Aleksander Lempinen

MLOps approach for application specific performance tuning for machine learning systems

Master's Thesis in Information Technology

August 5, 2023

University of Jyväskylä

Faculty of Information Technology

Author: Aleksander Lempinen

Contact information: `aleksander.lempinen@gmail.com`

Supervisor: Tommi Mikkonen

Title: MLOps approach for application specific performance tuning for machine learning systems

Työn nimi: TODO samma på finska

Project: Master's Thesis

Study line: Educational Technology

Page count: 25+0

Abstract: TODO abstract

Keywords: LaTeX, gradu3, Master's Theses, Bachelor's Theses, user's guide

Suomenkielinen tiivistelmä: TODO tiivistelmä suomeksi

Avainsanat: MLOPS, TODO

Glossary

ML	Machine Learning
MLOps	Machine Learning Operations
TODO	TODO

Contents

1 Introduction

Machine learning (ML) and Artificial Intelligence (AI) have been a hot topic of discussion in the past decade. While there is a mountain of academic research on ML methods and tools, there is a lack of attention paid to practical real-world challenges encountered when developing or running ML systems. DevOps has previously addressed similar challenges in software engineering and a field of MLOps which is DevOps applied to ML has emerged. Machine learning operations (MLOps) focuses on solving challenges related to operating real world machine learning systems (Kreuzberger, Kühl, and Hirschl 2023).

Machine learning (ML) systems are widely adopted and many organizations successfully have ML models running in production. Examples of machine learning systems in different fields include recommender systems , targeted ads (Domingos October 1, 2012), drug design (Domingos October 1, 2012) or search engines (Domingos October 1, 2012).

Find ref

Find ref

Most recent breakthroughs that have generated media attention have been in the fields of computer vision in the form of latent diffusion models (LDM) (Rombach et al. April 13, 2022) such as Stable Diffusion (Stability AI August 22, 2022) for generating images from prompts and natural language processing in the form of large language models (LLM) (Touvron et al. February 27, 2023) such as ChatGPT (OpenAI November 30, 2022). There have also been great developments in tooling for machine learning such as Tensorflow, Pytorch or scikit-learn for model development, Ray, Horovod or DeepSpeed for distributed training and MLFlow or Tensorboard for machine learning monitoring.

TF, Pytorch, scikit-learn refs

Ray, Horovod and DeepSpeed refs

MLFlow, Tensorboard refs

Despite wide adoption and many successes, there are still challenges with machine learning systems in practice . The required amount of computation for machine learning has been on the rise. According to an OpenAI technical blog the trend is exponential and more compute leads to better performance (Amodei and Hernandez May 16, 2018). Increased compute requirements also mean increased costs such as financial, operational or environmental. Strubell et al. (April 3, 2020) in their extended abstract bring attention to the environmental impact of training models and in particular hyperparameter tuning, during which costs of training many relatively inexpensive models quickly adds up.

Find ref, state of MLOps?

In addition to cost there may be other requirements for machine learning systems. For example machine learning systems on the edge might encounter system requirements such as latency and energy use or have limited resources such as memory or compute (Chen and Ran August 2019). Ways of meeting these requirements include hyperparameter tuning, reducing the amount of parameters in the model or model compression such as knowledge distillation (Chen and Ran August 2019).

Early stopping has been used as a cost optimization technique to reduce training time by stopping training when performance of the model stops improving on the validation set (Prechelt June 1, 1998). More recent work on larger models shows that models might still improve later if training continues for a longer time (Hoffer, Hubara, and Soudry January 1, 2018). Using early stopping with other performance metrics such as system metrics has not been as thoroughly studied.

The aim of this thesis is to investigate whether using early stopping with system metrics leads to more efficient hyperparameter tuning when there are resource constraints. Investigation is limited to a small set of widely available machine learning algorithms and datasets that do not require a lot of computation. While more complex and effective hyperparameter optimization methods exist only the simplest are used to simplify the experiments for clarity.

The theoretical significance of the thesis is to show that traditional hyperparameter optimization techniques can not only be used on machine learning performance metrics but also alternative metrics such as system metrics. The practical outcomes are reducing costs and allowing for quickly and efficiently tailoring models to fit specific system metric constrains.

This thesis is structured in the following manner: Chapter 1 provides an introduction and context for the thesis. Chapter 2 contains background information about machine learning, DevOps and MLOps and how they relate to each other. Chapter 3 describes the performed experiments and their methods and design including research questions, datasets and algorithms used. Chapter 4 presents the results of the experiments. Chapter 5 revisits the research questions and discusses the interpretation of the results, limitations, related work and future work. Chapter 6 concludes the thesis by summarizing key findings.

