Learning to Optimize: A Primer and A Benchmark

Tianlong Chen[1] **Xiaohan Chen**[2] **Wuyang Chen**[1] **Howard Heaton**[3] **Jialin Liu**[2]
Zhangyang Wang[1] **Wotao Yin**[2]

[1]The University of Texas at Austin
[2]Decision Intelligence Lab, Alibaba Group (U.S.)
[3]Typal Research, Typal LLC

Abstract Learning to optimize (L2O) is an emerging approach that leverages machine learning to develop optimization methods, aiming at reducing the laborious iterations of hand engineering. It automates the design of an optimization method based on its performance on a set of training problems. This data-driven procedure generates methods that can efficiently solve problems similar to those in training. In sharp contrast, the typical and traditional designs of optimization methods are theory-driven, so they obtain performance guarantees over the classes of problems specified by the theory. The difference makes L2O suitable for repeatedly solving a particular optimization problem over a specific distribution of data, while it typically fails on out-of-distribution problems. The practicality of L2O depends on the type of target optimization, the chosen architecture of the method to learn, and the training procedure. This new paradigm has motivated a community of researchers to explore L2O and report their findings. This article is poised to be the first comprehensive survey and benchmark of L2O for continuous optimization. We set up taxonomies, categorize existing works and research directions, present insights, and identify open challenges. We benchmarked many existing L2O approaches on a few representative optimization problems. For reproducible research and fair benchmarking purposes, we released our software implementation and data in the package Open-L2O at https://github.com/VITA-Group/Open-L2O.

1 Broader Impact Statement

The emergence of Learning to Optimize (L2O) has the potential to revolutionize optimization methods by automating the design process and reducing the need for laborious iterations of hand engineering. This approach can significantly improve efficiency and accuracy across a wide range of applications. The development of L2O has implications for fields such as computer science, engineering, mathematics, and data science. As L2O research continues to progress, it has the potential to transform how we approach optimization problems and improve our ability to solve complex real-world challenges. We do not find any apparent adverse ethical or societal implications of L2O research.

2 Submission Checklist

1. For all authors…

   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]

   (b) Did you describe the limitations of your work? [N/A] This is a survey paper and does not involve new methods. We discussed the limitations of the investigated works and the open directions in the Learning to Optimize (L2O) field in Section 5.

   (c) Did you discuss any potential negative societal impacts of your work? [No] We do not anticipate any adverse ethical or societal implications as we stated in the Broader Impact Statement.

 © 2023 the authors, released under CC BY 4.0

(d) Have you read the ethics author's and review guidelines and ensured that your paper conforms to them? `https://automl.cc/ethics-accessibility/` [Yes]

2. If you are including theoretical results…

   (a) Did you state the full set of assumptions of all theoretical results? [N/A] We did not include new theoretical results.

   (b) Did you include complete proofs of all theoretical results? [N/A] We did not include new theoretical results.

3. If you ran experiments…

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results, including all requirements (e.g., `requirements.txt` with explicit version), an instructive `README` with installation, and execution commands (either in the supplemental material or as a URL)? [Yes] We open-sourced the full implementation with instructions of reproducing the experimental results and provided the URL in the abstract.

   (b) Did you include the raw results of running the given instructions on the given code and data? [Yes]

   (c) Did you include scripts and commands that can be used to generate the figures and tables in your paper based on the raw results of the code, data, and instructions given? [Yes]

   (d) Did you ensure sufficient code quality such that your code can be safely executed and the code is properly documented? [Yes]

   (e) Did you specify all the training details (e.g., data splits, pre-processing, search spaces, fixed hyperparameter settings, and how they were chosen)? [Yes]

   (f) Did you ensure that you compared different methods (including your own) exactly on the same benchmarks, including the same datasets, search space, code for training and hyperparameters for that code? [Yes]

   (g) Did you run ablation studies to assess the impact of different components of your approach? [N/A] This is a survey paper and we do not introduce new methods.

   (h) Did you use the same evaluation protocol for the methods being compared? [Yes]

   (i) Did you compare performance over time? [No] This study primarily concentrated on the progression of iterations instead of the duration of time.

   (j) Did you perform multiple runs of your experiments and report random seeds? [Yes]

   (k) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

   (l) Did you use tabular or surrogate benchmarks for in-depth evaluations? [Yes] We used benchmarks that are widely used in the community: synthetic optimization functions and loss functions during the training of neural networks for MNIST classification.

   (m) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We provided the hardware information in the GitHub repository. To be specific, our experiments were conducted on a cluster with two GPUs (GeForce RTX 3080) and a 14-core CPU (Intel(R) Core(TM) i9-9940X).

   (n) Did you report how you tuned hyperparameters, and what time and resources this required (if they were not automatically tuned by your AutoML method, e.g. in a NAS approach; and also hyperparameters of your own method)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] We only used existing assets and synthetic data. Details of how synthetic data are generated are included in the GitHub repository mentioned in the abstract.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] We did not use any sensitive dataset, instead, we only used synthetic data and widely known open-source dataset MNIST.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] We did not use any sensitive dataset, instead, we only used synthetic data and widely known open-source dataset MNIST.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] We did not involve crowdsourcing or human subjects.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We did not involve crowdsourcing or human subjects.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] We did not involve crowdsourcing or human subjects.