From Predictions to Sustainability: Rethinking AutoML Priorities

Marius Lindauer

Green-AutoML Team* at LUH|AI:


Leona Hennig


Daphne Theodorakopoulos


Tanja Tornede

* and many more contributing to that vision

The Need for AutoML!?

Rise of Literacy

Photo by Anna Hunko on Unsplash

- Only priests were able to read and write
- People believed that they don't need to read and write
- They went to the holy buildings

- Today, everyone can read and write
- No one doubts the benefits of it

- ⇒ **Democratization of literacy**

Inspired by Andrew Ng

Rise of AI Literacy?

Photo by Max Duzij on Unsplash

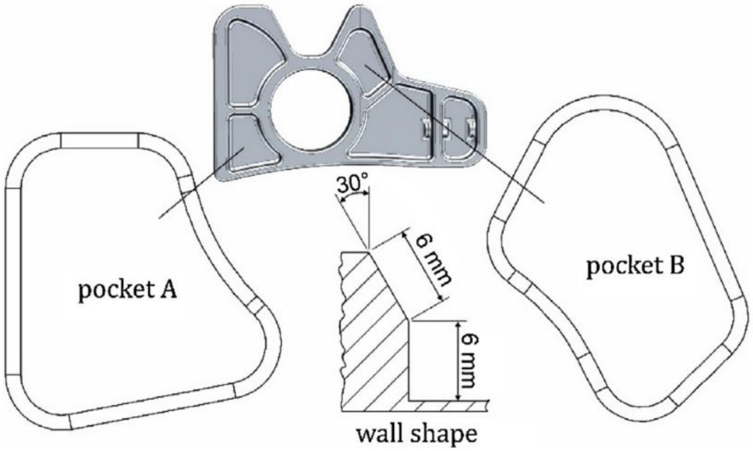
- Only highly educated people can program new AI applications
- Power only with the large IT companies

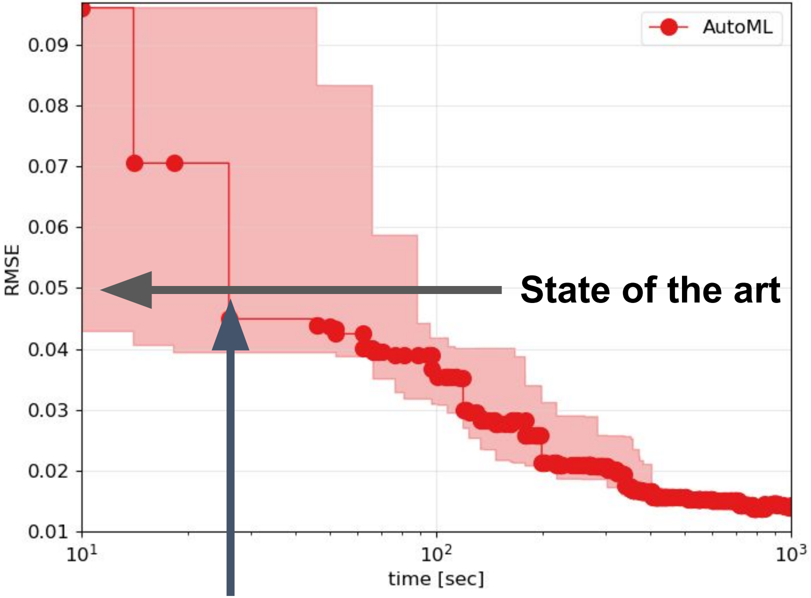
- In an age of limited resources, the need for efficient use gets more important

- **AutoML contributes to AI literacy!**

[See also my TEDx Talk]

Shape Error Prediction in Milling Processes





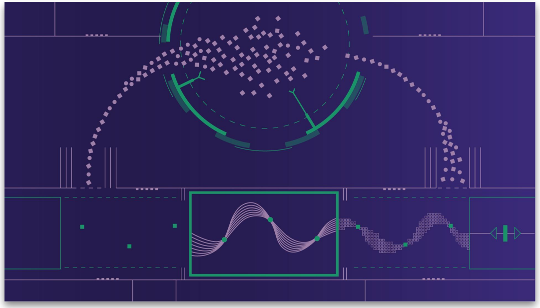
State of the art

Better than state of the art in less than 30sec!

From ML Alchemy to Science

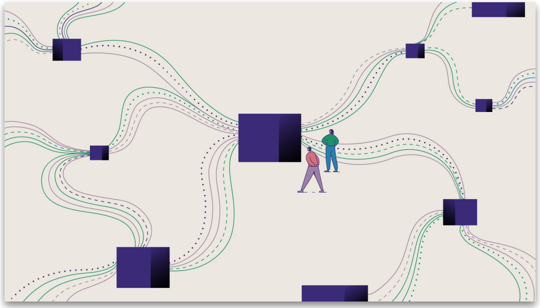
"You can teach an old dog new tricks" [Ruffinelli et al. 2019]
→ Hyperparameter tuning might not be the only required solution, but without it, it will also be hard.