2 Machine Learning Operations

Add outline of how ML, DevOps and MLOps fit together

2.1 Machine Learning

2.1.1 Overview

Intro to ML

Writing programs and developing algorithms to complete specific tasks is a labor intensive task requiring professional programming expertise. A different approach is to develop generic algorithms that can change behavior by learning. The field studying these types of algorithms is called machine learning. Machine learning algorithms learn by applying an optimization algorithm to adjust set of parameters called a model and this process is called training the model (LeCun, Bengio, and Hinton May 2015). A simplified machine learning workflow consists of splitting the data into training and test datasets, performing preprocessing separately for each dataset, training the model on the training dataset and finally evaluating the trained model on the test dataset. ref

Machine learning is widely used in applications like search, drug design or ad placement and can be also known as data mining or predictive analytics (Domingos October 1, 2012). Developing machine learning systems, which are systems that are based on machine learning, can be a difficult task. Unlike traditional software development, experiments with both code and data as inputs are central to machine learning development (Zaharia et al. 2018) and reproducibility of the experiments is often problematic. While plenty of research focuses on machine learning methods or even datasets and data quality, the biggest bottleneck is human cycles (Domingos October 1, 2012). Faster iterations improve the machine learning developer or researcher experience. An important metric to pay attention to and optimize is the mean iteration cycle for machine learning developers.

Types of ML

Machine learning can be practiced with two different goals in mind. First is explanatory modeling with the purpose of scientific theory building and testing and the second is predictive modeling mostly used outside of scientific research (Shmueli August 1, 2010). One practical difference is that unlike predictive modeling, explanatory modeling rarely uses holdout test sets or cross validation for evaluation (Shmueli August 1, 2010). Lack or presence of evaluation on a test set can be used as a heuristic to quickly determine whether a machine learning project is explanatory or predictive in nature. However, even explanatory modeling benefits from evaluating the predictive power (Shmueli August 1, 2010). Domingos (October 1, 2012) in their paper assume all machine learning is predictive in nature and state the following:

> The fundamental goal of machine learning is to generalize beyond the examples in the training set.

It is important to keep in mind the end goals of a machine learning project, because common practices in a research setting might not be applicable when creating machine learning systems.

Machine learning algorithms can be categorized as supervised learning, unsupervised learning or reinforcement learning. The main differences are related to whether the model learns by using "right answers" provided by labeled data in supervised learning, by finding structure in the dataset in unsupervised learning or by interacting with the world in reinforcement learning. Model evaluation is also different with supervised learning mostly relying on universal cross-validation on previously unseen data and unsupervised or reinforcement learning relying on internal evaluation metrics tied to the specific algorithm. Unsupervised learning has the advantage of not requiring labeled data which is an advantage for problems where labels are uncommon (Le et al. July 12, 2012). ref ref ref

Practical problems

2.1.2 Machine learning performance metrics

Intro

Training vs Validation metrics

Domain specific metrics

2.1.3 Hyperparameter optimization

Intro

Parameters given as part of a configuration to the machine learning model are called hyperparameters (Yang and Shami November 2020). For example learning rate, batch size or a classification threshold are hyperparameters set before the model is trained. Most common hyperparameter selection technique in practice is manually adjusting the hyperparameters and is humorously called graduate student descent.

ref

ref

Automatic tuning of hyperparameters can help achieve state-of-the-art performance in machine learning systems (Maclaurin, Duvenaud, and Adams April 2, 2015). Hyperparameter optimization techniques include grid search, random search, gradient based optimization and Bayesian optimization and they have different benefits and limitations (Yang and Shami November 2020).

Similar concepts to hyperparameter optimization are neural architecture optimization and meta modeling where model structure or modeling algorithm is treated as a tunable parameter (Baker et al. November 8, 2017). For example a machine learning system can automate the choosing of the number of neural network layers, the type of layers or even the type of machine learning algorithm. Tuning hyperparameters is generally a difficult task (Maclaurin, Duvenaud, and Adams April 2, 2015).

Practical problems

Traditional hyperparameter tuning methods such as Bayesian optimization are unfeasible for more than 10-20 hyperparameters (Maclaurin, Duvenaud, and Adams April 2, 2015). More advanced techniques are required if a larger amount of tunable hyperparameters is desired. Performance prediction is an important step to reduce the amount of computation required for neural architecture search and hyperparameter optimization (Baker et al. November 8, 2017).

