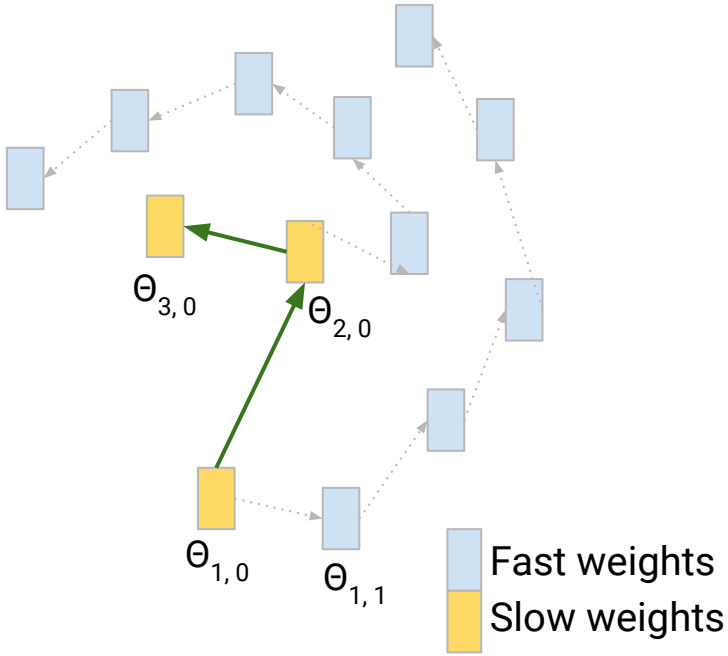
Damien Scieur, Edouard Oyallon, Alexandre d'Aspremont, Francis Bach

