All AI models are wrong, but some are useful … for power systems

08 April 2025, cresROAD, CRESYM

Dr. Jochen L. Cremer
Associate Professor
www.jochen-cremer.com

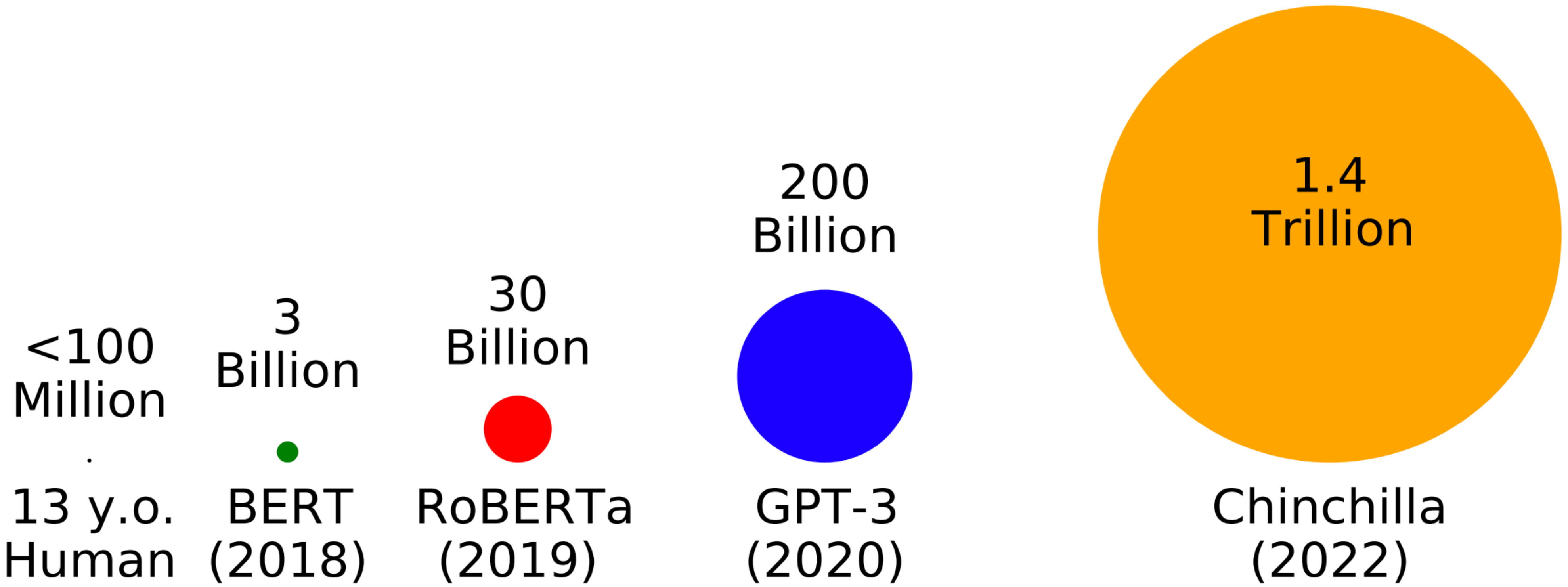
Join at: vevox.app ID: 154-454-820

Why not yet a technology breakthrough with AI in power systems?

lack of open data
unsuitable nn's
not reliable
lack of data
high risk
return on investment
limits to r&d funds
high uncertainty
lacks explainability
not accurate 100%
lack of trust
a bit complex
low trl
risk adverse
data challenges
statistical errors
conservatism
grid structure
outages rare events
data is weird
ps are quite complex
poor data
conservative/safe

TUDelft

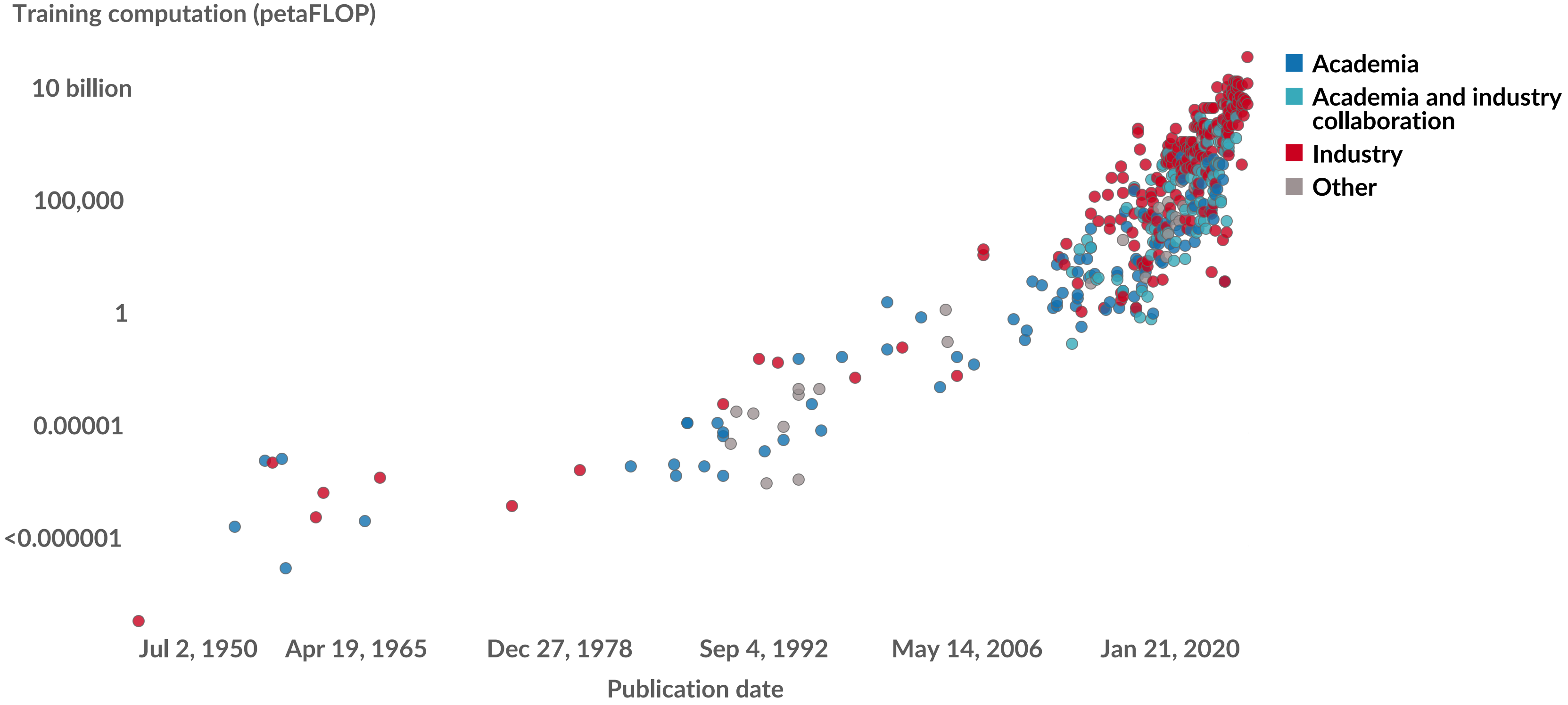
Is more and more data the answer?