Optimizing hyperparameter search

Early stopping

Early stopping is a technique in which model training is halted before completion. This is done to save on computational resources and the main reason for early stopping is predicting that the model will have poor performance. Performance prediction usually relies on predicting a machine learning performance metric such as RMSE or F1 score given the current model is performing during training compared to other models.

ref

ref

ref

2.2 DevOps

2.2.1 Overview

Intro to DevOps

Most common interpretation of DevOps is a focus on software quality, collaboration between development and operations, process speed and rapid feedback (Mishra and Otaiwi November 1, 2020; Waller, Ehmke, and Hasselbring April 3, 2015; Perera, Silva, and Perera September 2017). The exact definition is problematic as there is no consensus (Smeds, Nybom, and Porres 2015). DevOps can be viewed from different points of view such as culture, collaboration, automation, measurements and monitoring (Mishra and Otaiwi November 1, 2020; Waller, Ehmke, and Hasselbring April 3, 2015).

Continuous integration, continuous deployment and continuous monitoring are well known practices in DevOps (Waller, Ehmke, and Hasselbring April 3, 2015) describing the automatic nature of integrating, deploying and monitoring code changes. Performance profiling and monitoring are similar activities and the main difference is whether it's done during the development process or during operations respectively (Waller, Ehmke, and Hasselbring April 3, 2015). DevOps bridges the gap between evaluating performance during the development process and during operations (Brunnert et al. August 18, 2015).

TODO Devops lifecycle steps

Practical problems

TODO resource allocation/resource consumption, small memory software, benchmarking

2.2.2 System performance metrics

Intro

Performance metrics are fundamental to all activities involving performance evaluation such as profiling or monitoring (Brunnert et al. August 18, 2015). Common metrics involve measuring the CPU, but other metrics such as memory usage, network traffic or I/O usage are not as well defined as a CPU metric (Brunnert et al. August 18, 2015).

- Task Completion time
- Throughput
- Latency
- CPU usage
- GPU usage
- RAM usage
- VRAM usage
- I/O usage
- Network traffic

System metrics

Domain specific metrics

2.2.3 Continuous monitoring and benchmarking

Intro

Preprocessing

Training

Serving Latency

Practical problems

Resource demands might change depending on the inputs (Brunnert et al. August 18, 2015) making it important to systematically measure performance not only based on code changes but also on configuration changes or even data changes.

Continuous benchmarking

2.3 MLOps

2.3.1 Overview

TODO (Kreuzberger, Kühl, and Hirschl 2023) data scientists doing manual work issue from conclusions, definition of MLOps, limit to technical stuff and tooling

Requirements for a machine learning system are different depending on the task. For example speech and object recognition might have no particular performance requirements during training but has strict latency and computational resource restrictions when deployed to serve large amounts users (Hinton, Vinyals, and Dean March 9, 2015). One of the key areas of MLOps is using machine learning in production systems in addition to data processing and machine learning model training.

Performance measuring software is not new, but ML brings additional challenges in the form of models and data which requires a modified approach (Breck et al. 2017). It is also important to note, that not every data scientist or machine learning engineer working on machine learning systems has a software engineering background (Finzer 2013) and might lack the necessary knowledge to apply software engineering best practices to machine learning systems.

Projects involving machine learning often make common methodological mistakes that threaten the validity of the models, but are easily avoided. ref

TODO MLOps lifecycle steps

2.3.2 Machine learning development workflow

siirretään MLOps osioon

Projects involving machine learning often make common methodological mistakes that threaten the validity of the models, but are easily avoided. Several workflows for developing machine ref learning systems have emerged attempting to standardize the overall development process including CRISP-DM and KDD. These workflows will overall have similar steps. Using ref CRISP-DM as an example we have the following steps in the workflow: ref

1. Business and Data Understanding
2. Data Preparation
3. Modeling
4. Evaluation
5. Deployment

ref crisp-dm

Business and Data understanding refers to understanding the problem and the datasets.

Preprocessing, feature engineering

TODO what DevOps brings to ML

TODO Continuous Training

2.3.3 AutoML

Machine learning systems in addition to machine learning performance metrics and system performance metrics will have their performance metrics tied to product or organization metrics such as user churn rate or click-through rate (Shankar et al. September 16, 2022). Choosing the right metrics to evaluate a machine learning system is important and the metrics will be different for different machine learning systems (Shankar et al. September 16, 2022).

