Seminar Summer Term 2024

AutoML in the Age of Large Pre-trained Models

Katharina Eggensperger

/KEggensperger
katharina.eggensperger@uni-tuebingen.de
AutoML for Science

April 16th, 2024

Story Line Today

[20min] **Big Picture AutoML**

[?] Your *Questions*

[10min] **Organization**

[?] Your *Questions*

[15min] **Topics/Papers**

[?] Your *Questions*

[30min; if time left; unlikely] **How to give a good presentation**

The Big Picture

>> What is this about?

Machine learning is this …

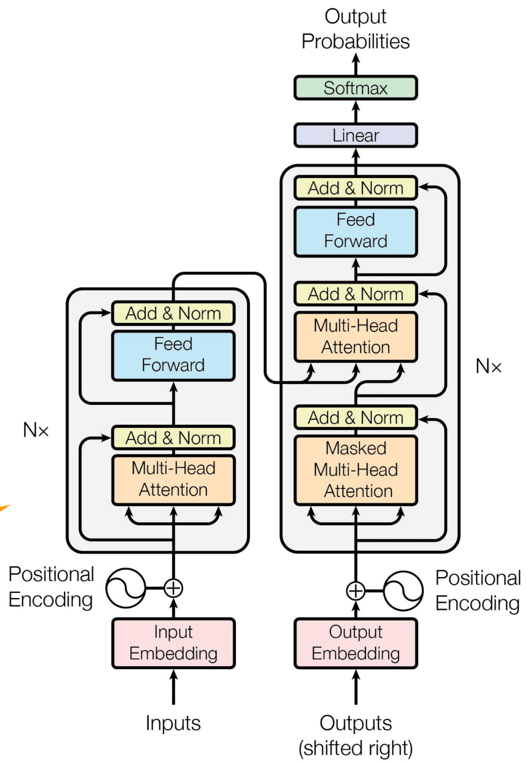
"Machine learning is the science of getting computers to act without being explicitly programmed."

by Andrew Ng
(probably inspired by Arthur Samuels)

… and this

source: XKDC

… and more recently also this



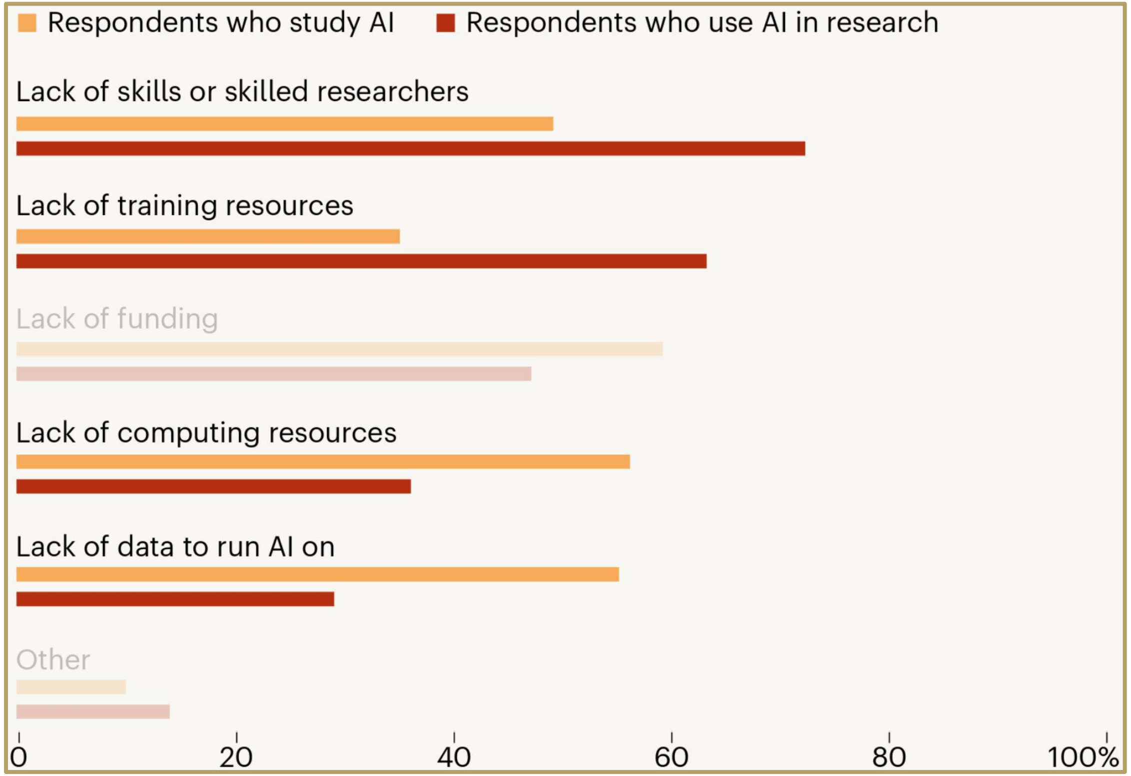
"Attention is all you need" paper by Vaswani, et al., 2017

What can AutoML do for this?