Tasks Automated by AutoML

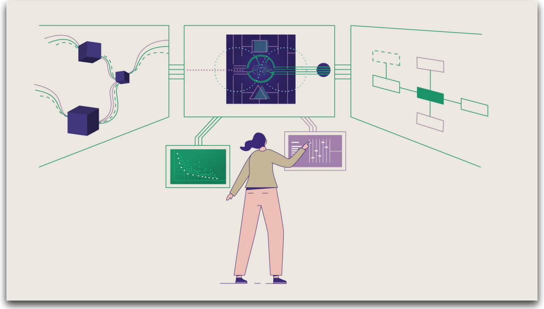


Hyperparameter Optimization

Neural Architecture Search

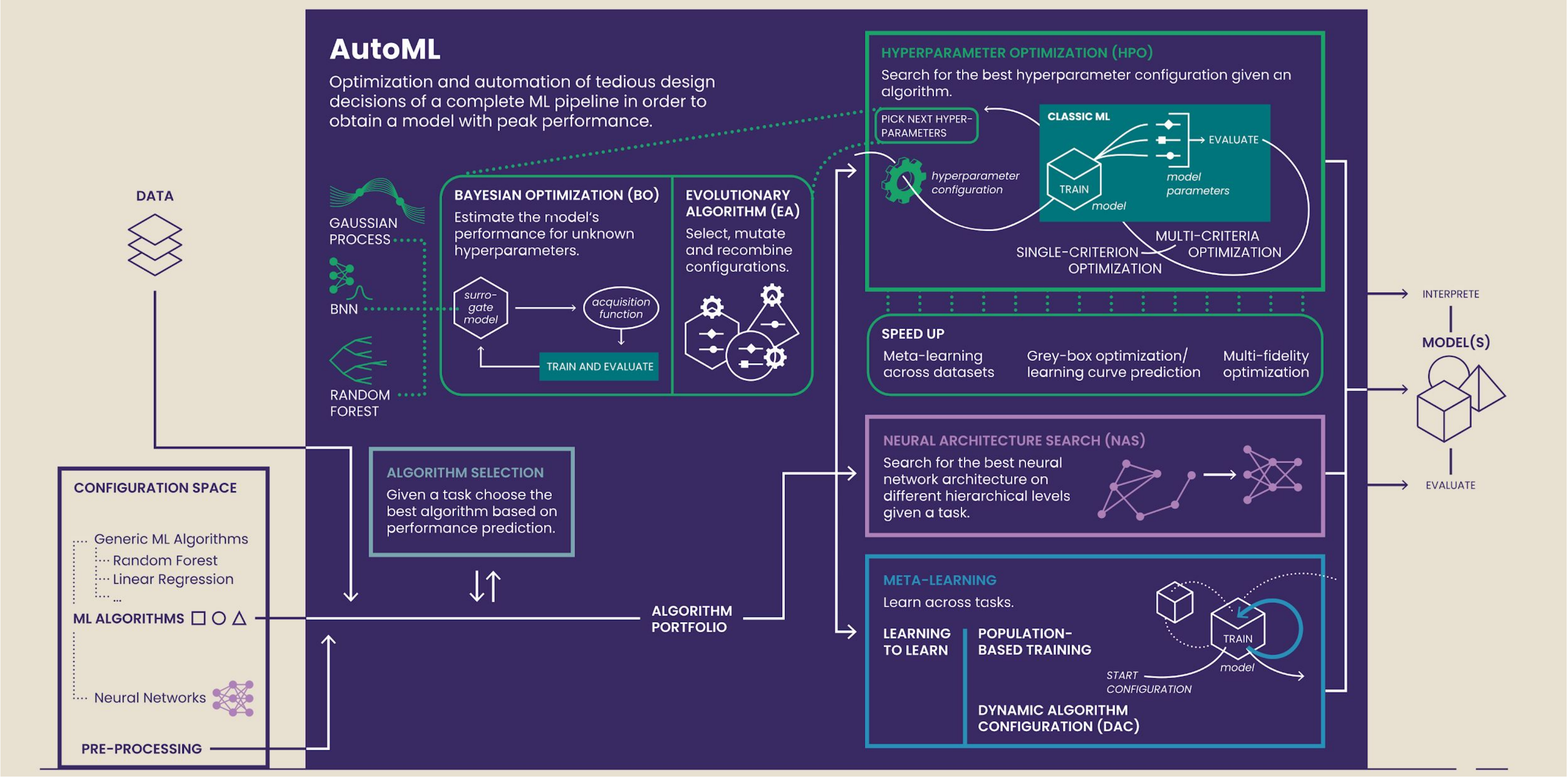


Meta Learning



Pipeline Design

AutoML A-Z



AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
DATA
GAUSSIAN PROCESS
BNN
RANDOM FOREST
BAYESIAN OPTIMIZATION (BO)
Estimate the model's performance for unknown hyperparameters.
surro-gate model
acquisition function
TRAIN AND EVALUATE
EVOLUTIONARY ALGORITHM (EA)
Select, mutate and recombine configurations.
HYPERPARAMETER OPTIMIZATION (HPO)
Search for the best hyperparameter configuration given an algorithm.
PICK NEXT HYPER-PARAMETERS
CLASSIC ML
EVALUATE
hyperparameter configuration
TRAIN
model
model parameters
MULTI-CRITERIA OPTIMIZATION
SINGLE-CRITERION OPTIMIZATION
SPEED UP
Meta-learning across datasets
Grey-box optimization/ learning curve prediction
Multi-fidelity optimization
NEURAL ARCHITECTURE SEARCH (NAS)
Search for the best neural network architecture on different hierarchical levels given a task.
META-LEARNING
Learn across tasks.
LEARNING TO LEARN
POPULATION-BASED TRAINING
TRAIN
model
START CONFIGURATION
DYNAMIC ALGORITHM CONFIGURATION (DAC)
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS
Neural Networks
PRE-PROCESSING
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance prediction.
ALGORITHM PORTFOLIO
INTERPRETE
MODEL(S)
EVALUATE

Advantages

AutoML enables

- 🚀 More **efficient** research and development of ML applications
  - → AutoML has been shown to outperform humans on subproblems
- 🧮 More **systematic** research and development of ML applications
  - → no (human) bias or unsystematic evaluation
- 📋 More **reproducible** research
  - → since it is systematic!
- 🦾 **Broader use** of ML methods
  - → less required ML expert knowledge
  - → better results with higher predictive performance
  - → not only limited to computer scientists

Sustainability is also our responsibility!

Current Focus of AutoML Research

Chevron-double-up Icons by Laisa Islam Ani
Graph Icons by Secret Studio
Chip Icons by srip

Better predictive performance (e.g., accuracy)

⇒ other quality indicators (e.g., energy efficiency) are often ignored

Scaling to larger models (e.g., LLMs)

⇒ AutoML needs to be more efficient
(e.g., via multi-fidelity optimization or expert knowledge integration)
⇒ Mindset less on energy efficiency but to apply AutoML to ever larger models
(each of training of them requires more and more energy)

Adaption to different hardware constraints
(e.g., embedded systems, smartphones)

⇒ Main objective: How can I get the best out of an AI on a given hardware module?
⇒ Rarely: How can I achieve the best with the fewest possible resources?

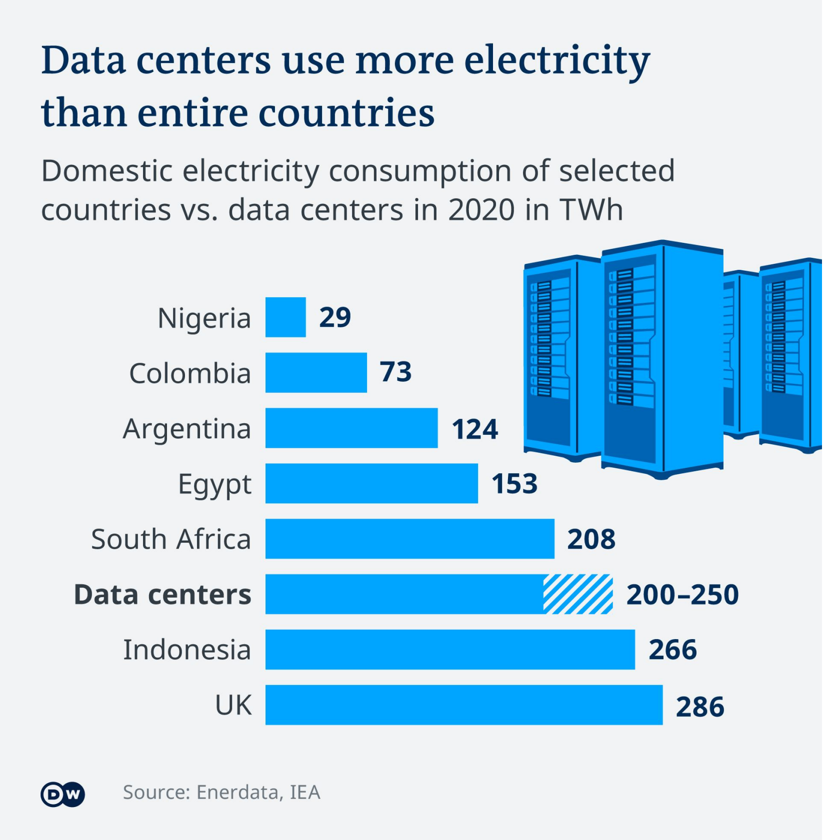
Energy Consumption of AutoML can be Huge

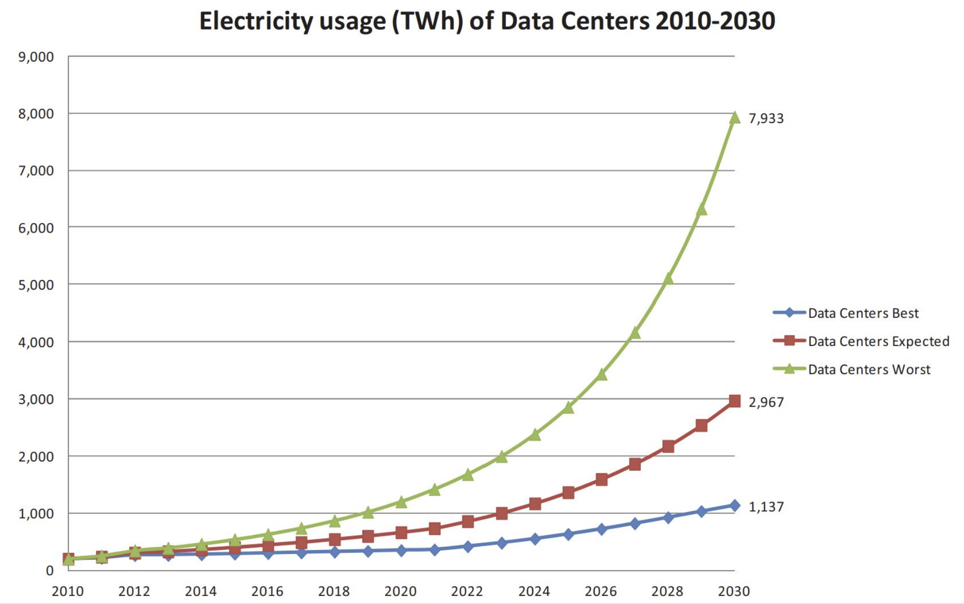
- Neural Architecture Search with Reinforcement Learning [Zoph et al. 2016]
  - 800 GPUs for 28 days
  - 250W TDP
  - 134.4 MWh
  - Yearly consumption of 30x 4-persons households
  - 483,840$ on a commercial cluster

- *Disclaimer*: Neural architecture search is more efficient than orders of magnitude by now. [White et al. 2023]

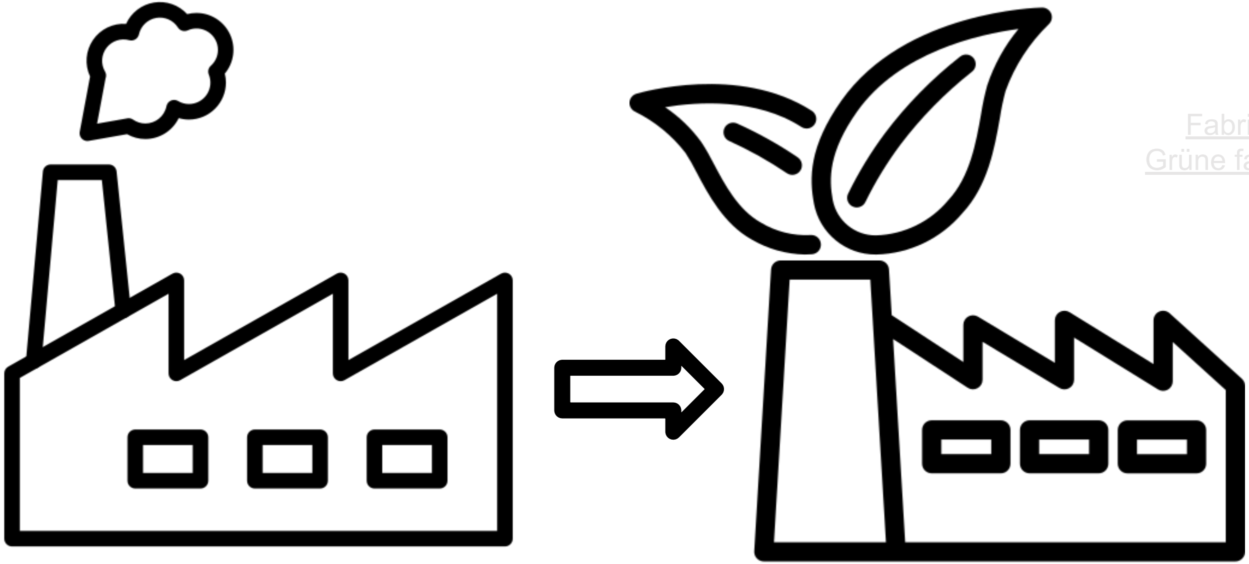
- However… some AutoML methods are still super expensive:
  - For AutoML-Zero, Real et al. (2020) trained 10^{12} deep neural networks