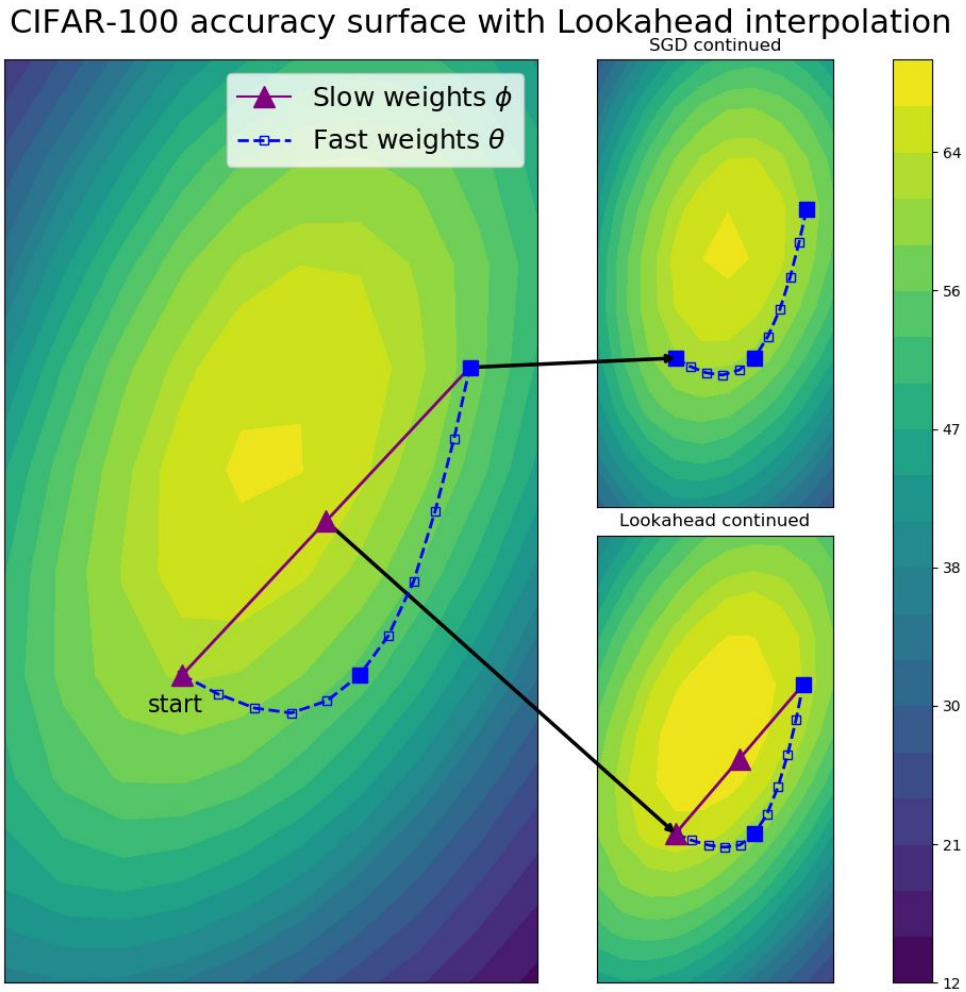
# Method

# Lookahead Optimizer

**Algorithm 1** Lookahead Optimizer:

**Require:** Initial parameters $\phi_0$, objective function $L$
**Require:** Synchronization period $k$, slow weights step size $\alpha$, optimizer $A$
**for** $t = 1, 2, \ldots$ **do**
  Synchronize parameters $\theta_{t,0} \leftarrow \phi_{t-1}$
  **for** $i = 1, 2, \ldots, k$ **do**
    sample minibatch of data $d \sim \mathcal{D}$
    $\theta_{t,i} \leftarrow \theta_{t,i-1} + A(L, \theta_{t,i-1}, d)$
  **end for**
  Perform outer update $\phi_t \leftarrow \phi_{t-1} + \alpha(\theta_{t,k} - \phi_{t-1})$
**end for**
**return** parameters $\phi$






- Project parameters of neural network into 2-D for visualization
- Lighter colors represent regions of higher accuracy

# Noisy Quadratic Analysis

- Simple proxy for neural network optimization (see work from James Martens, Roger Grosse, Tony Wu, Guodong Zhang et al.)

$$L(\theta) = \frac{1}{2}\theta^T \mathbf{A}\theta = \frac{1}{2}\sum_{i=1}^{d} a_i \theta_i^2 \triangleq \sum_{i=1}^{d} l(\theta_i).$$

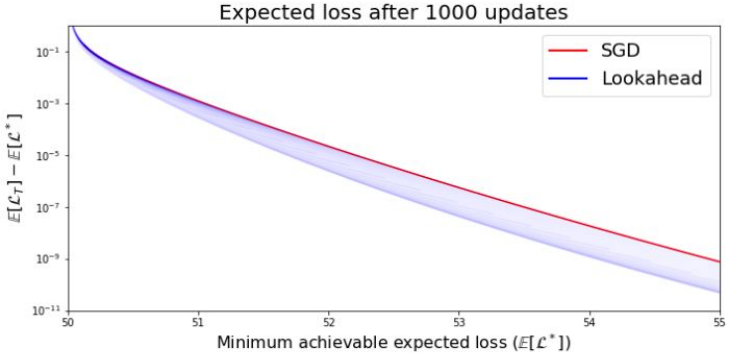
We assume that the gradient we obtain is noisy: for each dimension, instead of receive the true gradient $a_i\theta_i$, we get a noisy version $a_i\theta_i + c_i$, where $c_i \sim \mathcal{N}(0, \sigma_i^2)$.

# Noisy Quadratic Analysis

**Proposition 2** (Lookahead variance reduction). *Let $0 < \gamma < 2/L$ be the learning rate of SGD and Lookahead where $L = \max_i a_i$. In the noisy quadratic model, the iterates of SGD and Lookahead with SGD as its inner optimizer converge to $0$ in expectation and the variances converge to the following fixed points:*

$$V^*_{SGD} = \frac{\gamma^2 \mathbf{A}^2 \Sigma^2}{\mathbf{I} - (\mathbf{I} - \gamma \mathbf{A})^2} \tag{6}$$