Automated Machine Learning (AutoML) aims to minimize human intervention in completing data analytics tasks using machine learning algorithms (Yang and Shami November 2022).

2.3.4 Performance prediction and early stopping

3 Methods

3.1 Research setup

3.1.1 Scope

The scope of the study is limited to 5 performance metrics of 3 different ML models trained and tested on 3 different datasets using a distributed computing framework Ray Tune (Liaw et al. July 13, 2018).

TODO different models, different datasets

TODO Vertailukriteeristö: tapana ohjelmistopuolella + tapana koneoppimispuolella

TODO different resources (memory, time, accuracy)

3.1.2 Research Questions

This master's thesis asks the following research questions:

- *RQ1*: How early stopping affects performance metrics during hyperparameter optimization?
- *RQ2*: How early stopping affects performance metrics during neural architecture optimization?

3.1.3 Methodology

Methodology used is expanded from an existing methodology for machine learning experiment design (Fernandez-Lozano et al. December 1, 2016) to include AutoML and

3.2 Experiments

3.2.1 Datasets

MNIST (Deng November 2012)

3.2.2 Machine Learning algorithms

3.2.3 Metrics

3.2.4 Training and validation

3.2.5 Inference

4 Results

TODO This is a results chapter

5 Discussion

5.1 Research Questions revisited

5.1.1 Research question RQ1

5.1.2 Research question RQ2

5.1.3 Research question RQ3

5.2 Interpretation

5.2.1 Implications for research

5.2.2 Implications for practice

5.3 Limitations

5.3.1 Datasets

5.3.2 Machine Learning algorithms

5.3.3 Metrics

5.3.4 Training and validation

5.3.5 Inference

5.4 Related Work

To find relevant related work both reverse snowballing and forward snowballing is used on a set of MLOps papers previously known to the author.

Benchmarking ML systems (Cardoso Silva et al. December 2020)

5.5 Future Work

TODO This is a discussion chapter

6 Conclusions

Summary

TODO This is a conclusions chapter

Bibliography

Amodei, Dario, and Danny Hernandez. May 16, 2018. "AI and Compute", May 16, 2018. Visited on July 29, 2023. https://openai.com/research/ai-and-compute.

Baker, Bowen, Otkrist Gupta, Ramesh Raskar, and Nikhil Naik. November 8, 2017. "Accelerating Neural Architecture Search Using Performance Prediction". Preprint, November 8, 2017. Visited on January 25, 2023. https://doi.org/10.48550/arXiv.1705.10823. arXiv: 1705.10823 [cs]. http://arxiv.org/abs/1705.10823.

Breck, Eric, Shanqing Cai, Eric Nielsen, Michael Salib, and D. Sculley. 2017. "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction". In *Proceedings of IEEE Big Data.*

Brunnert, Andreas, Andre van Hoorn, Felix Willnecker, Alexandru Danciu, Wilhelm Hasselbring, Christoph Heger, Nikolas Herbst, et al. August 18, 2015. "Performance-Oriented DevOps: A Research Agenda". Preprint, August 18, 2015. Visited on January 17, 2023. https://doi.org/10.48550/arXiv.1508.04752. arXiv: 1508.04752 [cs]. http://arxiv.org/abs/1508.04752.

Cardoso Silva, Lucas, Fernando Rezende Zagatti, Bruno Silva Sette, Lucas Nildaimon dos Santos Silva, Daniel Lucrédio, Diego Furtado Silva, and Helena de Medeiros Caseli. December 2020. "Benchmarking Machine Learning Solutions in Production". In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA),* 626–633. 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA). https://doi.org/10.1109/ICMLA51294.2020.00104.

Chen, Jiasi, and Xukan Ran. August 2019. "Deep Learning With Edge Computing: A Review". *Proceedings of the IEEE* 107, number 8 (): 1655–1674. ISSN: 0018-9219, 1558-2256, visited on July 29, 2023. https://doi.org/10.1109/JPROC.2019.2921977. https://ieeexplore.ieee.org/document/8763885/.

Deng, Li. November 2012. “The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]”. *IEEE Signal Processing Magazine* 29, number 6 (): 141–142. ISSN: 1558-0792. https://doi.org/10.1109/MSP.2012.2211477.

Domingos, Pedro. October 1, 2012. “A Few Useful Things to Know about Machine Learning”. *Communications of the ACM* 55, number 10 (October 1, 2012): 78–87. ISSN: 0001-0782, visited on March 14, 2023. https://doi.org/10.1145/2347736.2347755. https://doi.org/10.1145/2347736.2347755.