Can we use this for AutoML?

But first: What is AutoML and why do we need it?

Barriers to machine learning



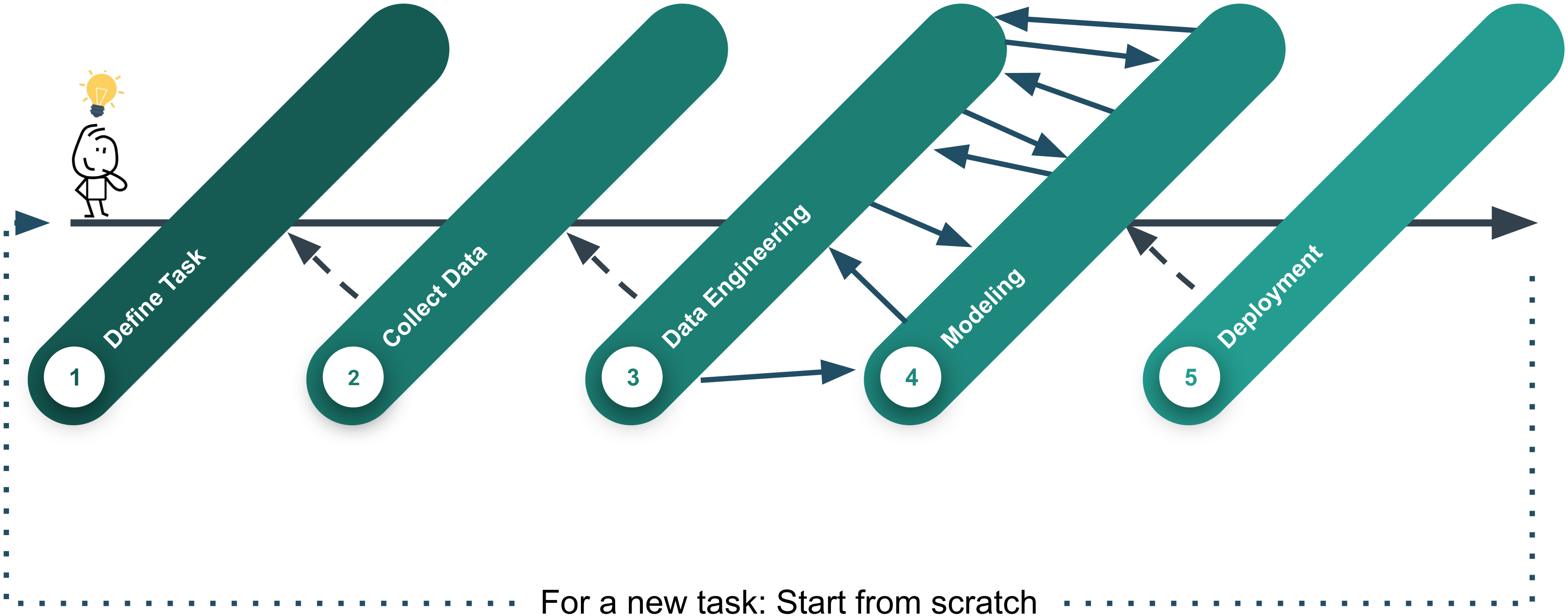
Source: Van Noorden et al. “AI and Science: What 1,600 researchers think” Nature 2023