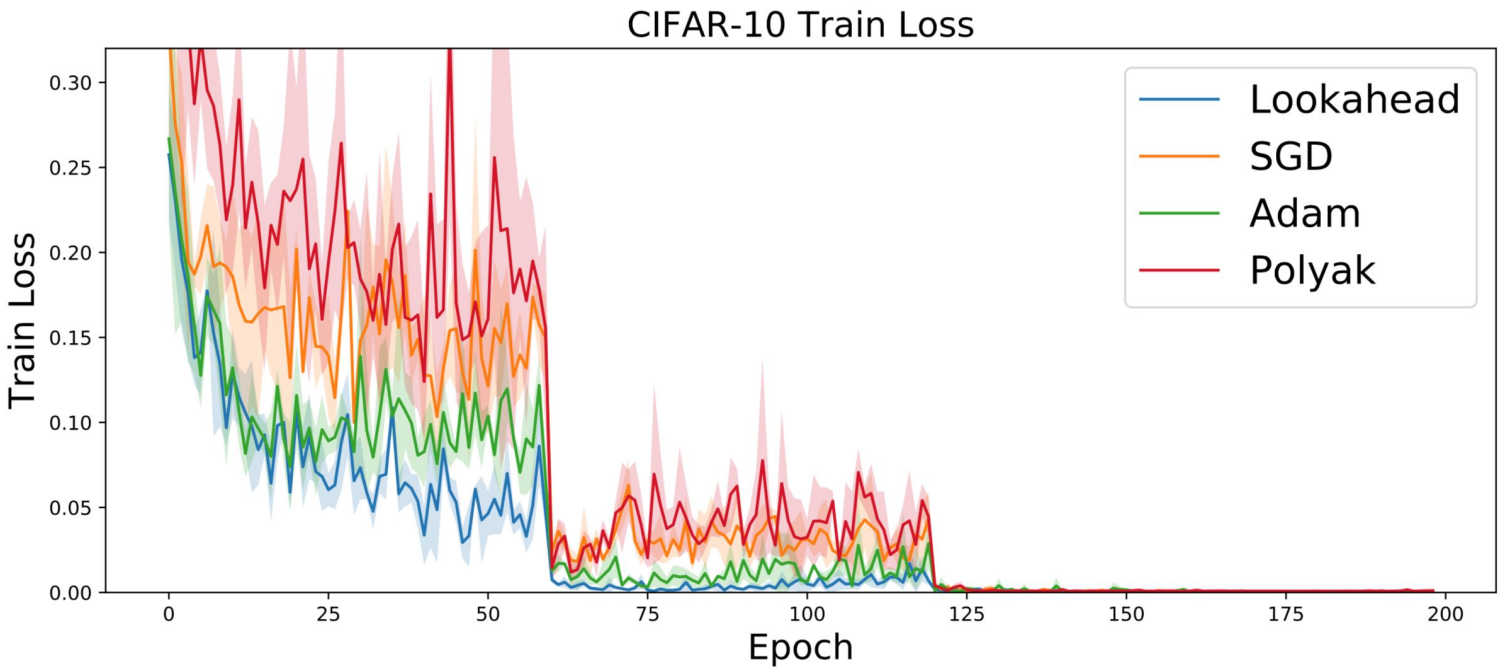
$$V^*_{LA} = \frac{\alpha^2(\mathbf{I} - (\mathbf{I} - \gamma \mathbf{A})^{2k})}{\alpha^2(\mathbf{I} - (\mathbf{I} - \gamma \mathbf{A})^{2k}) + 2\alpha(1-\alpha)(\mathbf{I} - (\mathbf{I} - \gamma \mathbf{A})^{k})} V^*_{SGD} \tag{7}$$