Fernandez-Lozano, Carlos, Marcos Gestal, Cristian R. Munteanu, Julian Dorado, and Alejandro Pazos. December 1, 2016. “A Methodology for the Design of Experiments in Computational Intelligence with Multiple Regression Models”. *PeerJ* 4 (December 1, 2016): e2721. ISSN: 2167-8359, visited on February 15, 2023. https://doi.org/10.7717/peerj.2721. pmid: 27920952. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5136129/.

Finzer, William. 2013. “The Data Science Education Dilemma”. *Technology Innovations in Statistics Education* 7 (2). Visited on January 17, 2023. https://doi.org/10.5070/T572013891. https://escholarship.org/uc/item/7gv0q9dc.

Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. March 9, 2015. “Distilling the Knowledge in a Neural Network”. Preprint, March 9, 2015. Visited on February 3, 2023. https://doi.org/10.48550/arXiv.1503.02531. arXiv: 1503.02531 `[cs, stat]`. http://arxiv.org/abs/1503.02531.

Hoffer, Elad, Itay Hubara, and Daniel Soudry. January 1, 2018. “Train Longer, Generalize Better: Closing the Generalization Gap in Large Batch Training of Neural Networks”. Preprint, January 1, 2018. Visited on August 2, 2023. https://doi.org/10.48550/arXiv.1705.08741. arXiv: 1705.08741 `[cs, stat]`. http://arxiv.org/abs/1705.08741.

Kreuzberger, Dominik, Niklas Kühl, and Sebastian Hirschl. 2023. “Machine Learning Operations (MLOps): Overview, Definition, and Architecture”. *IEEE Access* 11:31866–31879. ISSN: 2169-3536. https://doi.org/10.1109/ACCESS.2023.3262138.

Le, Quoc V., Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Jeff Dean, and Andrew Y. Ng. July 12, 2012. "Building High-Level Features Using Large Scale Unsupervised Learning". Preprint, July 12, 2012. Visited on February 3, 2023. https://doi.org/10.48550/arXiv.1112.6209. arXiv: 1112.6209 [cs]. http://arxiv.org/abs/1112.6209.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. May 2015. "Deep Learning". *Nature* 521, number 7553 (7553): 436–444. ISSN: 1476-4687, visited on June 15, 2023. https://doi.org/10.1038/nature14539. https://www.nature.com/articles/nature14539.

Liaw, Richard, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E. Gonzalez, and Ion Stoica. July 13, 2018. "Tune: A Research Platform for Distributed Model Selection and Training". Preprint, July 13, 2018. Visited on February 22, 2023. https://doi.org/10.48550/arXiv.1807.05118. arXiv: 1807.05118 [cs, stat]. http://arxiv.org/abs/1807.05118.

Maclaurin, Dougal, David Duvenaud, and Ryan P. Adams. April 2, 2015. "Gradient-Based Hyperparameter Optimization through Reversible Learning". Preprint, April 2, 2015. Visited on February 3, 2023. https://doi.org/10.48550/arXiv.1502.03492. arXiv: 1502.03492 [cs, stat]. http://arxiv.org/abs/1502.03492.

Mishra, Alok, and Ziadoon Otaiwi. November 1, 2020. "DevOps and Software Quality: A Systematic Mapping". *Computer Science Review* 38 (November 1, 2020): 100308. ISSN: 1574-0137, visited on January 17, 2023. https://doi.org/10.1016/j.cosrev.2020.100308. https://www.sciencedirect.com/science/article/pii/S1574013720304081.

Olson, Randal S., William La Cava, Patryk Orzechowski, Ryan J. Urbanowicz, and Jason H. Moore. December 11, 2017. "PMLB: A Large Benchmark Suite for Machine Learning Evaluation and Comparison". *BioData Mining* 10, number 1 (December 11, 2017): 36. ISSN: 1756-0381, visited on February 22, 2023. https://doi.org/10.1186/s13040-017-0154-4. https://doi.org/10.1186/s13040-017-0154-4.

OpenAI. November 30, 2022. "Introducing ChatGPT", November 30, 2022. Visited on July 24, 2023. https://openai.com/blog/chatgpt.

Perera, Pulasthi, Roshali Silva, and Indika Perera. September 2017. "Improve Software Quality through Practicing DevOps". In *2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer),* 1–6. 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer). https://doi.org/10.1109/ICTER.2017.8257807.

