

Chapter 3

Optimizing Hyper Meta Learning Models: An Epic

G. Devika

 <https://orcid.org/0000-0002-2509-2867>

Government Engineering College, Krishnarajapete, India

Asha gowda Karegowda

 <https://orcid.org/0000-0002-1353-4293>

Siddaganga Institute of Technology, India

ABSTRACT

Optimizing hyper meta learning models is a critical task in the field of machine learning, as it can improve the performance, efficiency, and scalability of these models. In this chapter, the authors present an epic overview of the process of optimizing hyper meta learning models. They discuss the key steps involved in this process, including task selection, model architecture selection, hyperparameter optimization, model training, model evaluation, and deployment. They also explore the benefits of hyper meta learning models and their potential future applications in various fields. Finally, they highlight the challenges and limitations of hyper meta learning models and suggest future research directions to overcome these challenges and improve the effectiveness of these models.

INTRODUCTION

Optimizing meta-learning models involves tuning the hyper parameters of the model to improve its performance on a given task. Meta-learning is a subfield of machine

DOI: 10.4018/978-1-6684-7659-8.ch003

learning that focuses on developing algorithms that can learn from experience to solve new problems quickly and efficiently (Chelsea et al, 2022). Meta-learning models are typically trained on a set of related tasks, and then used to quickly adapt to new tasks by leveraging the knowledge gained from the training tasks.

To optimize meta-learning models, it is important to select appropriate hyperparameters for the model, such as the learning rate, regularization strength, and network architecture. There are several techniques that can be used for hyperparameter optimization, including grid search, random search, Bayesian optimization, and gradient-based optimization (Yikai et al, 2019).

Grid search involves defining a set of possible values for each hyperparameter and testing all possible combinations of these values to find the combination that yields the best performance on a validation set. Random search involves randomly sampling hyperparameters from their defined distributions and evaluating their performance on a validation set (Adam et al, 2022). Bayesian optimization is a more advanced technique that uses Bayesian inference to build a probabilistic model of the objective function (i.e., the model performance) and select hyperparameters that are likely to yield the best performance. Gradient-based optimization involves optimizing the hyperparameters using gradient descent, where the gradient of the objective function with respect to the hyperparameters is computed and used to update the hyperparameters. In addition to hyperparameter optimization, other techniques can be used to improve the performance of meta-learning models, such as data augmentation, model ensembling, and transfer learning. By optimizing meta-learning models, it is possible to develop algorithms that can quickly adapt to new tasks and improve their performance over time.

Solving a machine learning use case involves a number of steps, including understanding the problem, collecting and preparing the data, selecting an appropriate machine learning model, training and evaluating the model, and deploying the model in a production environment. Here is a high-level overview of the process:

- a. **Problem Definition:** The first step in solving a machine learning use case is to define the problem you are trying to solve. This involves understanding the business objectives, the available data, and the potential impact of the solution.
- b. **Data Collection and Preparation:** Once the problem is defined, the next step is to collect and prepare the data. This involves identifying the relevant data sources, cleaning and pre-processing the data, and selecting appropriate features.
- c. **Model Selection:** After the data is prepared, the next step is to select an appropriate machine learning model. This depends on the type of problem and the characteristics of the data. Common types of machine learning models include supervised learning, unsupervised learning, and reinforcement learning.

31 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/optimizing-hyper-meta-learning-models/331477

Related Content

Smart Blockchain: A Formidable Combination of Analytics and Blockchain – A Survey

Gutha Jaya Krishna, Vadlamani Ravi, Bheemidi Vikram Reddy and Mohammad Zaheeruddin (2020). *Smart Systems Design, Applications, and Challenges* (pp. 43-62).

www.irma-international.org/chapter/smart-blockchain/249108

Ant Miner: A Hybrid Pittsburgh Style Classification Rule Mining Algorithm

Bijaya Kumar Nanda and Satchidananda Dehuri (2020). *International Journal of Artificial Intelligence and Machine Learning* (pp. 45-59).

www.irma-international.org/article/ant-miner/249252

An Overview and Analysis of Machine Learning Classification Algorithms in Healthcare

Soumitra Saha (2024). *Applying Machine Learning Techniques to Bioinformatics: Few-Shot and Zero-Shot Methods* (pp. 39-61).

www.irma-international.org/chapter/an-overview-and-analysis-of-machine-learning-classification-algorithms-in-healthcare/342717

Sentiment Analysis on Social Media: Recent Trends in Machine Learning

Ramesh S. Wadawadagi and Veerappa B. Pagi (2020). *Handbook of Research on Emerging Trends and Applications of Machine Learning* (pp. 508-527).

www.irma-international.org/chapter/sentiment-analysis-on-social-media/247579

Three-Layer Stacked Generalization Architecture With Simulated Annealing for Optimum Results in Data Mining

K. T. Sanvitha Kasthuriarachchi and Sidath R. Liyanage (2021). *International Journal of Artificial Intelligence and Machine Learning* (pp. 1-27).

www.irma-international.org/article/three-layer-stacked-generalization-architecture-with-simulated-annealing-for-optimum-results-in-data-mining/279277