Number of tokens seen during training (babylm.github.io)

Computation used to train notable AI systems, by affiliation of researchers

Computation is measured in total petaFLOP, which is 10^{15} floating-point operations[1] estimated from AI literature, albeit with some uncertainty. Estimates are expected to be accurate within a factor of 2, or a factor of 5 for recent undisclosed models like GPT-4.



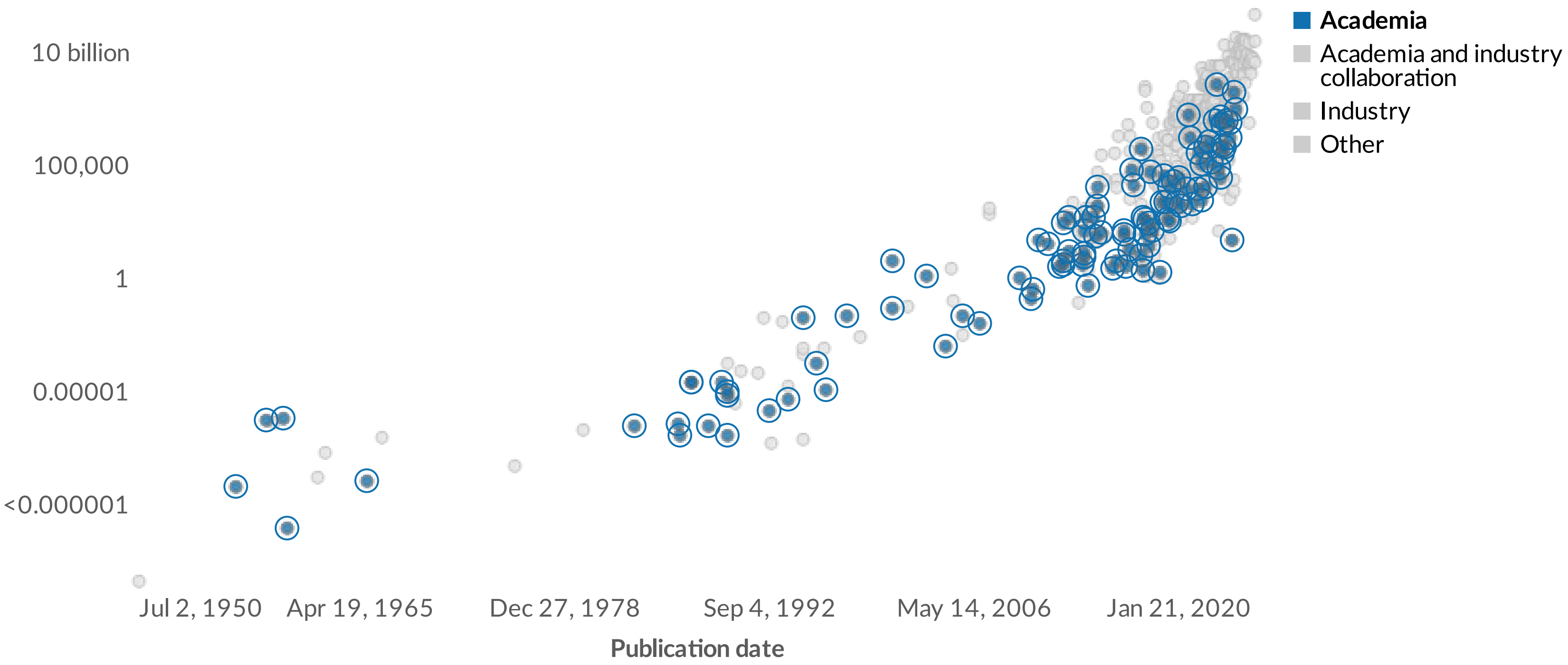
Data source: Epoch (2024)

Note: The Executive Order on AI refers to a directive issued by President Biden on October 30, 2023, aimed at establishing guidelines and standards for the responsible development and use of artificial intelligence within the United States.

1. Floating-point operation: A floating-point operation (FLOP) is a type of computer operation. One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Computation used to train notable AI systems, by affiliation of researchers

Computation is measured in total petaFLOP, which is 10^{15} floating-point operations[1] estimated from AI literature, albeit with some uncertainty. Estimates are expected to be accurate within a factor of 2, or a factor of 5 for recent undisclosed models like GPT-4.