… also true for industry

Challenges for Applying ML

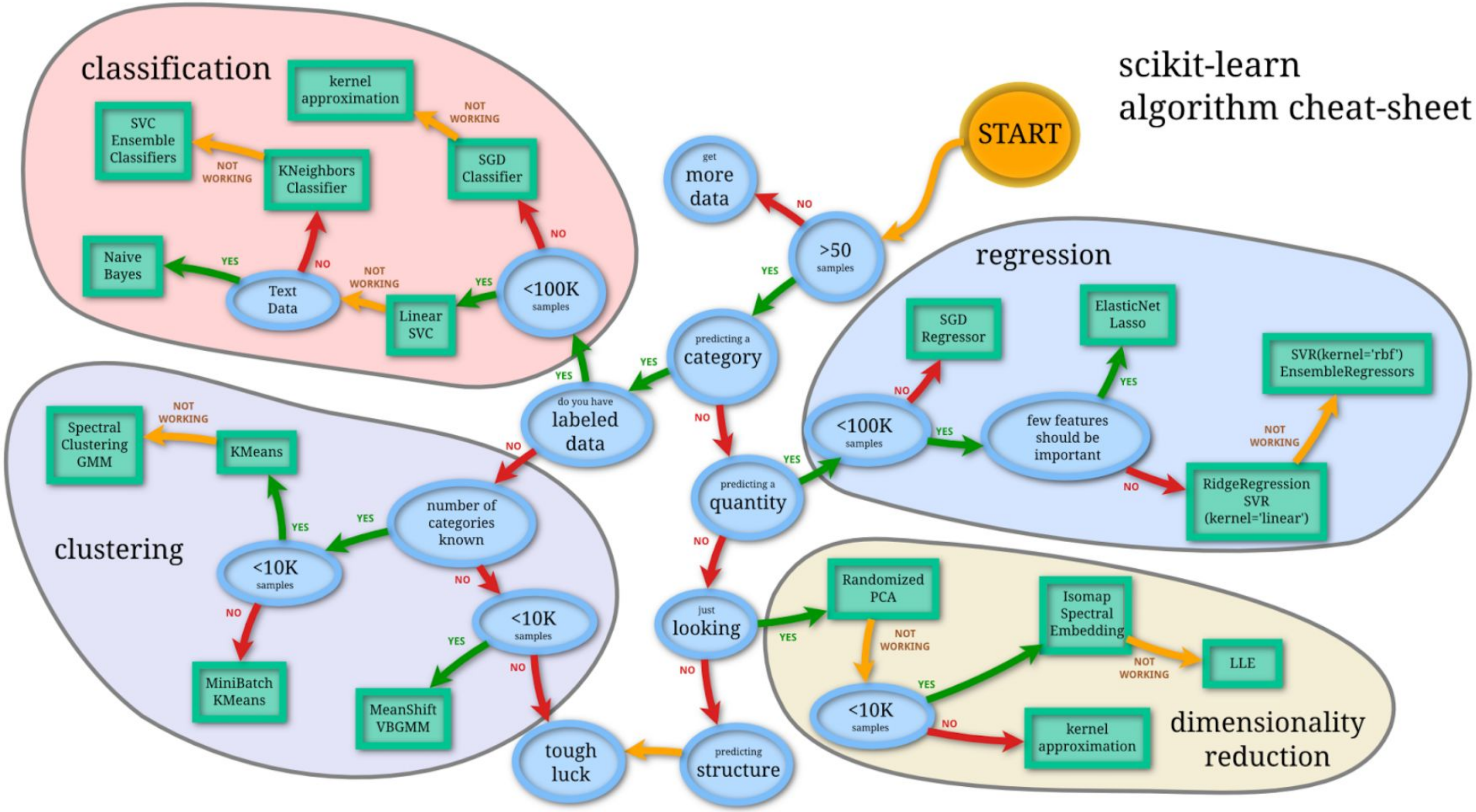
- Required Expertise
- Required Data
- Required Resources