Energy Consumption of Data Centers





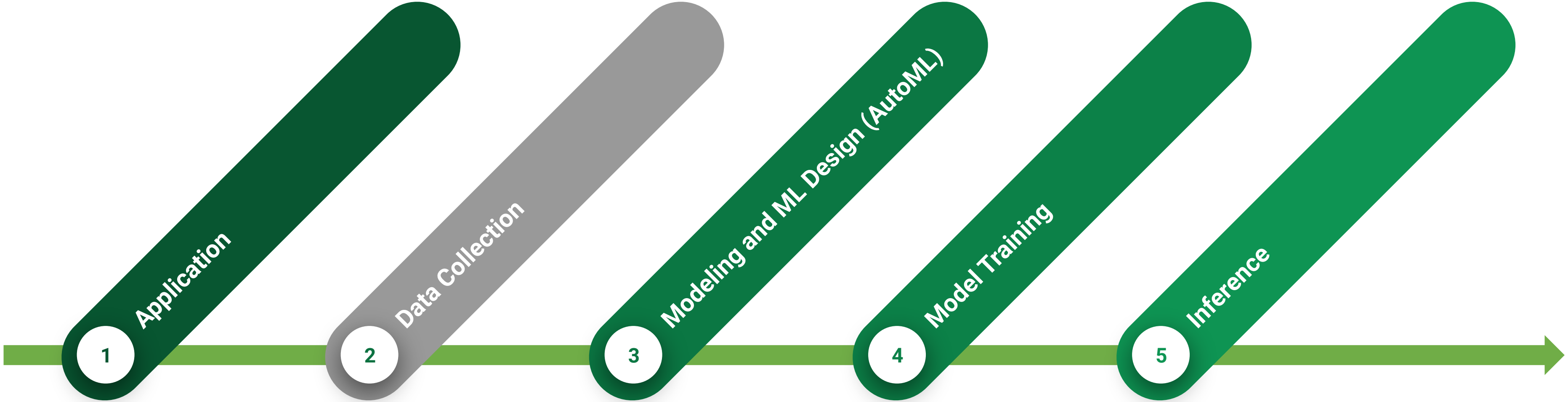
[Andrae. 2015]



Fabrik Icons by DinosoftLabs
Grüne fabrik Icons by kosonicon

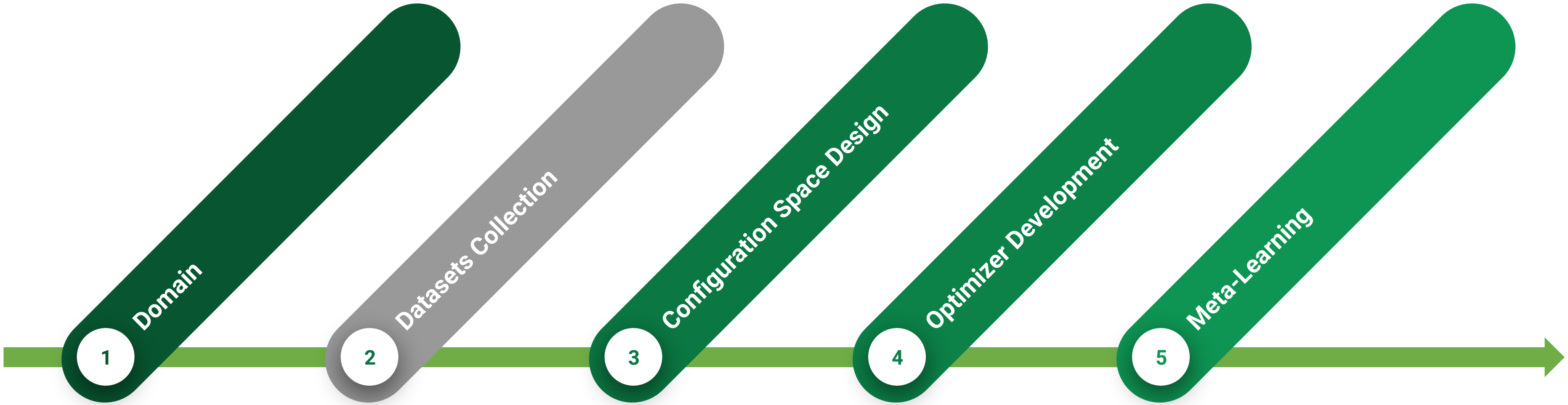
Green AutoML?

ML development



All of that requires compute power and consumes resources / produces CO_2e.

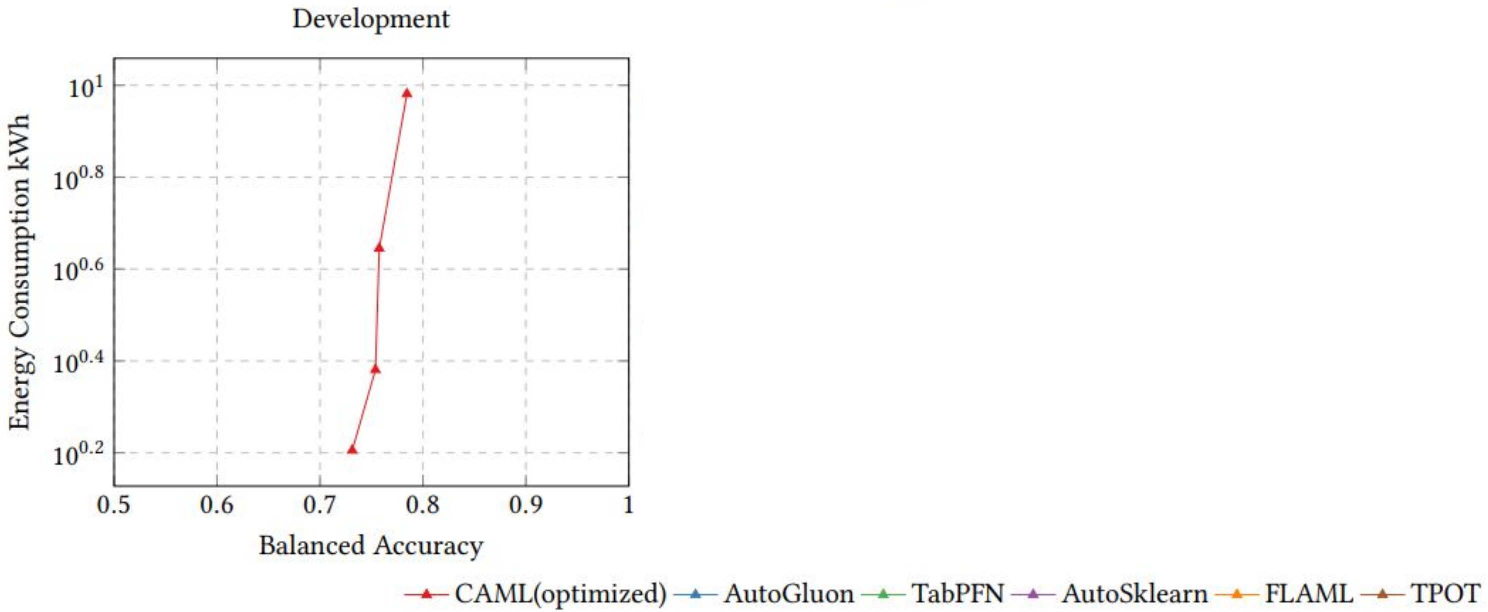
AutoML development



All of that requires compute power and consumes resources / produces CO_2e.

⇒ Diminishing effects since we develop AutoML for many applications and not only for one. Nevertheless, not negligible.

Energy-Consumption of AutoML Tools
[Neutatz et al. 2023 – WIP]

Meta-Learning AutoML Settings

- AutoML is sensitive to its own algorithmic meta-parameters
- Our prior work for making AutoML more efficient by meta-learning:
  - [Lindauer et al. 2018] studied the impact of the parameters of Bayesian Optimization for different HPO tasks (by using algorithm configuration)
  - [Feurer et al. 2022] meta-learned validation strategies and warmstartig portfolios for different datasets
  - [Neutatz et al. 2023] meta-learn the configuration space, validation strategy, ensembling and incremental training for different datasets and application-constraints
- **Assumption**: If we invest more time into the development of AutoML packages (incl. meta-learning), we save a lot of compute resources later on while using it

AutoML in Heavily Constrained Applications

[Neutatz et al. 2023]


Default AutoML Configuration

Validation Strategy:	Holdout 66/33
Ensembling:	yes
Incremental Training:	yes
Validation split reshuffle:	no
ML Hyperparameter space:	
SVM:	Yes
SVM_tol:	Yes
SVM_C:	Yes
Extra Trees:	Yes
KNN:	Yes
Multilayer Perceptron:	Yes
Any Feature Preprocessor:	Yes
302 hyperparameters	Yes

Adapt AutoML parameters to ML task and deactivate undesired ML hyperparameters

Dynamic AutoML Configuration

Validation Strategy:	Holdout 46/54
Ensembling:	no
Incremental Training:	yes
Validation split reshuffle:	yes
ML Hyperparameter space:	
SVM:	Yes
SVM_tol:	Yes
SVM_C:	No
Extra Trees:	Yes
KNN:	No
Multilayer Perceptron:	No
Any Feature Preprocessor:	No
302 hyperparameters	Yes/No

Searches for the optimal ML pipeline in the defined search space. A pipeline is defined by the selected ML hyperparameters.