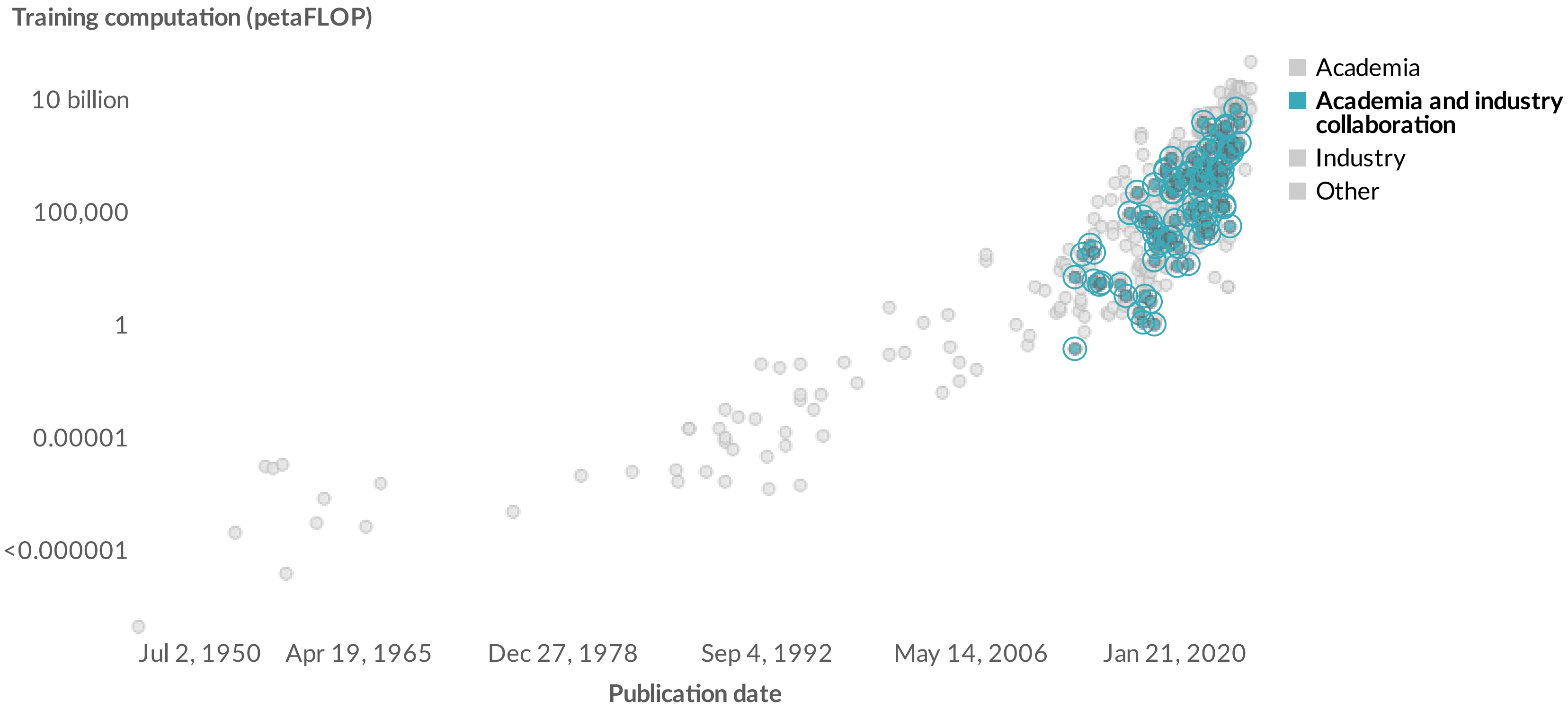
Data source: Epoch (2024)

Note: The Executive Order on AI refers to a directive issued by President Biden on October 30, 2023, aimed at establishing guidelines and standards for the responsible development and use of artificial intelligence within the United States.

1. **Floating-point operation**: A floating-point operation (FLOP) is a type of computer operation. One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Computation used to train notable AI systems, by affiliation of researchers

Computation is measured in total petaFLOP, which is 10^{15} floating-point operations[1] estimated from AI literature, albeit with some uncertainty. Estimates are expected to be accurate within a factor of 2, or a factor of 5 for recent undisclosed models like GPT-4.



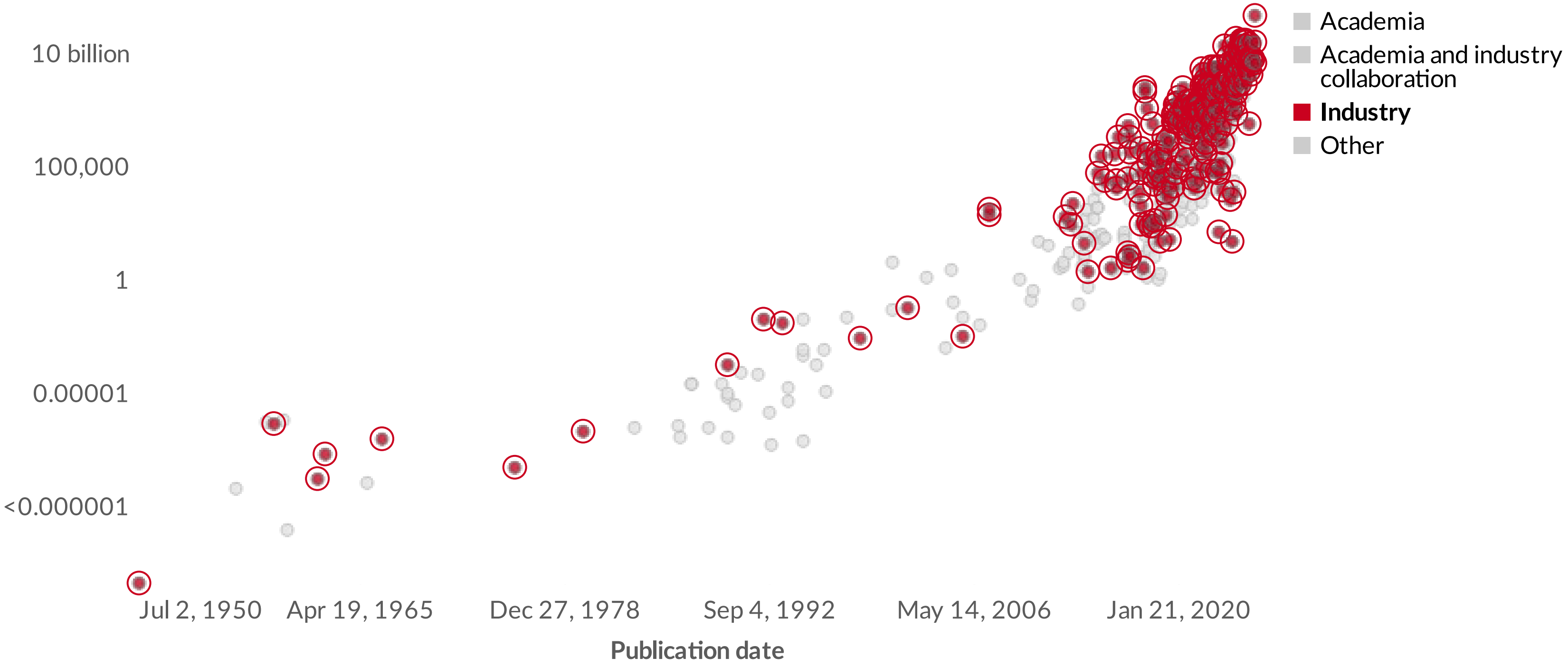
Data source: Epoch (2024)

Note: The Executive Order on AI refers to a directive issued by President Biden on October 30, 2023, aimed at establishing guidelines and standards for the responsible development and use of artificial intelligence within the United States.

1. **Floating-point operation**: A floating-point operation (FLOP) is a type of computer operation. One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Computation used to train notable AI systems, by affiliation of researchers

Computation is measured in total petaFLOP, which is 10^{15} floating-point operations[1] estimated from AI literature, albeit with some uncertainty. Estimates are expected to be accurate within a factor of 2, or a factor of 5 for recent undisclosed models like GPT-4.