ML workflow



For a new task: Start from scratch

Design Decisions

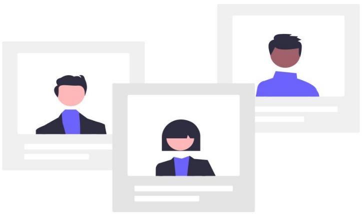


source:
https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

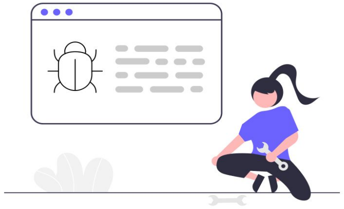
Challenges in Applying AI/ML these days

Required expertise in ML and AI

Long development time for new AI applications

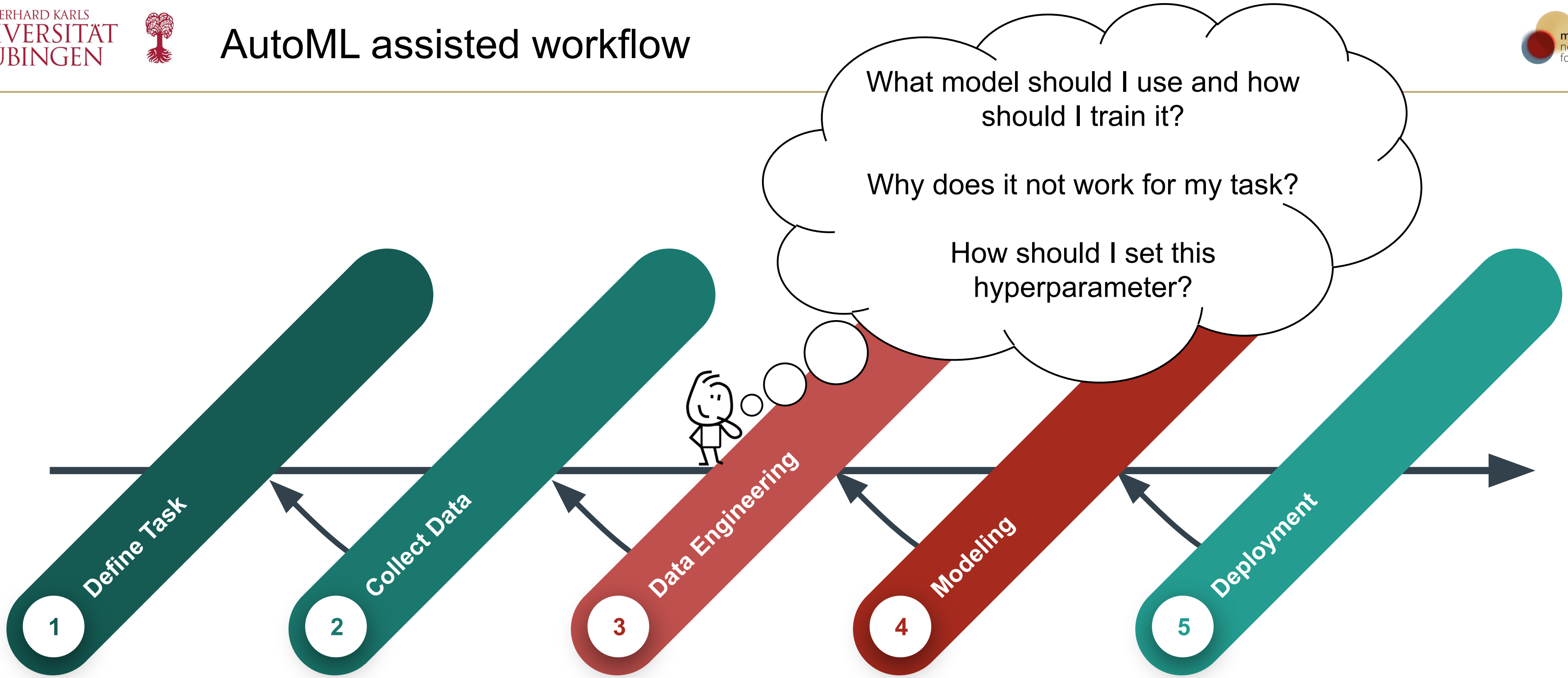


Few experts are available on the job market



Unstructured and error-prone development of AI application

AutoML assisted workflow



Advantages of AutoML

🚀 **Efficient** research and development

→ AutoML can yield state-of-the-art results

🧮 **Systematic** research and development

→ no (human) bias or non-systematic evaluation

🌍 **Broader use** of ML methods

→ less required ML expert knowledge

Challenges For AutoML

🔁 Each dataset requires **different optimal ML-designs**

→ design decisions have to be made for each dataset again

⌛ Training of a single ML model can be **expensive**

→we can not try many configurations

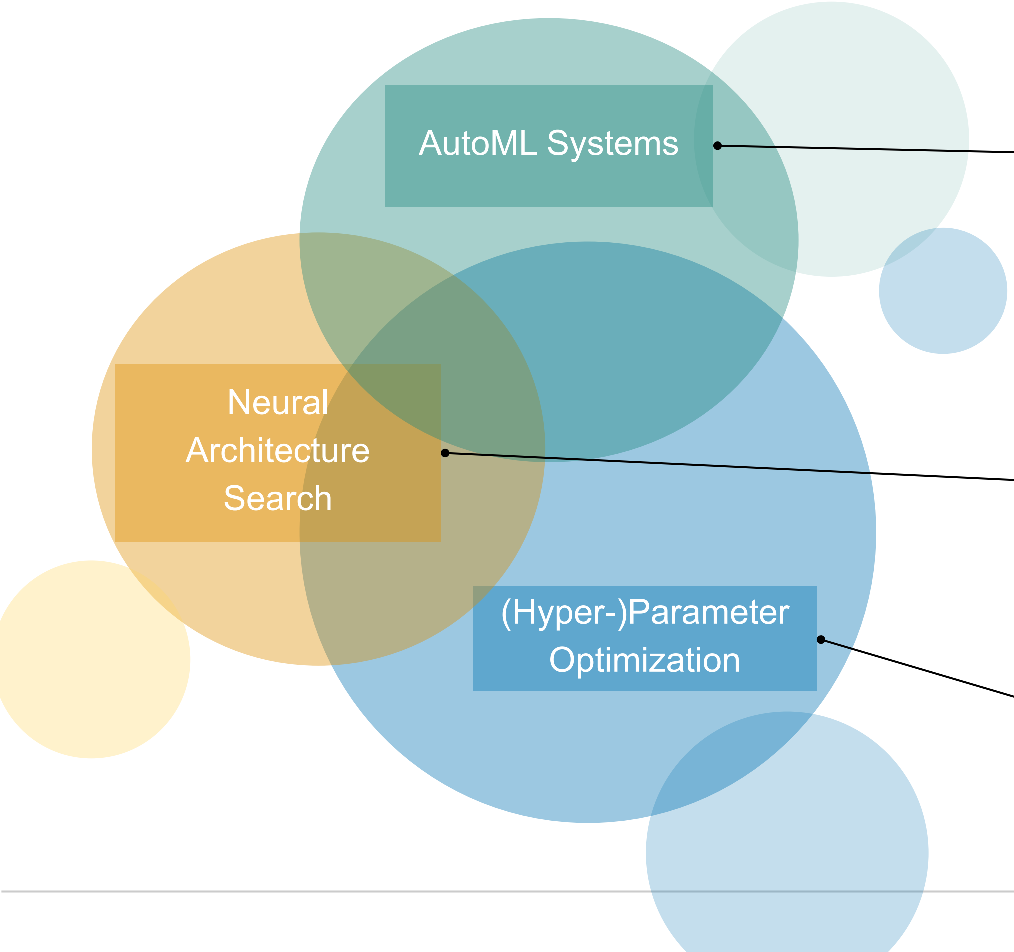
❓Mathematical **relations** are (often) **unknown**