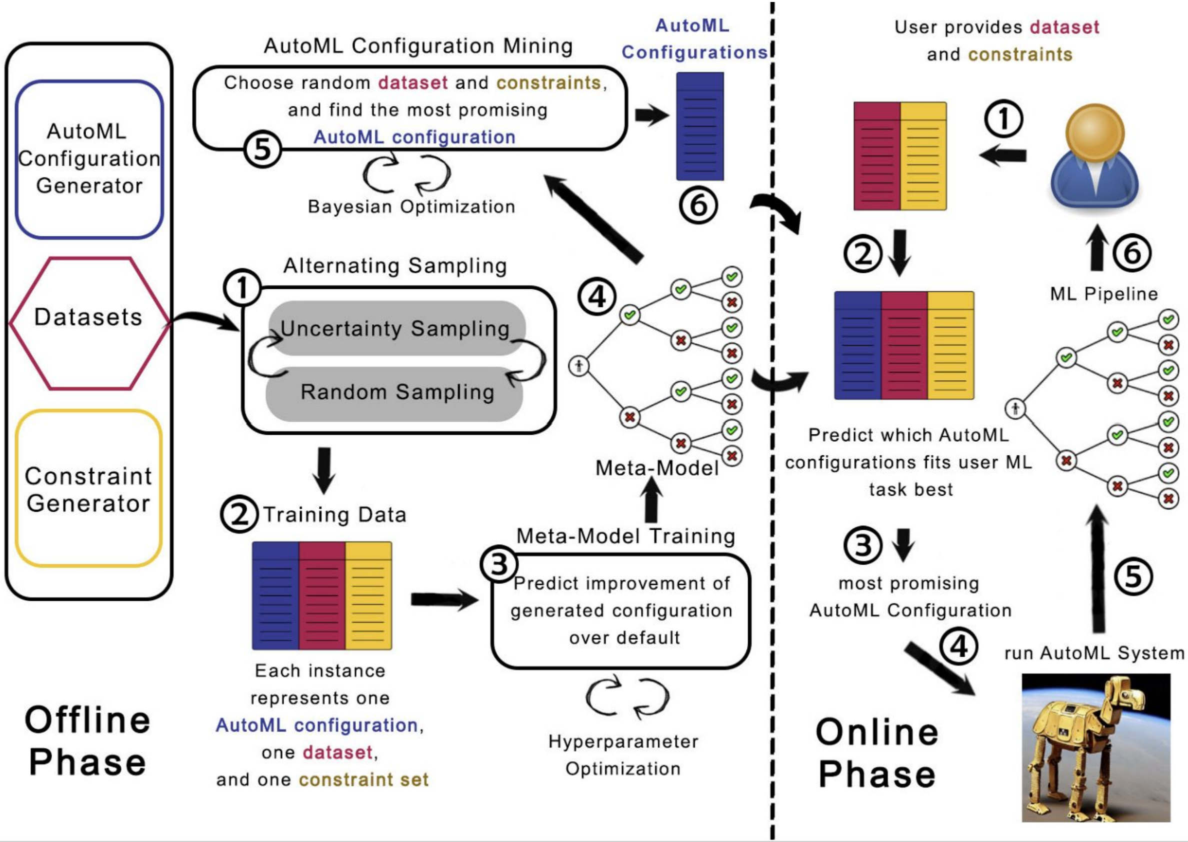
ML Pipeline

For SVM, the **model parameters** are the weights w:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\mathbf{w}^\mathsf{T}\mathbf{x}_i - b)\right)\right] + \lambda\|\mathbf{w}\|^2.$$

ML Hyperparameters:	
SVM:	Yes
SVM_tol:	1e-5
SVM_C:	1.0 (default)
Extra Trees:	No
KNN:	No
Multilayer Perceptron:	No
Any Feature Preprocessor:	No
302 hyperparameters	...

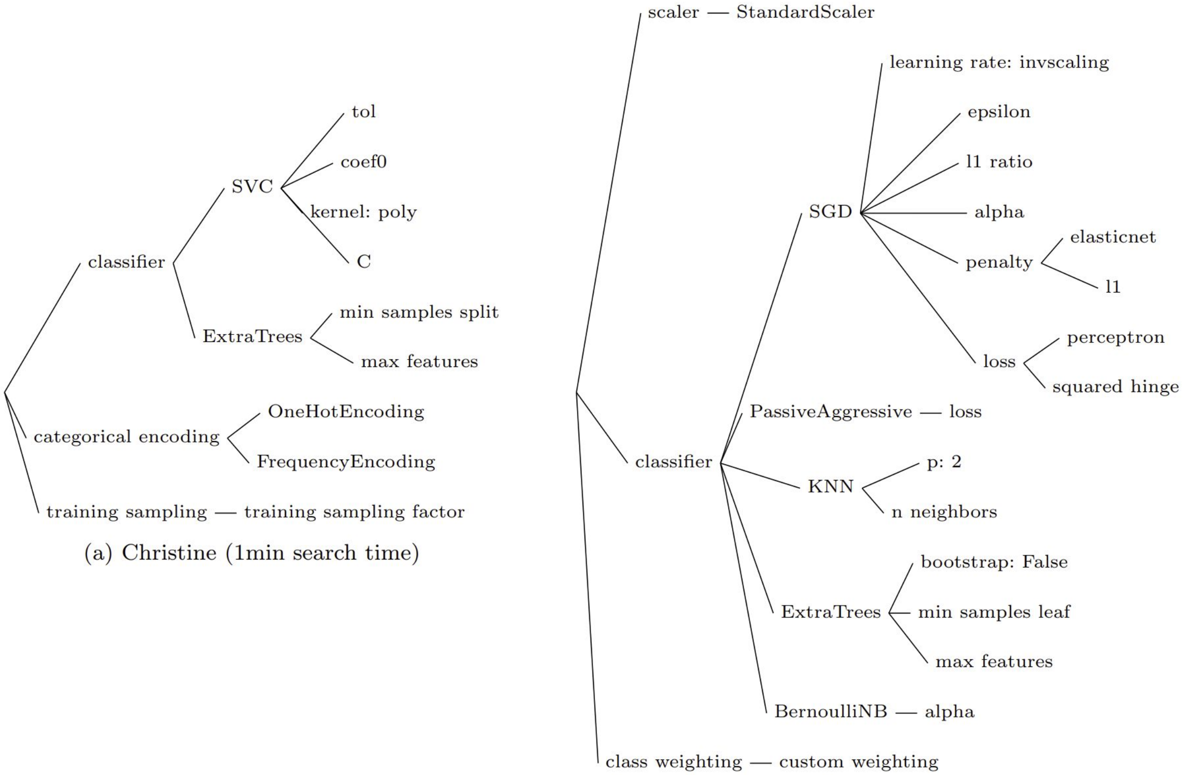
AutoML in Heavily Constrained Applications
[Neutatz et al. 2023]



Possible application constraints:

- AutoML budget
- Inference time
- Memory consumption
- Energy consumption
- Fairness thresholds
- …

Can it learn to select different configuration spaces? [Neutatz et al. 2023]



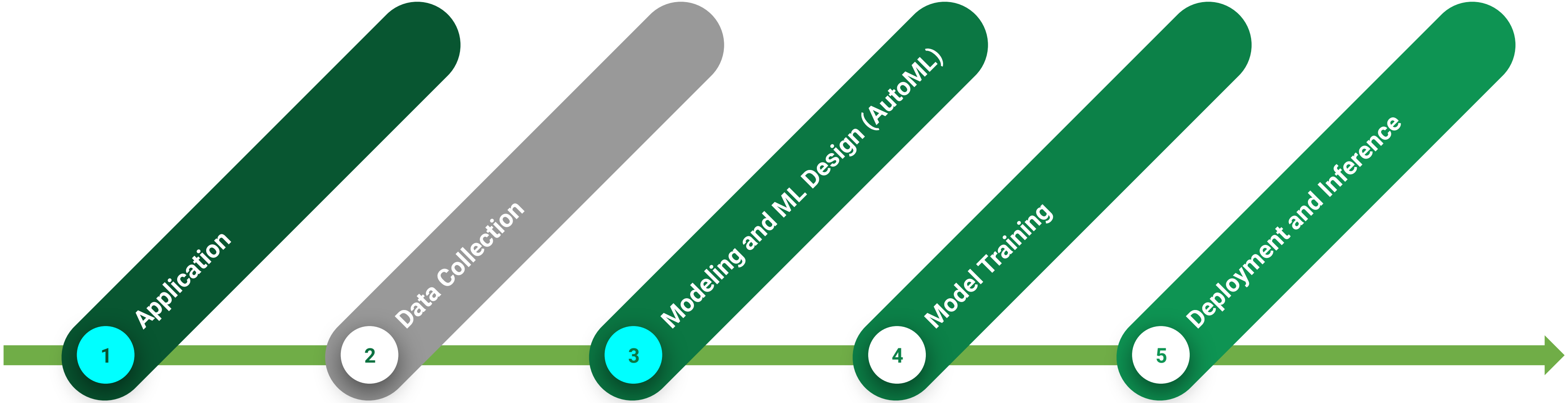
(a) Christine (1min search time)

(b) Robert (5min search time)

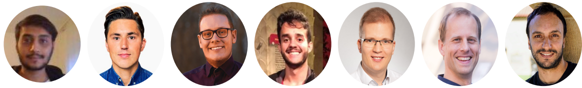
Take-Aways for Meta-Learning AutoML Conf.