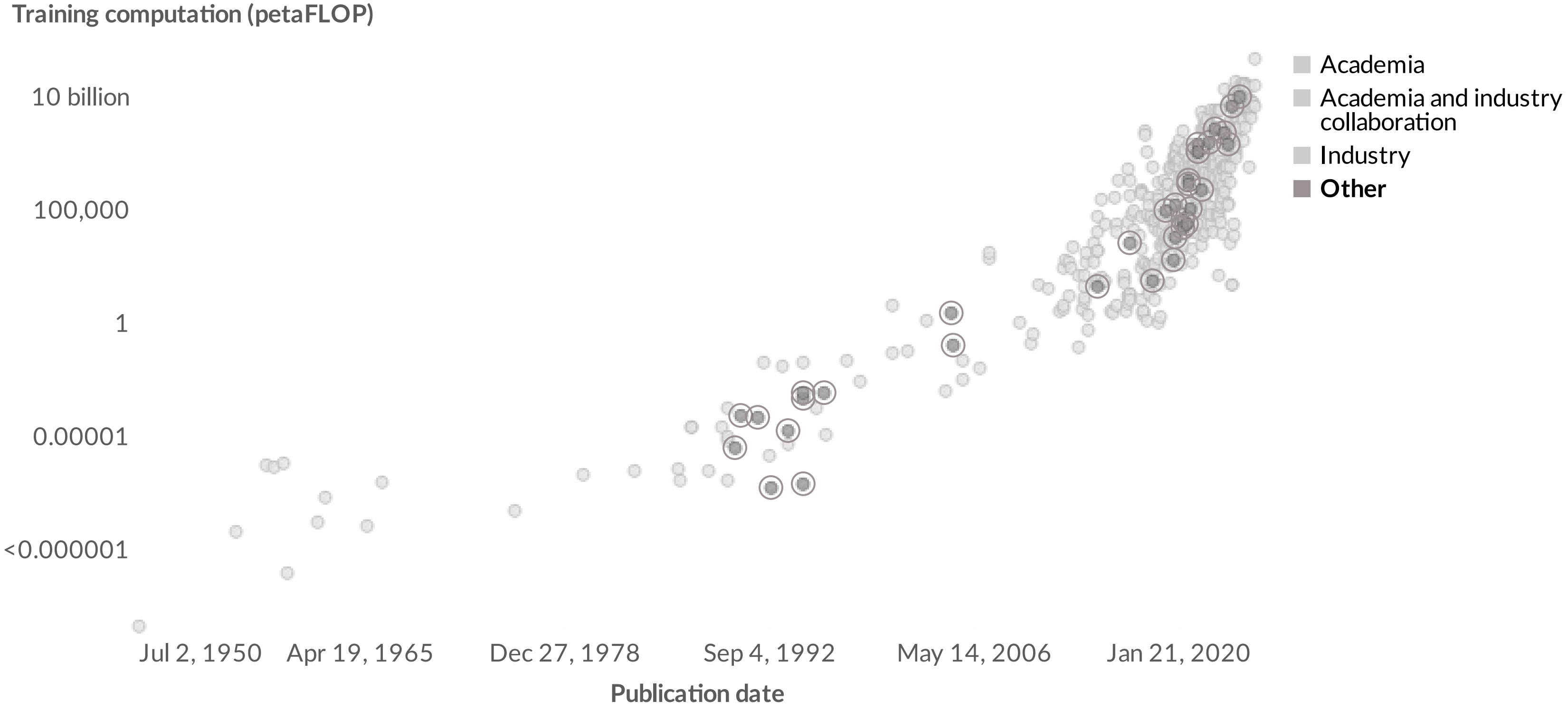
Data source: Epoch (2024)

Note: The Executive Order on AI refers to a directive issued by President Biden on October 30, 2023, aimed at establishing guidelines and standards for the responsible development and use of artificial intelligence within the United States.

1. **Floating-point operation**: A floating-point operation (FLOP) is a type of computer operation. One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Computation used to train notable AI systems, by affiliation of researchers

Computation is measured in total petaFLOP, which is 10^{15} floating-point operations[1] estimated from AI literature, albeit with some uncertainty. Estimates are expected to be accurate within a factor of 2, or a factor of 5 for recent undisclosed models like GPT-4.



Data source: Epoch (2024)

Note: The Executive Order on AI refers to a directive issued by President Biden on October 30, 2023, aimed at establishing guidelines and standards for the responsible development and use of artificial intelligence within the United States.

1. **Floating-point operation**: A floating-point operation (FLOP) is a type of computer operation. One FLOP represents a single arithmetic operation involving floating-point numbers, such as addition, subtraction, multiplication, or division.

Supervised Learning for Surrogate Models

Notation: Power system s, model m, parameter x

Objective: assess $m(x) \rightarrow y$ **<u>very</u>** fast and **<u>often</u>**

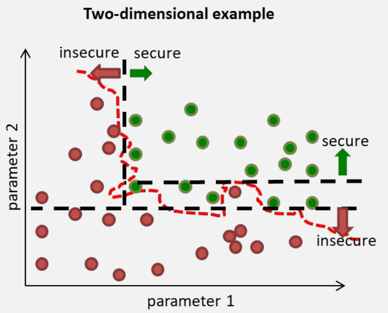
Surrogate approach

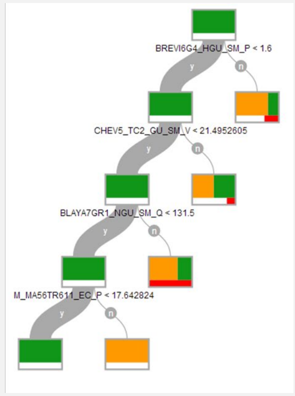
1. Generate a training dataset $\Omega^T = \{(x_i, y_i)\}_{i=1}^{N}$ where $y_i = m(x_i)$ from the full simulator
2. Train surrogate $f(x) \rightarrow \hat{y}$ with supervised loss $\sum_{i \in \Omega^T} ||y_i - \hat{y}_i||$
3. Use $f(x_j)$ for new j $\notin \Omega^T$

Benefit: speed at inference

Applications

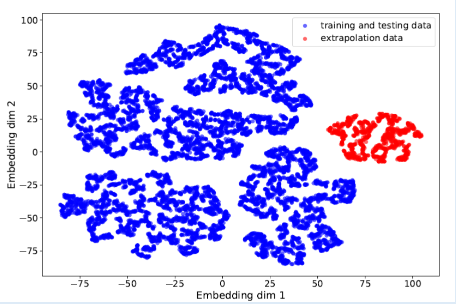
- Real-time dynamic security assessment ([1,2] and many others)