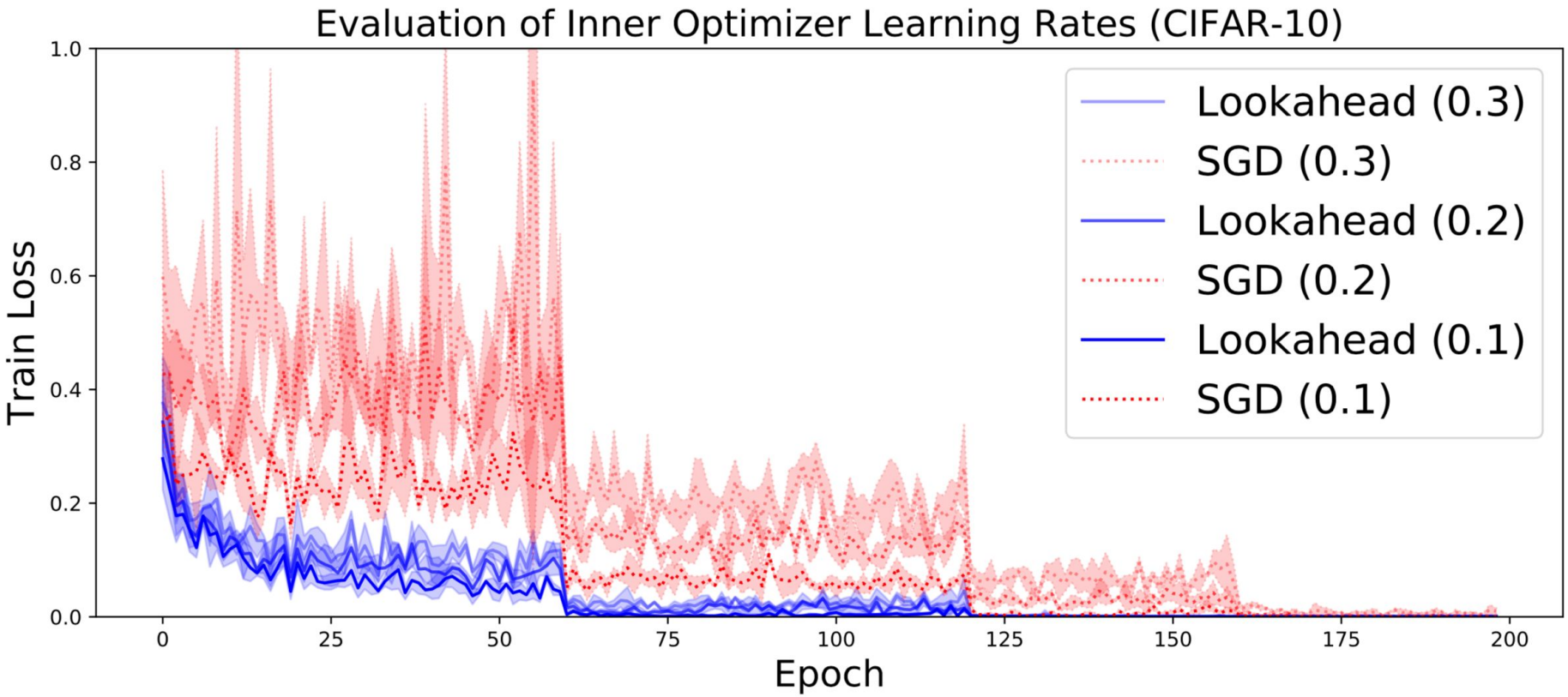
# Results

# CIFAR-10

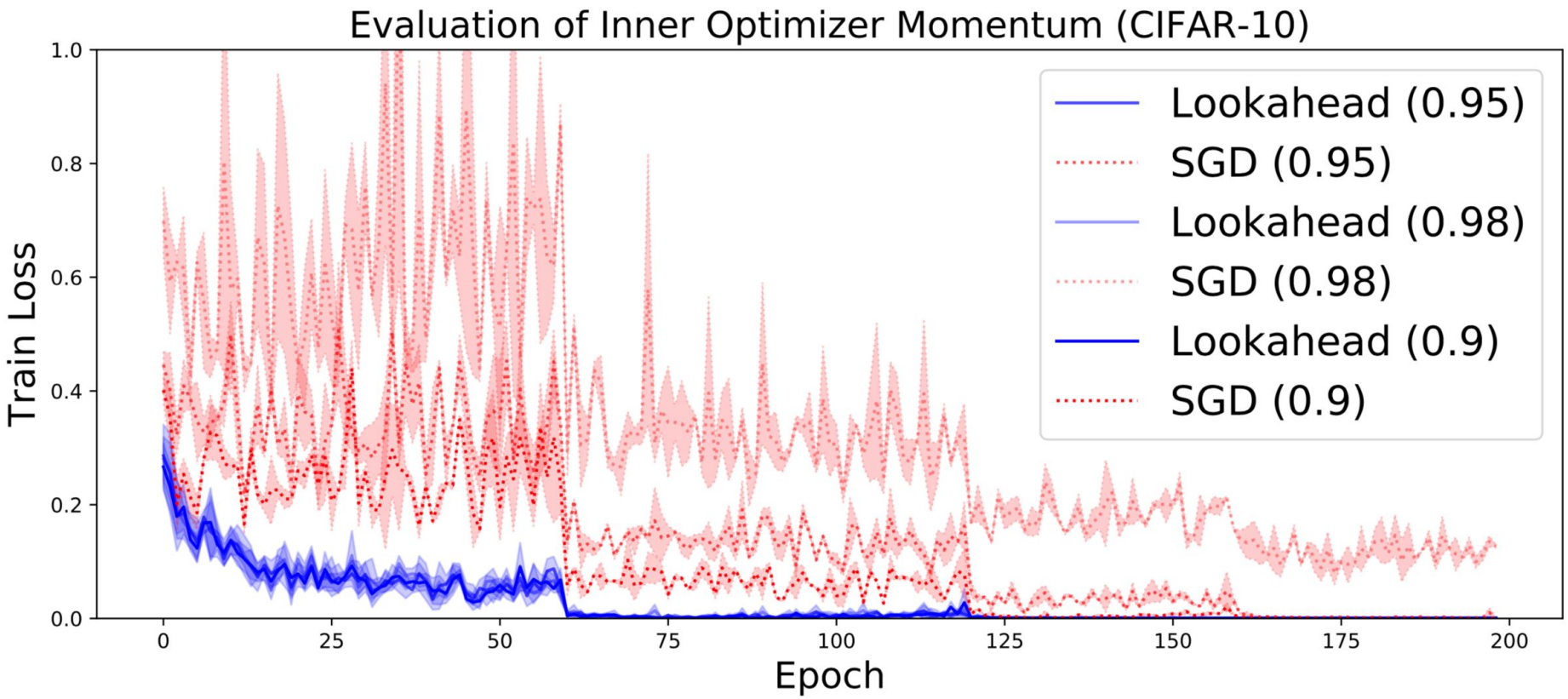
- Find best hyperparameters for inner optimizer, then perform small grid search on outer loop