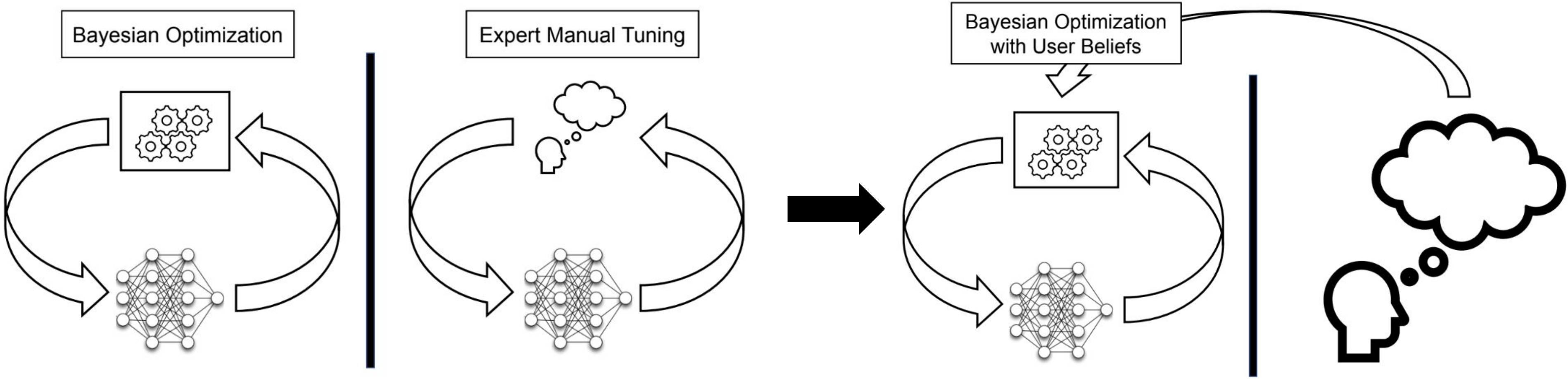

- **Assumption**: If we invest more time into the development of AutoML packages (incl. meta-learning), we save a lot of compute resources for using it
- **Positive** take-away:
  Yes, we can meta-learn how to configure AutoML systems and achieve new state-of-the-art performance
- **Negative** take-away:
  We cannot easily do it for large AutoML budgets (beyond 10min) without enormous compute resources
- **Future challenge**: How to configure AutoML on expensive tasks;

  "Expensive" can mean:
  - very expensive ML models (e.g., LLMs)
  - very complex configuration spaces with thousands of ML trainings


1
Application
2
Data Collection
3
Modeling and ML Design (AutoML)
4
Model Training
5
Deployment and Inference

Making Use of Human Expertise





- Bayesian approach based on Gaussian Processes [Souza et al. ECML 2020]

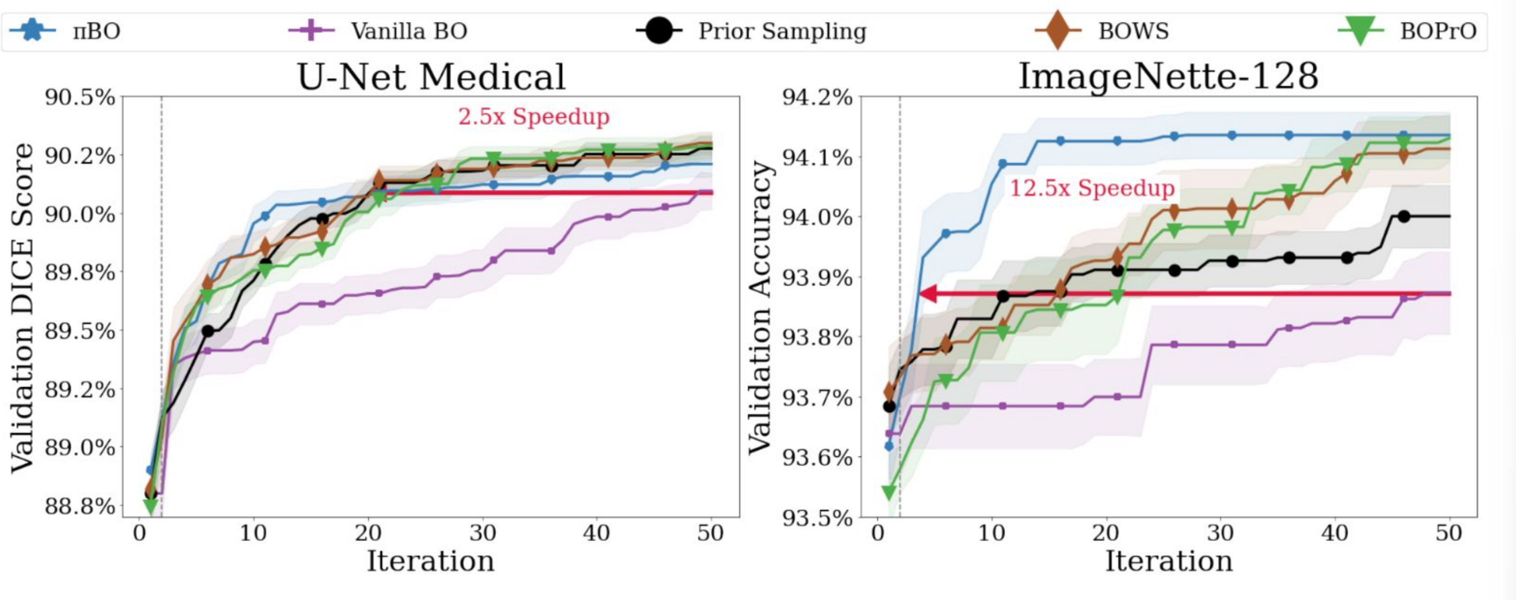
$$\mathcal{M}_g(\boldsymbol{x}) = p(f(\boldsymbol{x}) < f_\gamma | \boldsymbol{x}, \mathcal{D}_t) = \Phi\left(\frac{f_\gamma - \mu_{\boldsymbol{x}}}{\sigma_{\boldsymbol{x}}}\right),$$

- Practical, model-agnostic approach [Hvarfner et al. ICLR'22]

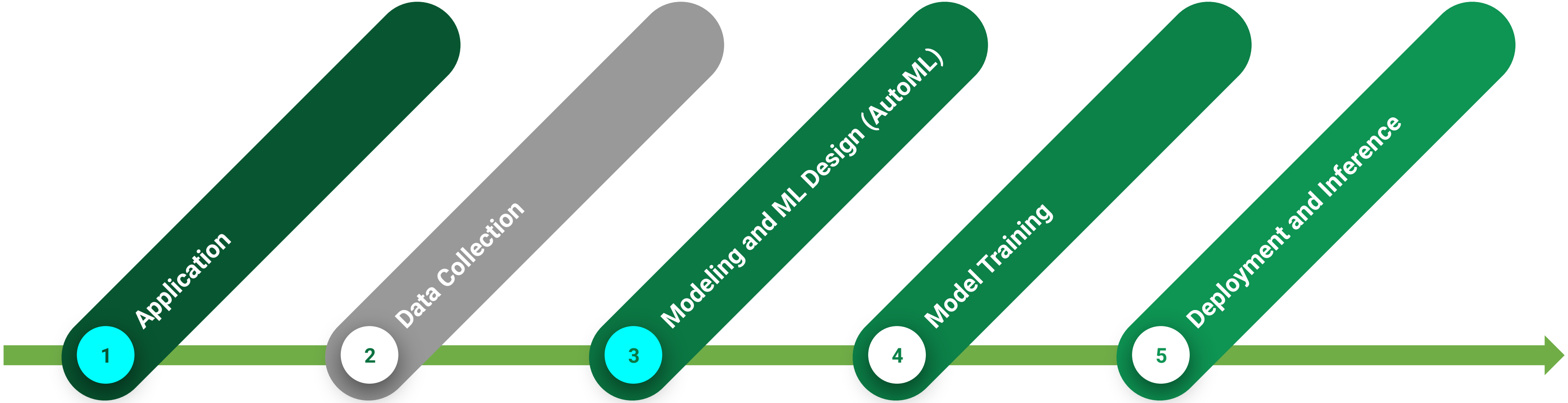
$$\boldsymbol{x}_n \in \underset{\boldsymbol{x} \in \mathcal{X}}{\arg\max}\, \alpha(\boldsymbol{x}, \mathcal{D}_n) \pi(\boldsymbol{x})^{\beta/n}$$