Challenges

- Out of distribution risks: What if s and m changes? e.g., topology changes
- What if the model is inaccurate $s \neq m$? e.g., inverter-based controls
- Need large, representative training data



[1] Wehenkel, Louis, and Mania Pavella. "Decision trees and transient stability of electric power systems." *Automatica* 27.1 (1991): 115-134.
[2] Cremer, Jochen L., Ioannis Konstantelos, and Goran Strbac. "From optimization-based machine learning to interpretable security rules for operation." *IEEE Transactions on Power Systems* 34.5 (2019): 3826-3836.
[3] Olayiwola Arowolo, Jochen Stiasny, Jochen L. Cremer, "Exploring Extrapolation of Machine Learning Models for Power System Time Domain Simulation", *Bulk Power System Dynamics and Control Symposium, 2025*

Physics-Informed Learning

Objective: surrogate learning enhanced with physics knowledge from model m

Idea: Incorporate physics residual (e.g. from a PDE or simulator) to geode learning and improve generalization

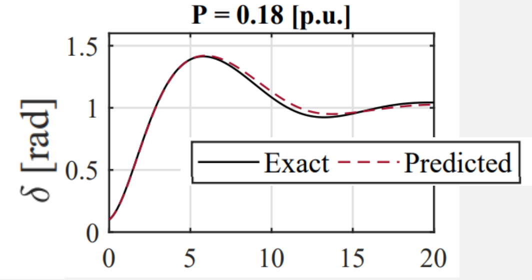
Physics-informed approach

1. Generate offline training dataset $\Omega^T = \{(x_i, y_i)\}_{i=1}^N$ with $y_i = m(x_i)$
2. Train surrogate $f(x) \rightarrow \hat{y}$ on composite loss $\sum_{i\in\Omega^T} \|y_i - \hat{y}_i\| + \mathcal{L}_{phys}\,(f(x_i), m)$
3. Use $f(x_j)$ for new j $\notin \Omega^T$

Benefits: Better generalisation performance with **fewer training samples**

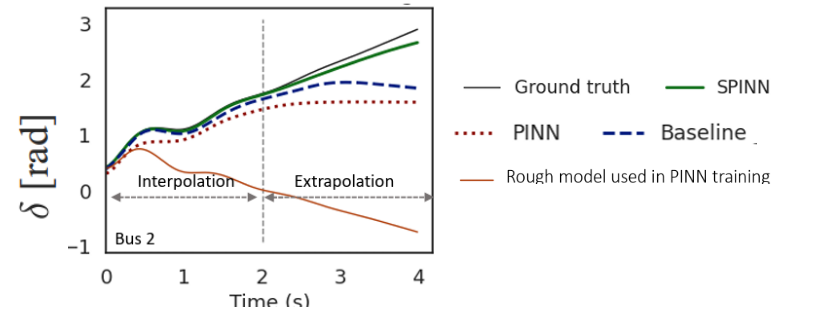
Applications

- Extrapolation in time-domain for dynamic analysis in power systems



Challenges

- Model inaccuracy $s \neq m$
- Changes in s or m
- Data sparsity
- Multi-loss scaling causes training instability
- Scaling issues to many physical loss terms in power systems



[4] Misyris, George S., Andreas Venzke, and Spyros Chatzivasileiadis. "Physics-informed neural networks for power systems." In *2020 IEEE power & energy society general meeting (PESGM)*, pp. 1-5. IEEE, 2020.
[5] Xie, Haiwei, Federica Bellizio, Jochen L. Cremer, and Goran Strbac. "Regularised Learning with Selected Physics for Power System Dynamics." In *2023 IEEE Belgrade PowerTech*, pp. 1-7. IEEE, 2023.

Weakly-Supervised (E2E) Learning

Objective: learn models $f(x)$ for downstream task even when exact labels $y_i = m(x_i)$ from the simulator m are unavailable, uncertain, or only indirectly defined.

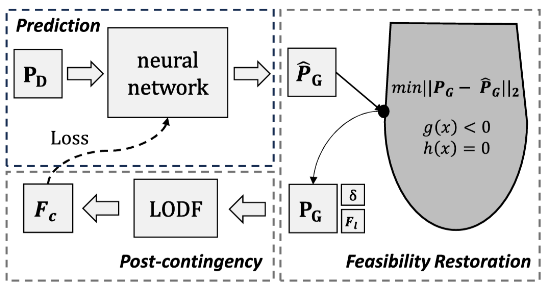
Approach

1. Generate many inputs $\Omega^T = \{(x_i)\}_{i=1}^N$
2. Model task loss $\sum_{i \in \Omega^T} \mathcal{L}(m(f(x_i))$
3. Use $f(x_j)$ for new j $\notin \Omega^T$

Benefits: learning for computationally expensive or ill-defined problems

Applications

- Learn to predict effective inputs to OPF[6]
- Replace conventional solvers with NN [7]
- Distribution system state estimation [8]
- N-k security constrained OPF [9]



Challenges

- Inexact supervision $s \neq m$ not so important as success defined by task-loss
- System shift in s or m
- Data coverage. Diverse samples are needed for generalization

TU Delft

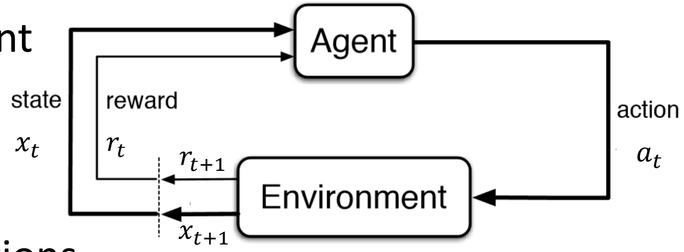
[6] Donti, P., Amos, B., & Kolter, J. Z. (2017). Task-based end-to-end model learning in stochastic optimization. *Advances in neural information processing systems*, *30*.
[7] Donon, B., Liu, Z., Liu, W., Guyon, I., Marot, A., & Schoenauer, M. (2020). Deep statistical solvers. *Advances in Neural Information Processing Systems*, *33*, 7910-7921.
[8] Habib, B., Isufi, E., van Breda, W., Jongepier, A., & Cremer, J. L. (2023). Deep statistical solver for distribution system state estimation. *IEEE Transactions on Power Systems*, *39*(2), 4039-4050.
[9] Giraud, B. N., Rajaei, A., & Cremer, J. L. (2024). Constraint-driven deep learning for n-k security constrained optimal power flow. *Electric Power Systems Research*, *235*, 110692.