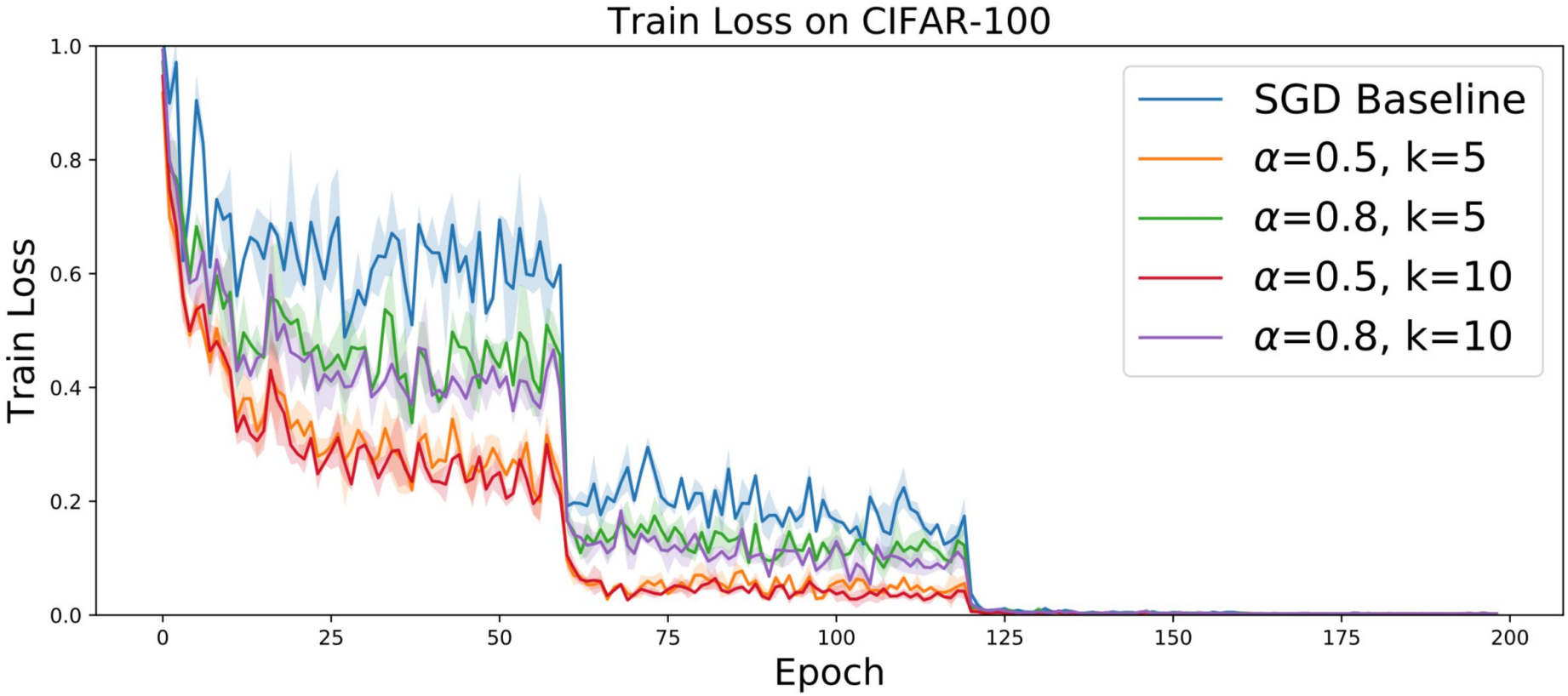
# Hyperparameter Robustness



# Hyperparameter Robustness

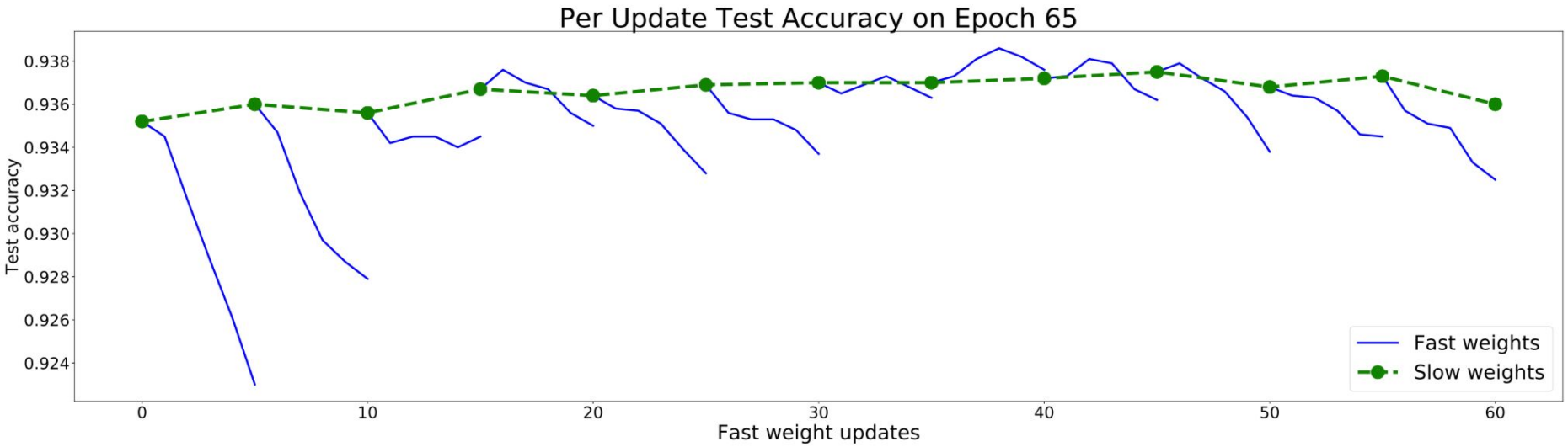


# Hyperparameter Robustness (LA hyperparams)



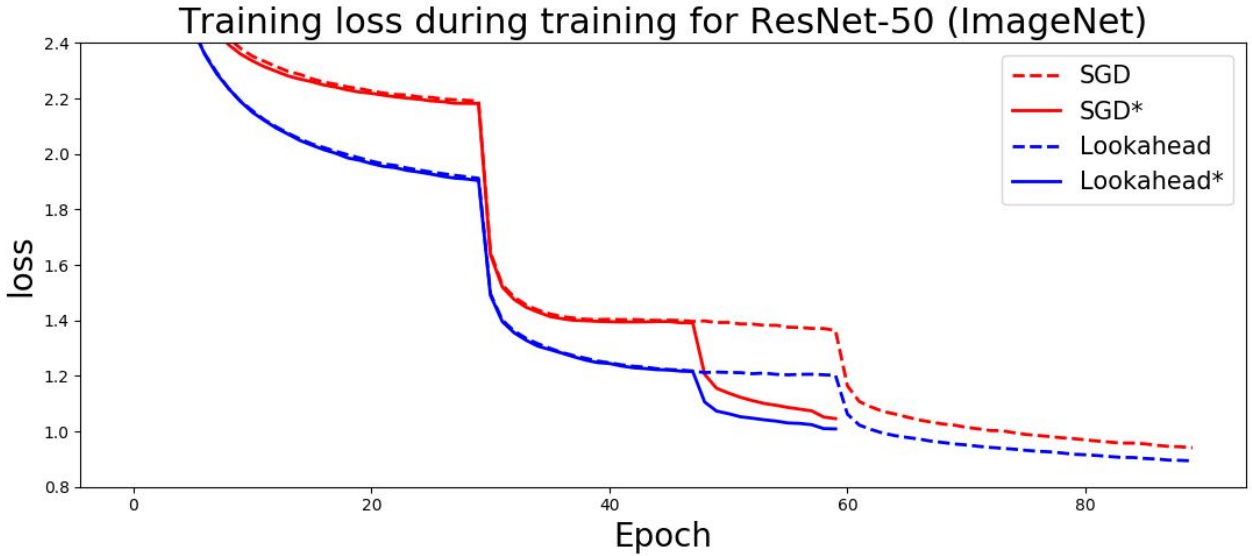
# Fast / Slow Weight Intuition

- Inner updates can degrade performance on both training and test set, while outer update restores performance



# ImageNet

- Standard benchmark for image classification: over 1.28 million training images and 50,000 validation images