- Expert priors and efficient multi-fidelity optimization [Malik et al. MetaLearn'22, NeurIPS'23]

PiBO [Hvarfner et al. ICLR'22]



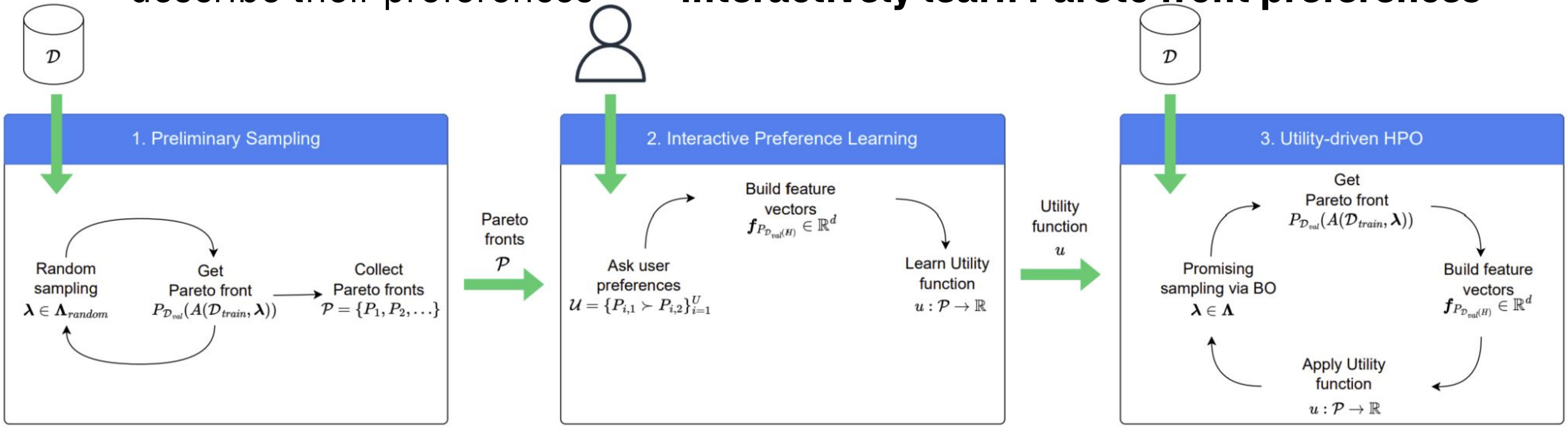
- ➔ Uses expert knowledge to speed up Bayesian Optimization
- ➔ Robust also against wrong believes
- ➔ Substantially speeds up AutoML


1
Application
2
Data Collection
3
Modeling and ML Design (AutoML)
4
Model Training
5
Deployment and Inference

Interactive HPO in Multi-Objective Problems via Preference Learning [Giovanelli et al. 2023]



- Multi-objective (Auto)ML gets more and more important
  - e.g., hardware-aware NAS, fairness-aware AutoML or energy-efficient AutoML
- **Practical challenge**: Different multi-objective indicators lead to different approximated Pareto fronts and users cannot always mathematically describe their preferences $\Rightarrow$ **interactively learn Pareto front preferences**



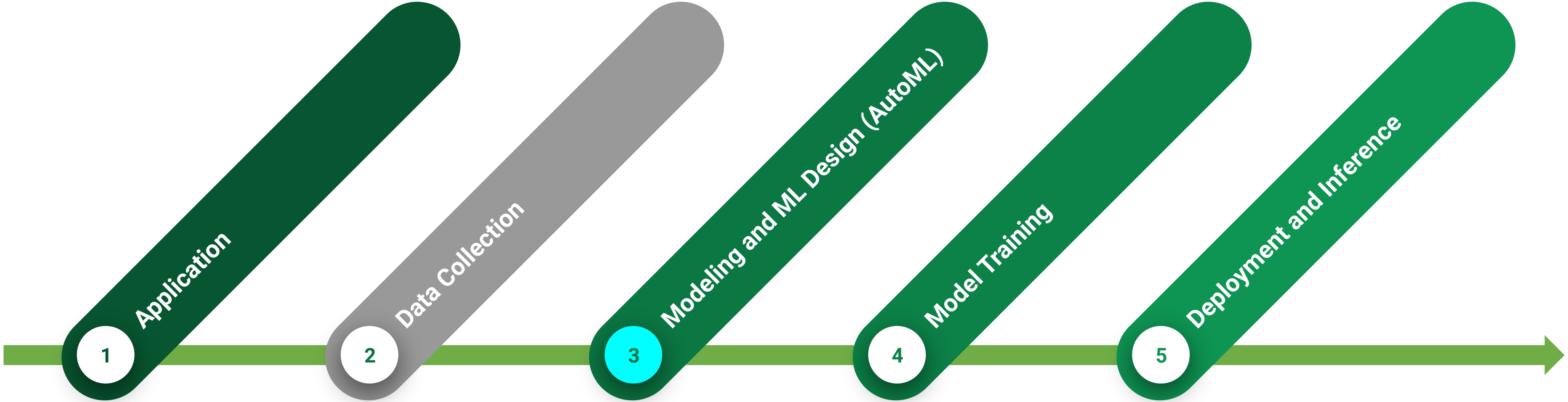
Evaluation of Preference-Learned Indicators

[Giovanelli et al. 2023]

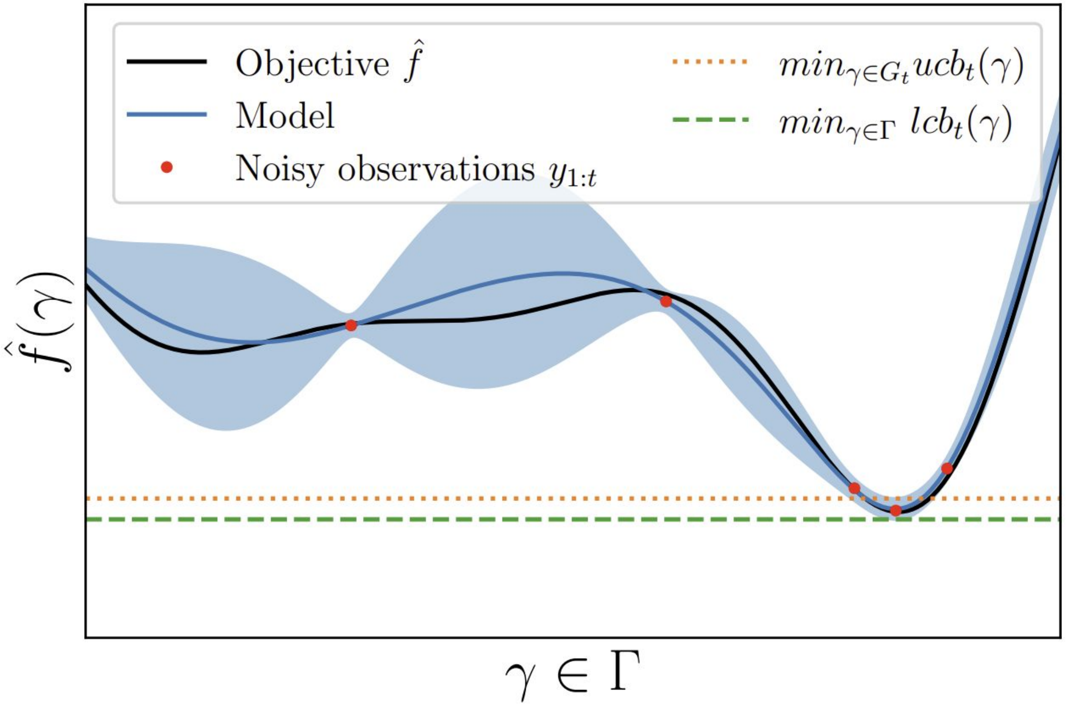


- Benchmark:
  - LCBench
  - Accuracy vs. Energy-Consumption
- Let's assume : User randomly chose a multi-objective (MO) indicator, but was actually hoping for the behavior of another MO indicator
- ⇒ learned preferences are better than randomly choosing a MO indicator

PB\IB	$HV \uparrow$	$SP \downarrow$	$MS \uparrow$	$R2 \downarrow$
$HV \uparrow$	0.76 (±0.17) \ **0.77 (±0.17)**	**0.76 (±0.17)** \ 0.52 (±0.24)	**0.76 (±0.17)** \ 0.52 (±0.21)	0.76 (±0.17) \ **0.77 (±0.16)**
$SP \downarrow$	**0.01 (±0.01)** \ 0.03 (±0.02)	**0.01 (±0.01)** \ **0.01 (±0.0)**	**0.01 (±0.01)** \ 0.04 (±0.03)	**0.01 (±0.01)** \ 0.04 (±0.02)
$MS \uparrow$	**0.61 (±0.09)** \ 0.19 (±0.08)	**0.61 (±0.09)** \ 0.19 (±0.12)	0.61 (±0.09) \ **0.65 (±0.06)**	**0.61 (±0.09)** \ 0.23 (±0.11)
$R2 \downarrow$	0.23 (±0.16) \ **0.22 (±0.16)**	**0.23 (±0.16)** \ 0.47 (±0.23)	**0.23 (±0.16)** \ 0.45 (±0.21)	0.23 (±0.16) \ **0.21 (±0.16)**