Reinforcement Learning

Notation: Environment S, action a, state x

Objective: $\pi(a|x)$ to maximise $J(\pi) = \mathbb{E}_{\pi}[\sum_{t=0}^{T} \gamma^t r(x_t, a_t)]$

Idea: Learn by interacting with the environment
No supervision, no explicit y_i labels



Approach

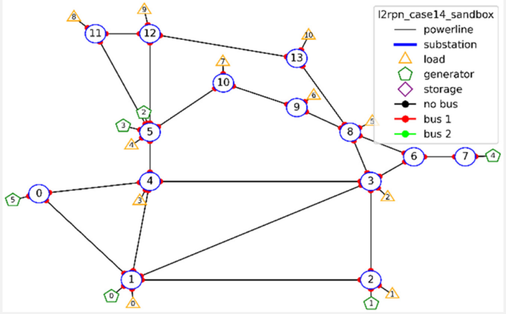
1. Interact with environment S
2. Collect many state-action-reward transitions

$$\Omega^T = \{(x_t, a_t, r_t, x_t')\}$$

3. Use π online for new states $t \notin \Omega^T$

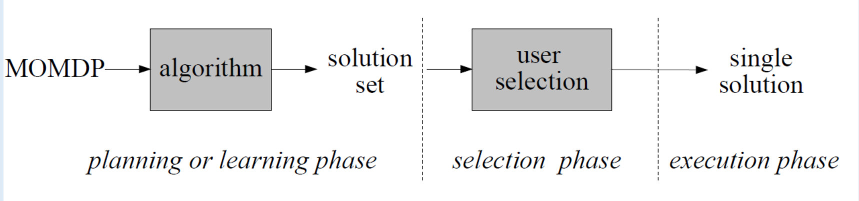
Applications

- Control PV inverters [10]
- Demand response [11]
- Topological reconfiguration[12]



Challenges

- Changes in the environment S
- High-dimensional state & action spaces (often heuristics are applied)
- Are the actions physically feasible?
- Safety & risks: How to explore safely?
- How about Model Predictive Control and Multi-Stage Optimization?



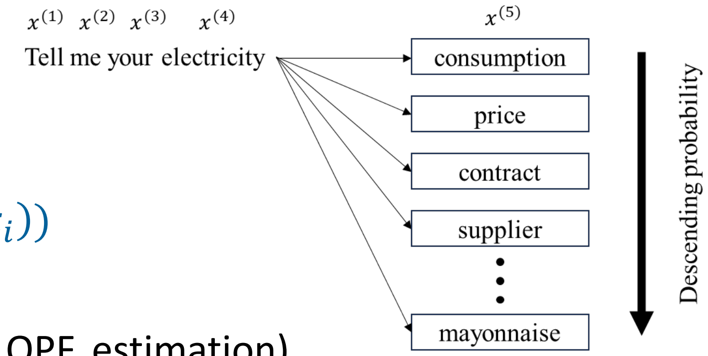
TU Delft

[10] Vergara, P. P., Salazar, M., Giraldo, J. S., & Palensky, P. (2022). Optimal dispatch of PV inverters in unbalanced distribution systems using reinforcement learning. *International Journal of Electrical Power & Energy Systems*, *136*, 107628.
[11] Čović, N., Cremer, J. L., & Pandžić, H. (2024). Learning a reward function for user-preferred appliance scheduling. *Electric power systems research*, *235*, 110667.
[12] Marot, A., Donnot, B., Romero, C., Donon, B., Lerousseau, M., Veyrin-Forrer, L., & Guyon, I. (2020). Learning to run a power network challenge for training topology controllers. *Electric Power Systems Research*, *189*, 106635.

Self-Supervised Learning

Objective: Learn a **useful internal representation** from unlabeled data by solving a **pretext task** — no human-labeled or simulator-labeled outputs required.

Idea: instead of training on (x_i, y_i) train on auto-generated pseudo-labels or tasks constructed from structure x_i



Approach

1. Generate many inputs $\Omega^T = \{(x_i\ )\}_{i=1}^N$
2. Define self-supervised pretext loss $\mathcal{L}_{pretext}(f(x_i))$
3. Train encoder $\sum_{i\in\Omega^T} \mathcal{L}_{pretext}(f(x_i))$
4. Use $f(x)$ for downstream *task* (e.g. forecasting, OPF, estimation)

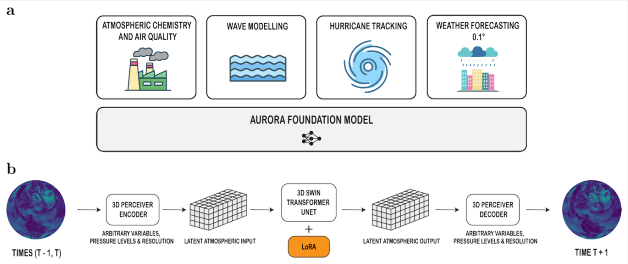
Benefits: Good initialization when little data, good transfer to downstream tasks

Challenges

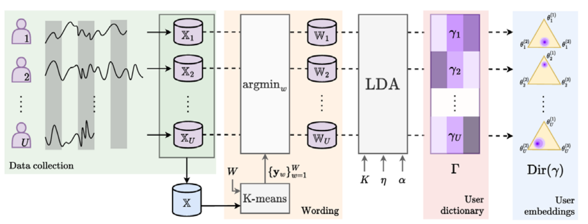
- Design pretext loss and model architectures with broad set of tasks, grid conditions, topologies
- Generate large data sets
- …

Applications

- Natural Language Processing
- Weather foundational models
- Earth system foundational models [13]



Load forecasting of users [14]



Grid foundation models (GFM) [15]

[13] Bodnar, C., Bruinsma, W. P., Lucic, A., Stanley, M., Vaughan, A., Brandstetter, J., ... & Perdikaris, P. (2024). A foundation model for the earth system. *arXiv preprint arXiv:2405.13063*.
[14] Bölat, Kutay, and Simon Tindemans. "GUIDE-VAE: Advancing Data Generation with User Information and Pattern Dictionaries." *arXiv preprint arXiv:2411.03936* (2024).
[15] Hamann, H. F., Gjorgiev, B., Brunschwiler, T., Martins, L. S., Puech, A., Varbella, A., ... & Sobolevsky, S. (2024). Foundation models for the electric power grid. Joule, 8(12), 3245-3258.