- ResNet-50 has 25 million parameters, works for ResNet-152 as well



# Test time performance

| Optimizer | CIFAR-10 | CIFAR-100 |
|---|---|---|
| SGD | $95.23 \pm .19$ | $78.24 \pm .18$ |
| Polyak | $95.26 \pm .04$ | $77.99 \pm .42$ |
| Adam | $94.84 \pm .16$ | $76.88 \pm .39$ |
| Lookahead | $95.27 \pm .06$ | $78.34 \pm .05$ |

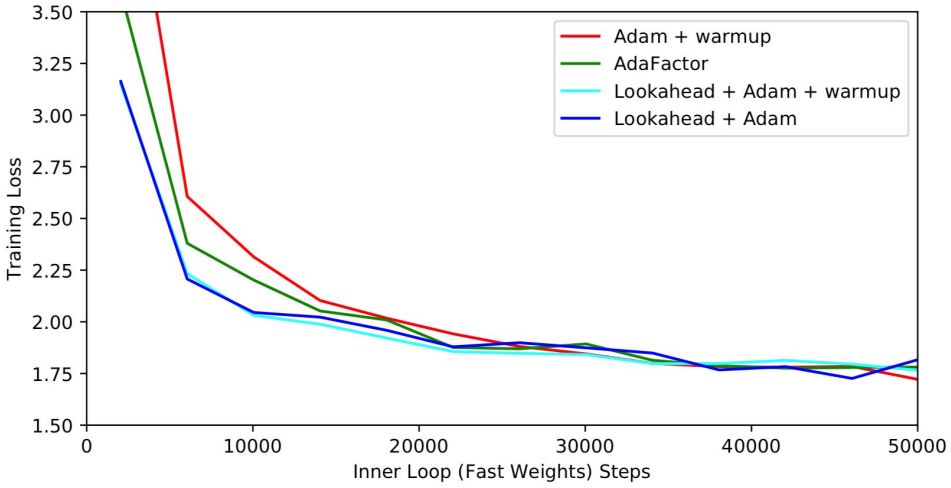
Table 1: CIFAR Final Validation Accuracy.