→ gradient-based optimization not easily possible

🎛️ 🎚️ Optimization in **highly complex spaces**

→ including categorical, continuous and conditional dependencies

AutoML Research



Find the best performing **algorithm** and **configuration** given a searchspace and costfunction:

$$(\mathcal{A}^*, \boldsymbol{\lambda}^*) \in \underset{\mathcal{A} \in \mathbf{A}, \boldsymbol{\lambda} \in \boldsymbol{\Lambda}}{\arg\min}\, c(\mathcal{A}_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

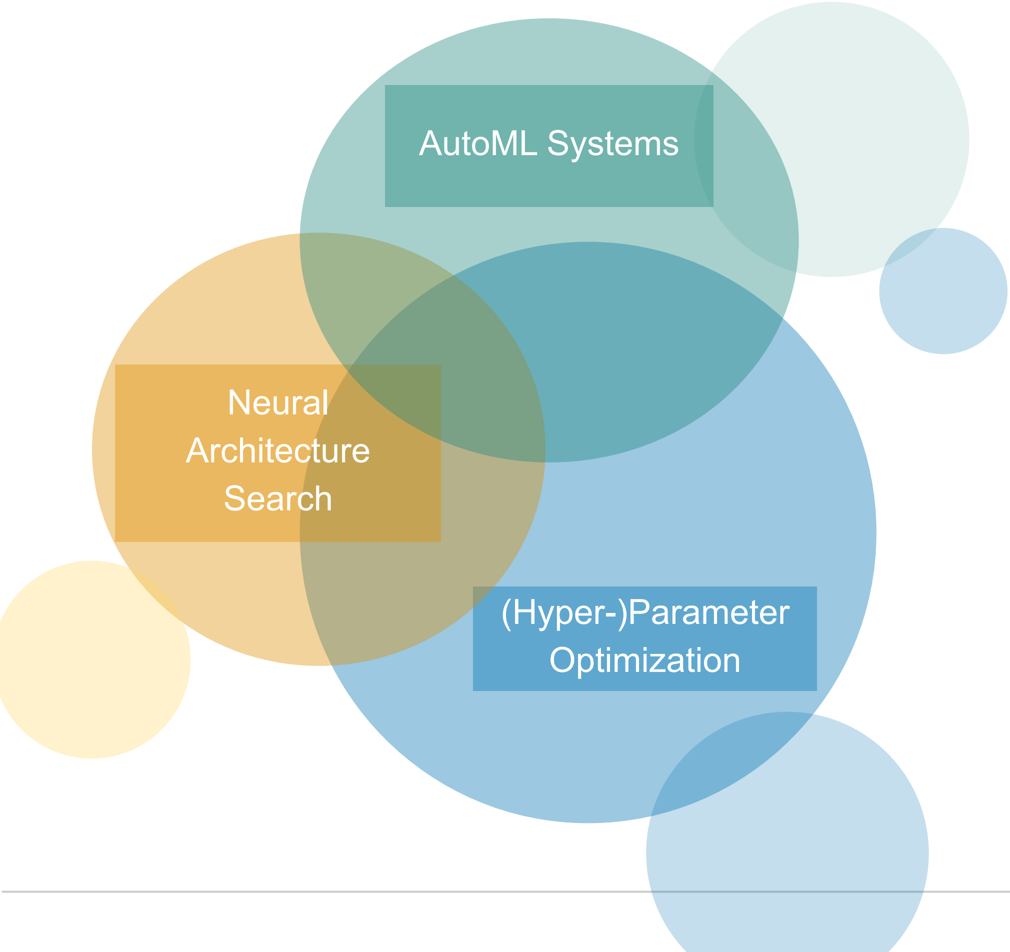
Find the best performing **neural architecture** given a searchspace and costfunction:

$$\boldsymbol{\lambda}^* \in \underset{\boldsymbol{\lambda} \in \boldsymbol{\Lambda}}{\arg\min}\, c(N_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

Find the best performing **configuration** given a searchspace and costfunction:

$$\boldsymbol{\lambda}^* \in \underset{\boldsymbol{\lambda} \in \boldsymbol{\Lambda}}{\arg\min}\, c(\mathcal{A}_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

What can AutoML do for/with large-pretrained models



Have you ever had a HPO / NAS problem?