Graph Neural Networks

Objective: Improve generalization performance in learning tasks on network-structured systems (like power grids)

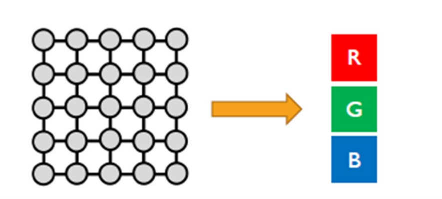
Idea: embedding graph topology directly into the model architecture as bias

Approach

1. Construct graph $G = (V, \mathcal{E})$ with features on nodes and edges
2. Define f_{GNN} and learn with message passing on supervised loss $\sum_{i \in \Omega^T} \|y_i - \hat{y}_i\|$
3. Use $f(x_j)$ for new j $\notin \Omega^T$ or on unseen graphs G'

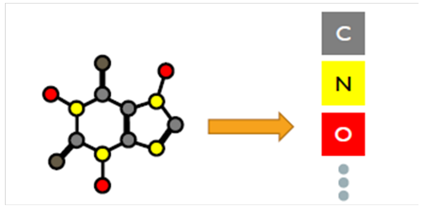
Benefits: Data efficient, generalisation to changes in topologies

Example: $p \times p$ RGB image



$\Omega = \mathbb{Z}_p \times \mathbb{Z}_p$ $\mathcal{C} = \mathbb{R}^3$

Example: molecular graph

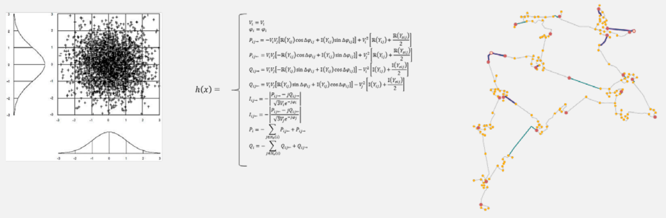


$\Omega = \{1,..n\}$ $\mathcal{C} = \mathbb{R}^c$

Applications

- Graph neural solvers [16] for ACOPF [17]
- Distribution system state estimation

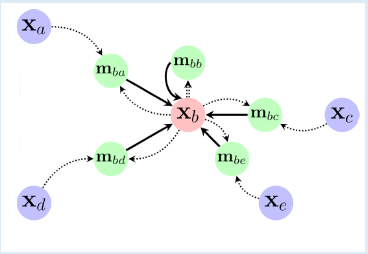
Noisy measurements Power flow equations Topology



Challenges

- Model inaccuracy $s \neq m$
- Long-range dependencies are difficult to learn. *Power system topology is sparse*
- Challenging to learn for *global* problems (e.g. ACOPF)

Good to learn local relationships



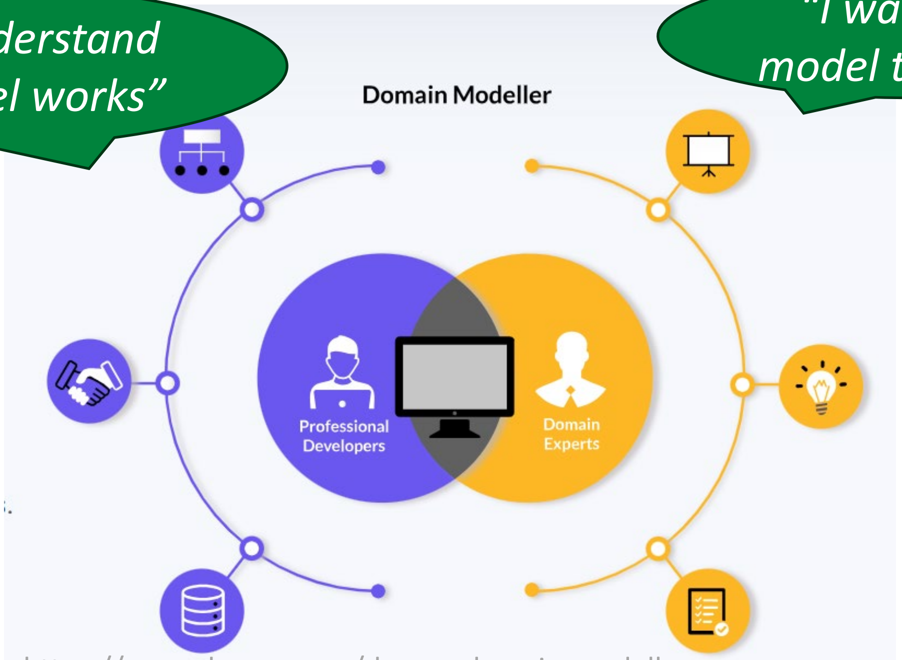
[16] Donon, B., Donnot, B., Guyon, I., & Marot, A. (2019, July). Graph neural solver for power systems. In *2019 international joint conference on neural networks (ijcnn)* (pp. 1-8). IEEE.
[17] Piloto, L., Liguori, S., Madjiheurem, S., Zgubic, M., Lovett, S., Tomlinson, H., ... & Witherspoon, S. (2024). Canos: A fast and scalable neural ac-opf solver robust to n-1 perturbations. *arXiv preprint arXiv:2403.17660.*

Explainable & Interpretable AI

Objective: provide **human-understandable reasoning** behind AI decisions.

“I want to understand how the model works”

“I want the model to work”



https://www.devum.com/devum-domain-modeller

Applications

- Interpretable structures (e.g. decision trees) for security assessments [18]
- Post-hoc explanations to complex models for transient stability based, e.g. SHAP values [19]

