RADemics

Optimization Algorithms for Efficient Training of Deep Predictive Models

V.Padma, Dr. Sridhar, Mr. Deepak Sharma
SRM UNIVERSITY, S-VYASA DEEMED TO BE UNIVERSITY,
IFTM UNIVERSITY

Optimization Algorithms for Efficient Training of Deep Predictive Models

[1]V. Padma, Assistant Professor, Department of Mathematics, SRM University, Kattankulathur, Chennai-603203. padmagurumoorthy@gmail.com

[2]Sridhar, Professor and Director Academics, Department of School of Computer Science and Applications, S-VYASA Deemed to be University, Bengaluru - 560059. drsridhar@svyasa.edu.in

[3]Deepak Sharma, Assistant Professor, Department of Mathematics, School of Sciences, IFTM University, Lodhipur Rajput Moradabad, Uttar Pradesh -244102. deepak.sharma@iftmuniversity.ac.in

Abstract

The growing complexity of deep learning applications has intensified the need for optimization algorithms that accelerate model convergence, improve training stability, and enhance predictive performance. This book chapter, Optimization Algorithms for Efficient Training of Deep Predictive Models, presents a comprehensive examination of mathematical optimization principles and advanced training strategies for neural networks. The discussion covers gradient descent algorithms, momentum-based optimization, adaptive learning rate methods, second-order optimization, Hessian-based techniques, and validation-driven learning rate scheduling. Particular attention is directed toward convergence behavior, gradient stability, hyperparameter selection, regularization, and computational efficiency. Advanced strategies, including Adam, AdamW, Nesterov Accelerated Gradient, Bayesian hyperparameter optimization, and mixed-precision training, are examined in relation to their contributions to efficient model development. The chapter also addresses challenges associated with non-convex loss landscapes, vanishing and exploding gradients, overfitting, and resource-intensive training procedures. Comparative evaluation through convergence analysis, validation performance, computational cost, and generalization assessment provides a foundation for selecting suitable optimization strategies across diverse deep learning architectures. By integrating theoretical principles with practical training considerations, this chapter establishes a structured framework for developing accurate, computationally efficient, and robust predictive models across contemporary artificial intelligence applications.

Keywords: Deep Learning Optimization, Gradient Descent, Adaptive Learning Rate, Convergence Acceleration, Hyperparameter Optimization, Training Efficiency

Introduction

Optimization algorithms play a crucial role in improving the training efficiency, convergence speed, and predictive performance of deep learning models [1]. Deep predictive architectures frequently contain millions of trainable parameters, requiring effective optimization strategies to minimize training errors and identify suitable parameter configurations [2]. Gradient-based

methods enable iterative parameter updates by estimating the direction of error reduction during training [3]. Selecting an appropriate optimization algorithm depends on model complexity, dataset characteristics, computational resources, and learning objectives [4]. Efficient optimization reduces unnecessary computational expenditure while supporting accurate and stable model development across diverse predictive applications [5].

Stochastic Gradient Descent (SGD) remains a widely used optimization technique because of its computational simplicity and effectiveness in training large neural networks [6]. The algorithm updates model parameters using gradients calculated from individual samples or small batches of training observations [7]. Momentum-based SGD incorporates information from previous updates to accelerate convergence and reduce oscillations during optimization [8]. Adaptive learning rate methods, including Adam, RMSprop, and Adagrad, adjust parameter updates according to gradient behaviour and historical information [9]. These techniques offer practical advantages when training deep architectures with complex loss surfaces, sparse features, and varying gradient magnitudes [10].

Learning rate scheduling and hyperparameter optimization significantly influence the effectiveness of deep predictive model training [11]. A learning rate that remains excessively high can cause unstable updates, while an excessively low value can slow convergence and increase training time [12]. Strategies such as step decay, cosine annealing, and warm-up scheduling help regulate parameter updates throughout training [13]. Hyperparameters, including batch size, momentum, regularization strength, and network depth, also affect optimization behaviour. Grid search, random search, and Bayesian optimization provide systematic approaches for identifying suitable configurations [14]. Careful tuning improves convergence stability, generalization performance, and computational efficiency across different learning tasks [15].