→ Say hi to your neighbour! (~5min)

AutoML Research: Hyperparameter Optimization

Find the best performing configuration given a searchspace and costfunction:

$$\boldsymbol{\lambda}^* \in \arg\min_{\boldsymbol{\lambda}\in\boldsymbol{\Lambda}} c(\mathcal{A}_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

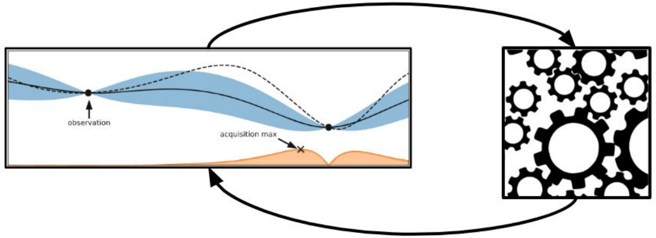
(Hyper-)Parameter Optimization

Methods:

- Bayesian Optimization
- Evolutionary Algorithms
- Speed-up with
  - Transfer-Learning
  - Multi-Fidelity
  - Human-priors

Success Stories:

- Tuning Alpha Go
- W&B, Optuna and more



Session #1, #5 and #6

AutoML Research: Neural Architecture Search

Find the best performing neural architecture given a searchspace and costfunction:

$$\boldsymbol{\lambda}^* \in \arg\min_{\boldsymbol{\lambda} \in \boldsymbol{\Lambda}} c(N_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

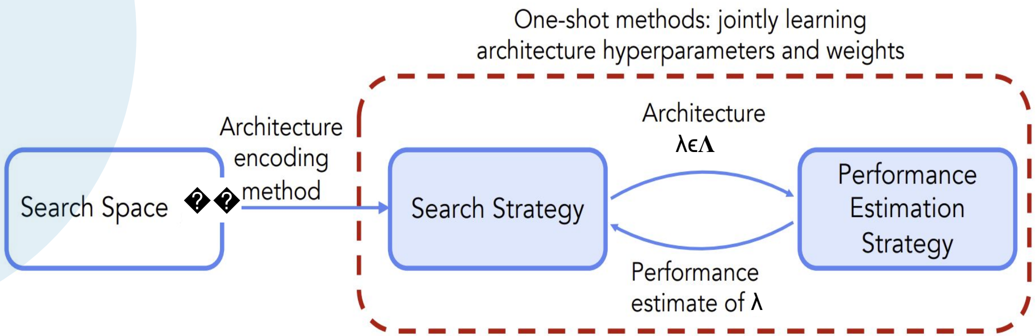
Neural Architecture Search

Methods:

- Black-Box NAS
- One-Shot NAS
- Speed-up:
  - Zero-cost Proxies
  - Performance Estimation

Success Story: Hardware-aware NAS for embedded devices

Note: NAS is a decade-old problem, but mainstream since 2017; probably the most popular AutoML Problem

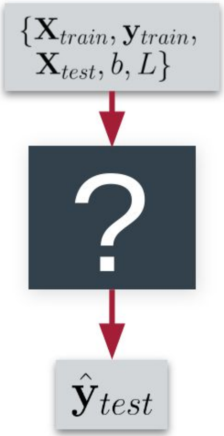


AutoML Research: AutoML Systems

AutoML Systems

Methods:

- HPO and NAS
- Ensembling
- Speed-up:
  - Meta-Learning



Success Stories:

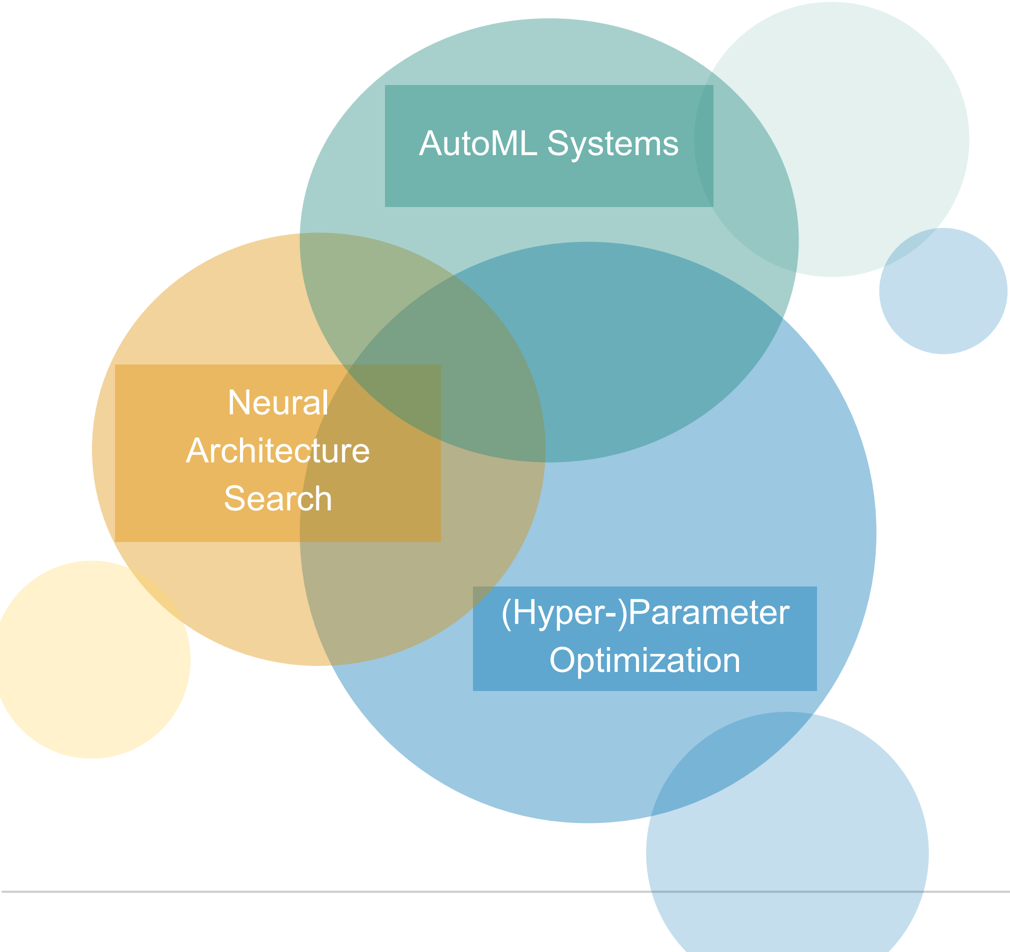
- hundreds of (academic) applications
- many open-source tools