Prechelt, Lutz. June 1, 1998. "Automatic Early Stopping Using Cross Validation: Quantifying the Criteria". *Neural Networks* 11, number 4 (June 1, 1998): 761–767. ISSN: 0893-6080, visited on August 2, 2023. https://doi.org/10.1016/S0893-6080(98)00010-0. https://www.sciencedirect.com/science/article/pii/S0893608098000100.

Rombach, Robin, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. April 13, 2022. "High-Resolution Image Synthesis with Latent Diffusion Models". Preprint, April 13, 2022. Visited on July 22, 2023. https://doi.org/10.48550/arXiv.2112.10752. arXiv: 2112.10752 [cs]. http://arxiv.org/abs/2112.10752.

Shankar, Shreya, Rolando Garcia, Joseph M. Hellerstein, and Aditya G. Parameswaran. September 16, 2022. "Operationalizing Machine Learning: An Interview Study". Preprint, September 16, 2022. Visited on December 7, 2022. https://doi.org/10.48550/arXiv.2209.09125. arXiv: 2209.09125 [cs]. http://arxiv.org/abs/2209.09125.

Shmueli, Galit. August 1, 2010. "To Explain or to Predict?" *Statistical Science* 25, number 3 (August 1, 2010). ISSN: 0883-4237, visited on March 14, 2023. https://doi.org/10.1214/10-STS330. arXiv: 1101.0891 [stat]. http://arxiv.org/abs/1101.0891.

Smeds, Jens, Kristian Nybom, and Ivan Porres. 2015. "DevOps: A Definition and Perceived Adoption Impediments". In *Agile Processes in Software Engineering and Extreme Programming,* edited by Casper Lassenius, Torgeir Dingsøyr, and Maria Paasivaara, 166–177. Lecture Notes in Business Information Processing. Cham: Springer International Publishing. ISBN: 978-3-319-18612-2. https://doi.org/10.1007/978-3-319-18612-2_14.

Stability AI. August 22, 2022. "Stable Diffusion Public Release". Stability AI, August 22, 2022. Visited on July 24, 2023. https://stability.ai/blog/stable-diffusion-public-release.

Strubell, Emma, Ananya Ganesh, and Andrew McCallum. April 3, 2020. "Energy and Policy Considerations for Modern Deep Learning Research". *Proceedings of the AAAI Conference on Artificial Intelligence* 34, number 09 (09 April 3, 2020): 13693–13696. ISSN: 2374-3468, visited on July 29, 2023. https://doi.org/10.1609/aaai.v34i09.7123. https://ojs.aaai.org/index.php/AAAI/article/view/7123.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, et al. February 27, 2023. "LLaMA: Open and Efficient Foundation Language Models". Preprint, February 27, 2023. Visited on July 24, 2023. https://doi.org/10.48550/arXiv.2302.13971. arXiv: 2302.13971 [cs]. http://arxiv.org/abs/2302.13971.

Waller, Jan, Nils C. Ehmke, and Wilhelm Hasselbring. April 3, 2015. "Including Performance Benchmarks into Continuous Integration to Enable DevOps". *ACM SIGSOFT Software Engineering Notes* 40, number 2 (April 3, 2015): 1–4. ISSN: 0163-5948, visited on January 17, 2023. https://doi.org/10.1145/2735399.2735416. https://doi.org/10.1145/2735399.2735416.

Yang, Li, and Abdallah Shami. November 2020. "On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice". *Neurocomputing* 415 (): 295–316. ISSN: 09252312, visited on January 25, 2023. https://doi.org/10.1016/j.neucom.2020.07.061. arXiv: 2007.15745 [cs, stat]. http://arxiv.org/abs/2007.15745.

———. November 2022. "IoT Data Analytics in Dynamic Environments: From An Automated Machine Learning Perspective". *Engineering Applications of Artificial Intelligence* 116 (): 105366. ISSN: 09521976, visited on January 25, 2023. https://doi.org/10.1016/j.engappai.2022.105366. arXiv: 2209.08018 [cs, eess]. http://arxiv.org/abs/2209.08018.

Zaharia, M., A. Chen, A. Davidson, A. Ghodsi, S. Hong, A. Konwinski, Siddharth Murching, et al. 2018. "Accelerating the Machine Learning Lifecycle with MLflow". *IEEE Data Eng. Bull.,* visited on March 14, 2023. https://www.semanticscholar.org/paper/Accelerating-the-Machine-Learning-Lifecycle-with-Zaharia-Chen/b2e0b79e6f180af2e0e559f2b1faba66b2bd578a.