| Optimizer | LA | SGD |
|---|---|---|
| Epoch 50 - Top 1 | 75.13 | 74.43 |
| Epoch 50 - Top 5 | 92.22 | 92.15 |
| Epoch 60 - Top 1 | 75.49 | 75.15 |
| Epoch 60 - Top 5 | 92.53 | 92.56 |

Table 2: Top-1 and Top-5 single crop validation accuracies on ImageNet.

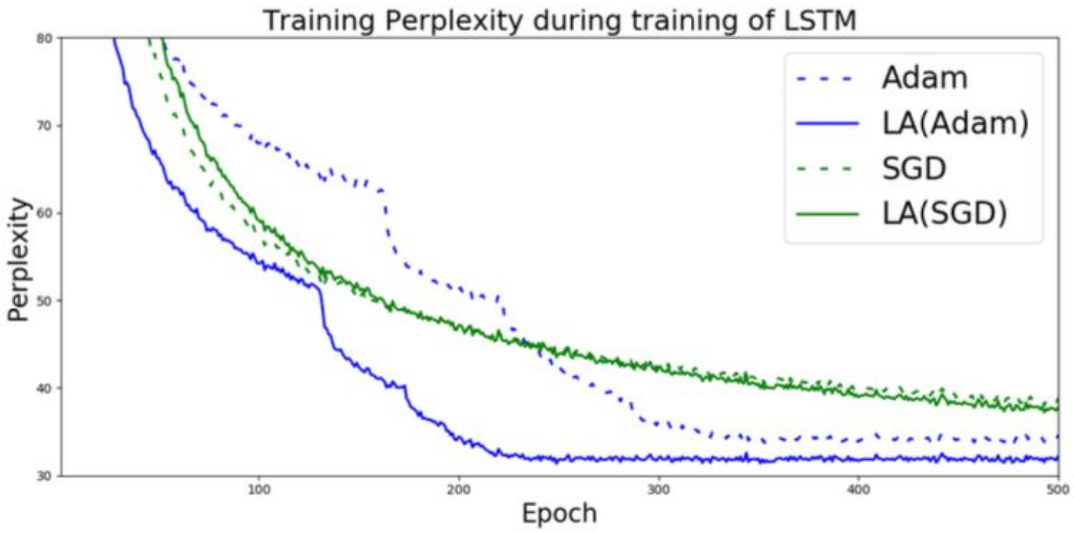
# Neural Machine Translation

- WMT14 English-to-German task with Transformer Model
- Lookahead is *robust*: achieve faster training convergence without using tuned, ramp-up learning rate schedules



# Penn Tree Bank

- Benchmark to model prediction of next word given previous words




| Optimizer | Train | Val. | Test |
|---|---|---|---|
| SGD | 43.62 | 66.0 | 63.90 |
| LA(SGD) | 35.02 | 65.10 | 63.04 |
| Adam | 33.54 | 61.64 | 59.33 |
| LA(Adam) | **31.92** | **60.28** | **57.72** |
| ASGD | - | 61.18 | 58.79 |

# Summary

- Faster convergence with little hyperparameter tuning on a variety of datasets and models, big and small
- Fast weights can degrade test and training accuracy, but slow weights restore performance
- Extensions: learning rate scheduling, family of methods that maintain memory information

# How to Use

Simple interface, in TensorFlow and PyTorch:

```
optimizer = # {any optimizer} e.g. tf.train.AdamOptimizer
if args.lookahead:
    optimizer = Lookahead(optimizer, la_steps=args.la_steps,
la_alpha=args.la_alpha)
```

Code: https://github.com/michaelrzhang/lookahead

Contact: michael@cs.toronto.edu

# Thank you!

UNIVERSITY OF TORONTO

VECTOR INSTITUTE

# Interaction with SWA