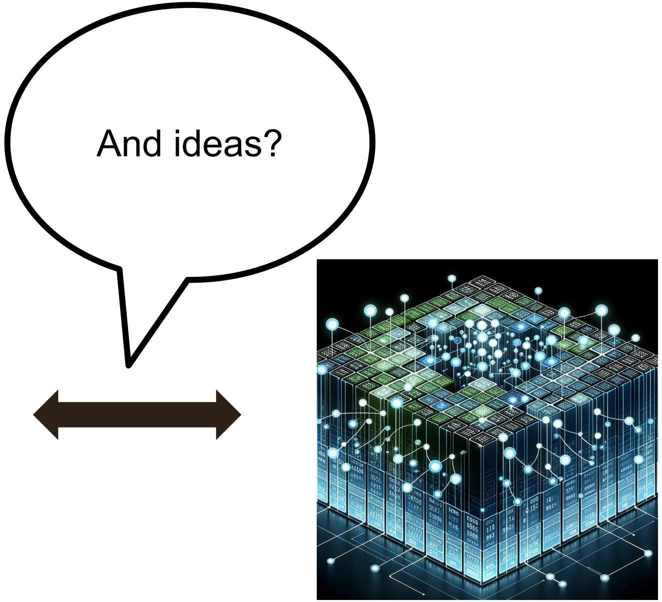
Find the best performing algorithm and configuration given a searchspace and costfunction:

$$(\mathcal{A}^*, \boldsymbol{\lambda}^*) \in \underset{\mathcal{A}\in\mathbf{A}, \boldsymbol{\lambda}\in\boldsymbol{\Lambda}}{\arg\min}\, c(\mathcal{A}_{\boldsymbol{\lambda}}, \mathcal{D}_{train}, \mathcal{D}_{valid})$$

Session #2, #3

What can AutoML do for/with large-pretrained models





generated by OpenAI's DALL-E

Organization

>> When, where and how?

Organizational information

When? Tuesdays 12:00 (c.t.) - 14:00 (actually 12:15-13:45)

Where? MvL6, lecture hall (might change)

Expected Background Knowledge

- Machine Learning
- Deep Learning (this includes transformers)
- (optional) Practical experience with ML/DL

Grades: Presentation / Slides / Report / Participation

Note: You're expected to participate in all sessions

All info is on my website: https://keggensperger.github.io/teaching/2024-summer-seminar

🎁 Features of this seminar

Main Features:

- Get a **broad overview** of current methods in AutoML
- Understand **challenges for research** in AutoML
- Learn how AutoML interacts **with large pre-trained models**

Bonus Features:

- **A lot of focus on interaction!** Active participation and reading the papers beforehand is necessary.
  → You will gain a much better understanding

- **Feedback survey after each presentation**: Anonymous feedback from everyone after the presentation.
  →Learn what others like about your presentation and how to improve

- **Give and receive feedback from a fellow student**: Everyone gets a *study buddy* to practice the presentation and discuss open questions. To get feedback from me, you should have talked to your study buddy (and incorporated their feedback)
  → Higher quality of presentations

The ideal timeline (for the presenter)

> -14d	-7d	-2d	0	<+2d	end of term
Read Paper	**Prepare Presentation**	**Get Feedback**	**Presentation & Feedback**	**Send Slides**	**Send Report**
Read and understand your topic (also take into account further info, e.g., videos, blogposts, posters, slides, reviews)	Also get feedback from your study buddy	(optional) Get Feedback from me, e.g., Fridays 10-12	Enjoy!		