1
Application
2
Data Collection
3
Modeling and ML Design (AutoML)
4
Model Training
5
Deployment and Inference

Automatic Termination of Bayesian Optimization for Hyperparameter Optimization
[Markarova et al. 2022]



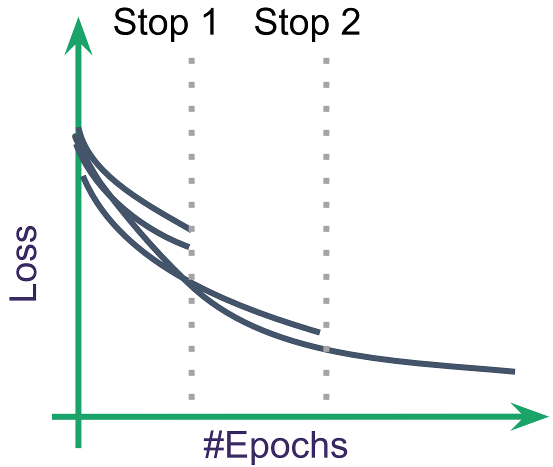
- **Motivation**: Stop HPO if there is likely nothing to gain anymore

  ⇒ *potentially saves a lot of compute resources and energy*

- **Termination of BO**:
  If the uncertainty on the incumbent loss is larger than BO's uncertainty, terminate HPO

  ⇒ **Our extension:** How to adapt to state-of-the-art multi-fidelity optimization?

Automatic Termination of Multi-fidelity HPO
[Graf et al. 2023 – WIP]

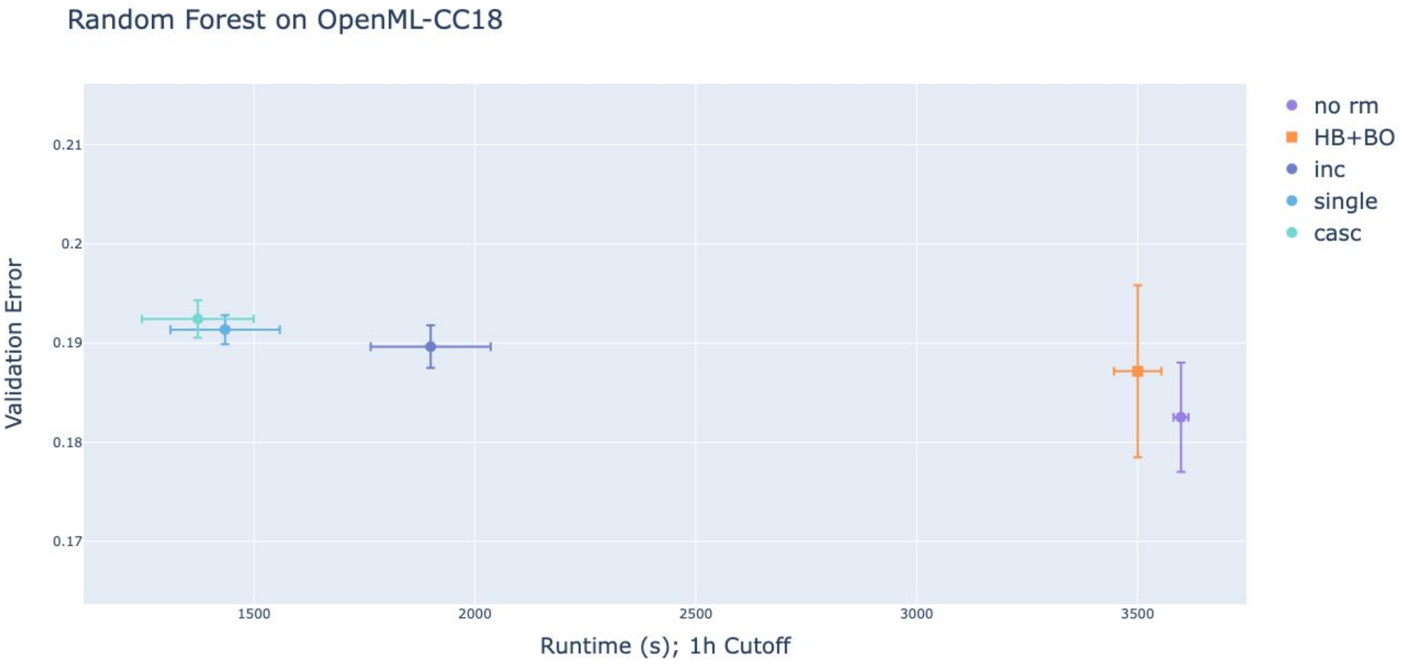


Several design options
for automatic termination:

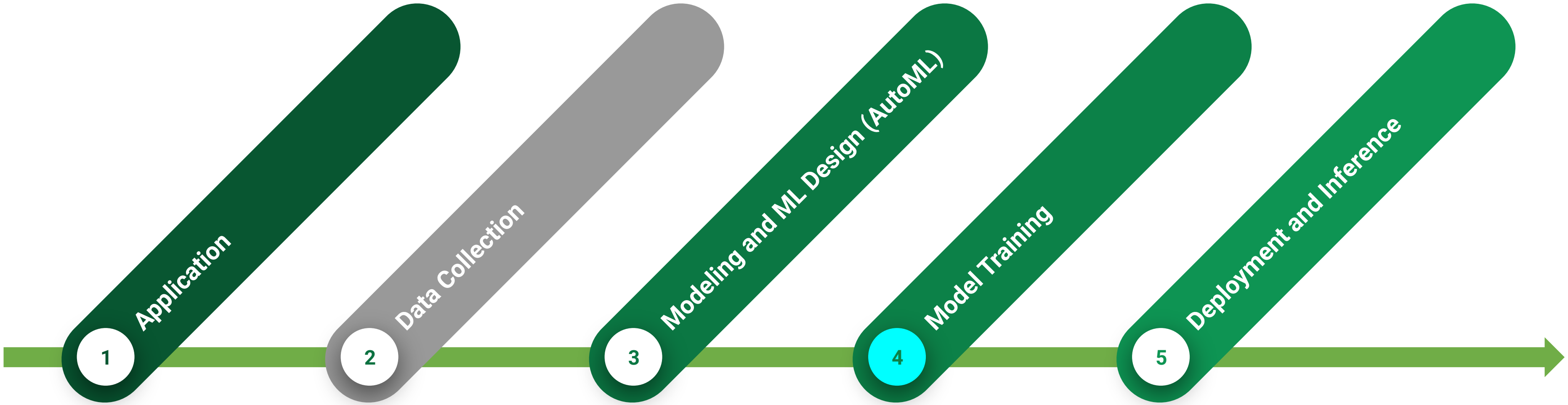
1. Terminate fidelities sequentially or independently?
2. Terminate overall if highest fidelity or all fidelities are terminated?
3. Terminate all lower fidelities if a higher fidelity was terminated?
4. Terminate a fidelity for all subsequent HB runs?

5. Adapt the search space for higher fidelities if lower fidelity was terminated?

Preliminary Results
[Graf et al. 2023 – WIP]

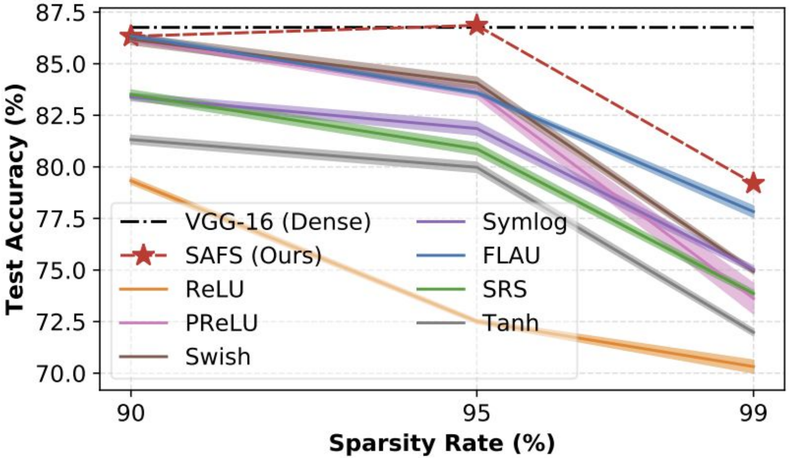


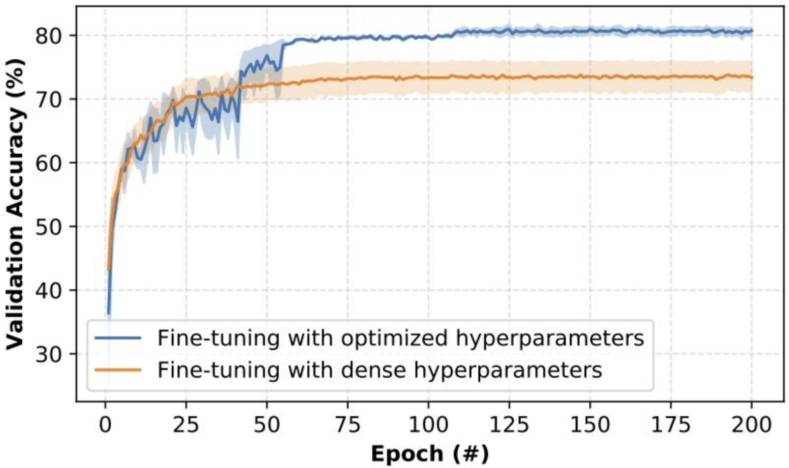
- **HB+BO**: BOHB [Falkner et al. 2018]
- **no rm**: fidelities are terminated once and not permanently
- **inc**: incremental termination of fidelities
- **single**: independent termination of fidelities
- **casc**: lower fidelities will also be removed