The presentation

Your presentation lasts ~40 minutes and should consist of:

- **20 minutes presentation**
  - introduction to the topic
  - summary of your paper: methods & experiments
  - conclusion: strengths / weaknesses

  → see also separate presentation on “How to give a good presentation”

- **20 minutes discussion**
  - clarification questions
  - additional content
  - a short quiz
  - prepared discussion questions

The report

3-5 pages (format TBD; excluding references) covering the following

A **review** in the context of AutoML including (1-2 pages)

- Motivation: Why does this method matter for AutoML?
- 3 strengths
- 3 weaknesses
- 3 questions you would ask the authors (can be questions that came up during your discussion)

An **assessment** of an AI-generated summary (1-2 pages)

- What’s wrong?
- What’s correct?
- What’s missing

A list with **further** material that you’ve used, e.g. (0.5 page)

- Code
- Public reviews
- Blogposts
- Video Tutorials

Questions?

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

machine learning new perspectives for science

Dates Overview

16.04.2024	Organization
23.04.2024	Intro AutoML I
30.04.2024	Intro AutoML II (if needed)
07-21.05.2024	no meeting

Large Pre-Trained Models for AutoML

28.05.2024	#1 Bayesian Optimization (OptFormer;LLM4BO)
04.06.2024	no meeting
11.06.2024	#2 Tabular Data (TabPFN;XTab)
18.06.2024	#3 Data Science (CAAFEE;MLAgent)
25.06.2024	no meeting
02.07.2024	#4 Neural Architecture Search (GPT4NAS;GPT-NAS)
09.07.2024	no meeting

AutoML for Large Pre-Trained Models

16.07.2024	#5 FineTuning (QuickTune;EvoPrompt;Bandits4LLMs)
23.07.2024	#6 ScalingLaws (TensorV;ScalingLaws4BO)

Session schedule:

x 2

1. 20min Presentation
2. 20min Q&A
3. 5min Feedback

Large models 4 AutoML

#1 Session: Bayesian Optimization

Bayesian Optimization is the basis for most hyperparameter tuning methods.
Can we improve it with LLMs?

1. [OptFormer] Chen et al. Towards learning universal hyperparameter optimizers with transformers (NeurIPS'22)
2. [LLM4BO] Liu et al. Large language models to enhance Bayesian optimization (ICLR'24)

#2 Session: DL for Tabular Data

Tabular data is the most common data type and offers the most diverse landscape of ML solutions.
Can we use one transformers instead?

1. [TabPFN] Hollman et el. TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second (ICLR'23)
2. [XTab] Zhu et al. XTab: Cross-table Pretraining for Tabular Transformers (ICML'23)

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Large models 4 AutoML (cont'd)

machine learning new perspectives for science

#3 Session: Data Science

Data Science contains many opportunities for AutoML.
Can LLMs help here?

1. [CAAFE] Hollman et al. Large Language Models for Automated Data Science: Introducing CAAFE for Context-Aware Automated Feature Engineering (NeurIPS'24)
2. [MLAgent] Huang et al. Benchmarking Large Language Models as AI Research Agents (arxiv'23)

#4 Session: Neural Architecture Search

NAS is arguably the largest research area of AutoML (by #papers).
Can we find better architectures using transformers?

1. [GPT4NAS] Zheng et al. Can GPT-4 Perform Neural Architecture Search? (arxiv'23)
2. [GPT-NAS] Yu et al. GPT-NAS: Evolutionary Neural Architecture Search with the Generative Pre-Trained (arxiv'22)

#5 Session: Using LLMs

Finetuning and Prompt-Engineering are novel challenges in the age of LLMs. Can we use AutoML for this?

1. [QuickTune] Arango et al. Quick-Tune: Quickly Learning Which Pretrained Model to Finetune and How (ICLR'24)
2. [EvoPrompt] Guo et al. Connecting Large Language Models with Evolutionary Algorithms Yields Powerful Prompt Optimizers (ICLR'24)
3. [Bandits4LLMs] Xia et al. Which LLM to Play? Convergence-Aware Online Model Selection with Time-Increasing Bandits (WWW'24)

#6 Session: Scaling Laws

Building large models is extremely expensive. How can we approach optimizing such large models?

1. [TensorV] Yang et al. Tuning large neural networks via zero-shot hyperparameter transfer (NeurIPS'21)
2. [ScalingLaws4BO] Kadra et al. Scaling Laws for Hyperparameter Optimization (Neurips'23)

Questions?

What’s next?

`>> What should I do now?`

Your Next Task?

Do you want to participate?
Is the topic interesting for you? Do you have enough background? Are you available on all dates?

No → Please cancel your registration on ILIAS.

Yes → Make sure you're registered on ILIAS

Link to survey will be sent to everyone registered on ILIAS on Friday noon.

→ Fill out the survey and pick 5 papers

Assignment will be done based on the time you registered on ILIAS/the time you filled out the survey

→ Assignment will be done based on the time you registered on ILIAS / when you filled out the survey

https://keggensperger.github.io/teaching/2024-summer-seminar

Deadlines:

- **Friday, 19.04.2024 (noon)**
  - Register on ILIAS
- **Monday, 22.04.2024, (noon)**
  - Fill out survey

→ You will hear back from me **before next Tuesday**