1
Application
2
Data Collection
3
Modeling and ML Design (AutoML)
4
Model Training
5
Deployment and Inference

Learning Activation Functions for Sparse Neural Networks [Loni et al. 2023]


- Sparsifying networks can help to save a lot of compute power
- **Insights**:
  1. Using the same activation function class as for the dense network is suboptimal for pruning
  2. Hyperparameters have to adapted accordingly

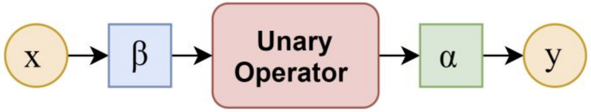




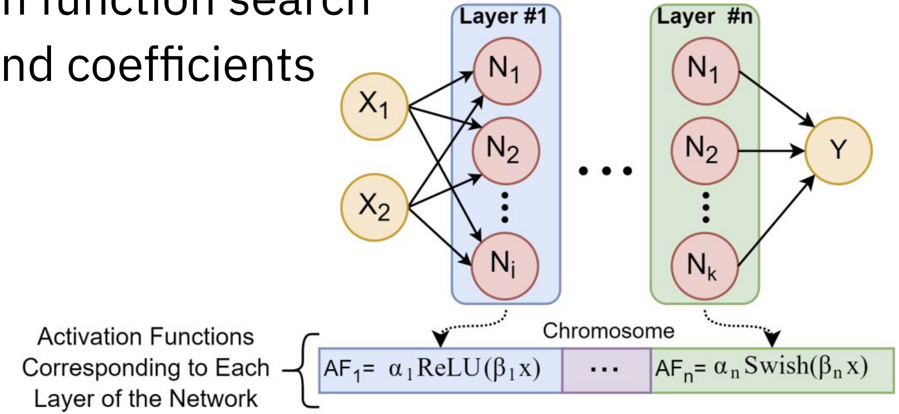
Learning Activation Functions for Sparse Neural Networks [Loni et al. 2023]

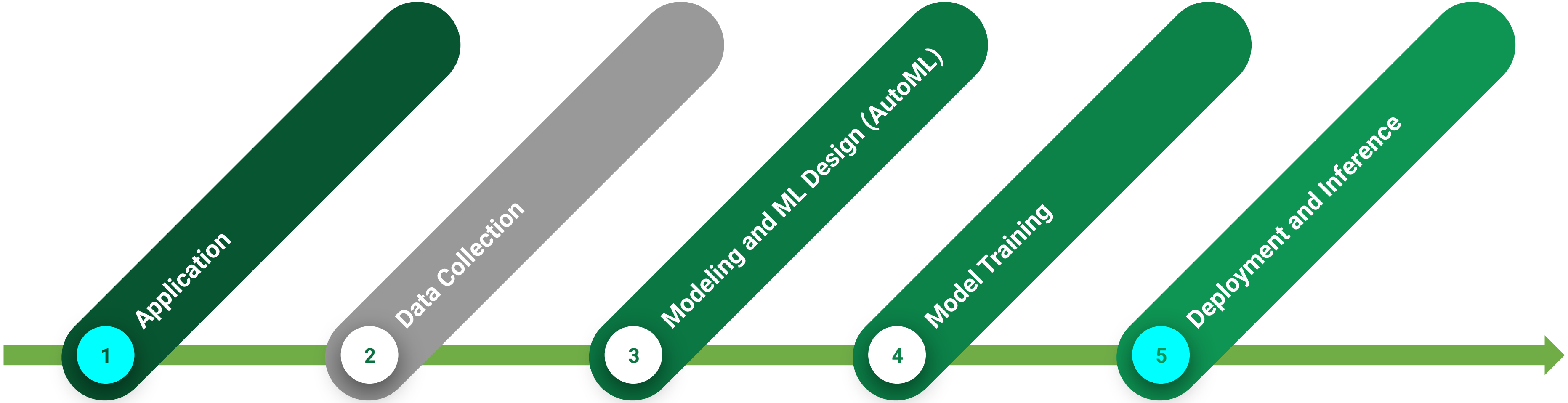
Take-Aways:

- Search for activation functions for the pruning process
- Activation functions should even differ for different layers
  - Symlog and Acon in early layers
  - Swish in middle layers
- Stage 1: Use EA (LAHC) for activation function search
- Stage 2: Apply SGD-based HPO to find coefficients



(a)



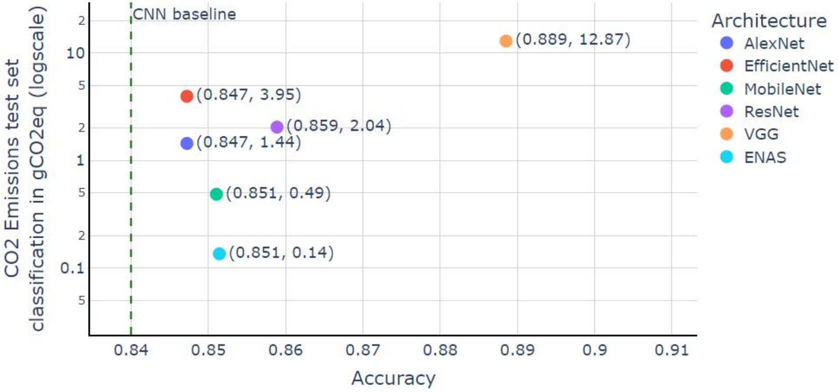

1
Application
2
Data Collection
3
Modeling and ML Design (AutoML)
4
Model Training
5
Deployment and Inference

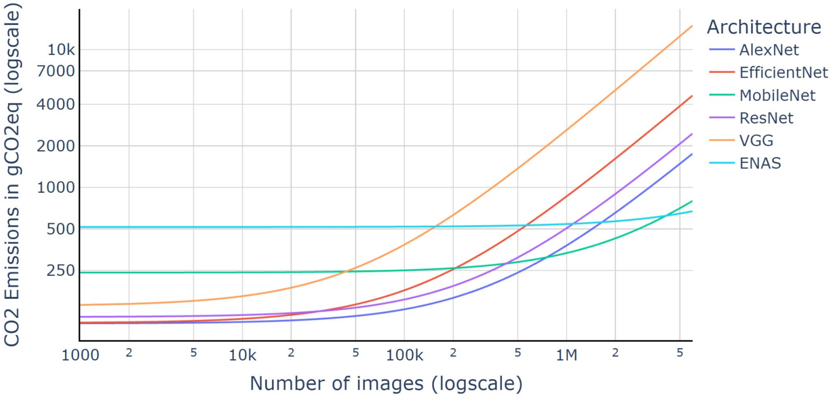
Green AutoML for Plastic Litter Detection

[Theodorakopoulos et al. 2023]

Insights:

1. Architecture of DNNs with better accuracy
2. Architecture with lower CO_2 emissions
3. CO_2 emissions of AutoML training is compensated at inference





Green AutoML [Tornede et al. 2023]

Energy-efficient AutoML
Data compression,
Zero-cost AutoML,
multi-fidelity,
intelligent stopping, …

Searching for Energy-Efficient Models
Model size constraint,
Energy-aware objective functions,
Energy efficient architectures,
Model compression, …

AutoML for Sustainability
Plastic Litter Detection,
Green Assisted Driving,
Predictive Maintenance, …

Create Attention
Tracking emissions,
awareness among students,
researchers, industry partners, …

Challenges Beyond AutoML

1. How to measure the energy consumption of AI correctly?
2. How to translate energy consumption to CO_2e?
3. How to account for the data collection?
4. How to estimate the long term use after model deployment?
5. How to assess the benefits of foundational research?

All resources are finite and we have to be act responsibly.

Find Us

AutoML.org

@AutoML_org

automl

@AutoML_org

LUH|AI

@AIHannover

LUH-AI

@luh-ai

Funded by:

European Research Council
Established by the European Commission

Deutsche Forschungsgemeinschaft

Federal Ministry of Education and Research

Federal Ministry for Economic Affairs and Energy

Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection

Niedersachsen

DAAD

Green AutoML [Tornede et al. 2023]

Energy-efficient AutoML

Data compression,
Zero-cost AutoML,
multi-fidelity,
intelligent stopping, …

Searching for Energy-Efficient Models

Model size constraint,
Energy-aware objective functions,
Energy efficient architectures,
Model compression, …

AutoML for Sustainability

Plastic Litter Detection,
Green Assisted Driving,
Predictive Maintenance, …

Create Attention

Tracking emissions,
awareness among students,
researchers, industry partners, …