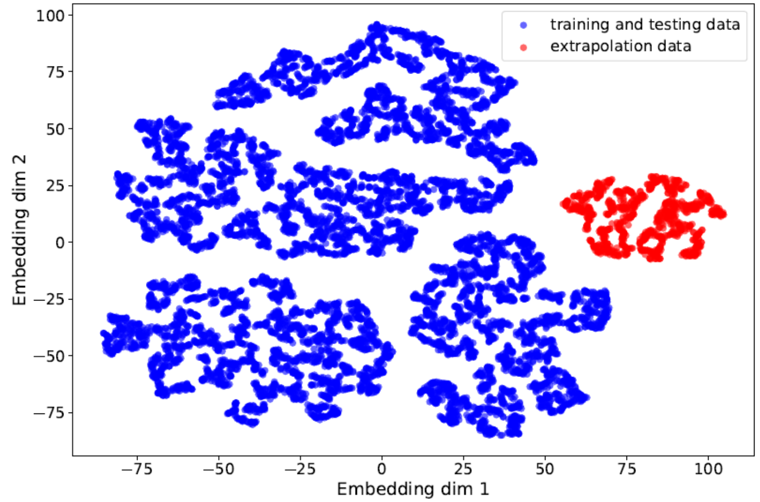
Challenges

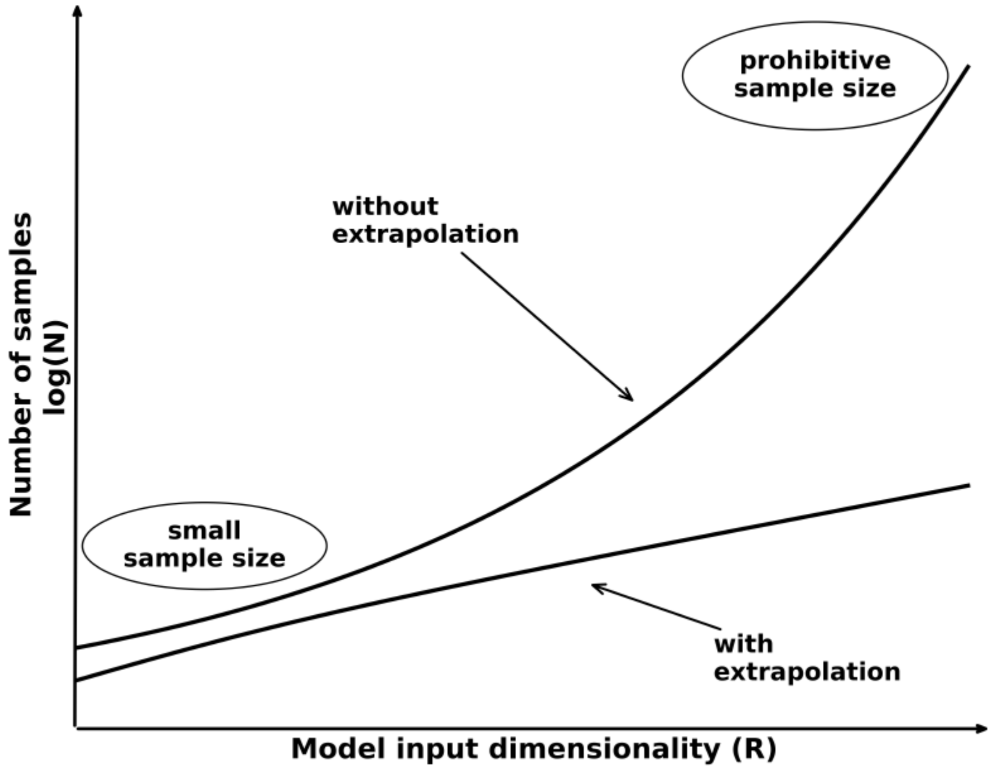
- Some trained models may not be able to state performance guarantees
- Is this action physical compliant?

[18] Cremer, Jochen L., Ioannis Konstantelos, and Goran Strbac. "From optimization-based machine learning to interpretable security rules for operation." *IEEE Transactions on Power Systems* 34, no. 5 (2019): 3826-3836.
[19] Hamilton, R. I., & Papadopoulos, P. N. (2023). Using SHAP values and machine learning to understand trends in the transient stability limit. *IEEE Transactions on Power Systems*, *39*(1), 1384-1397.

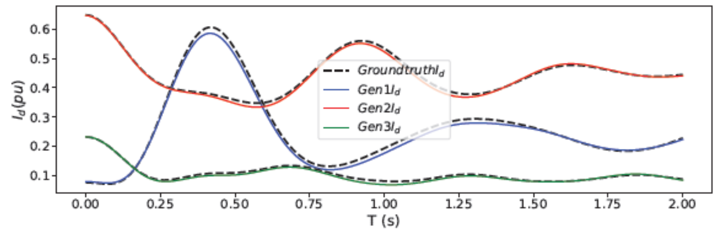
Generalisation to changes in s or m

The model performs well not just on training data, but on ***unseen scenarios*** *— new grid states, topologies, contingencies, or time horizons.*



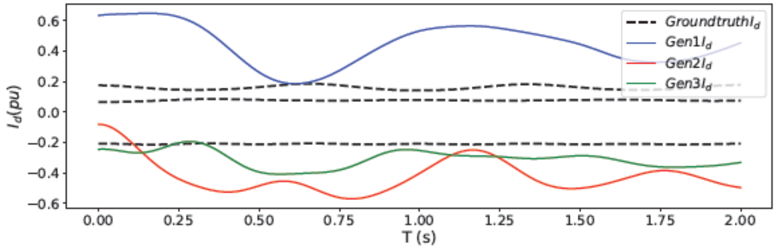


Extrapolation in continuous domain



(a) I_d current trajectory

Extrapolation in nonlinear domain (discrete)



(a) I_d current trajectory

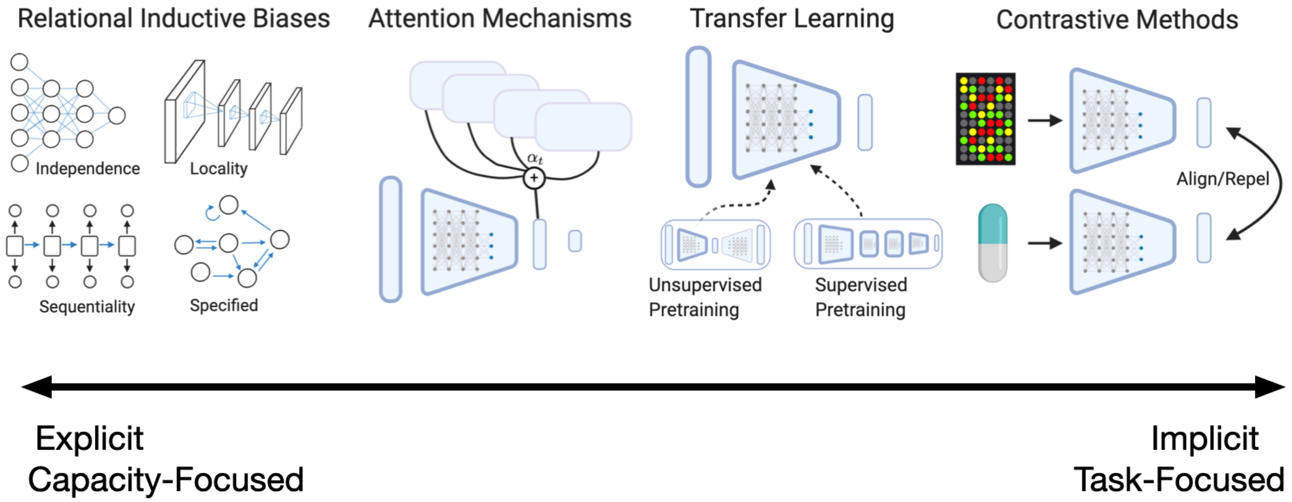
TU Delft

AI·EFFECT
DRIVING ENERGY TRANSITION THROUGH AI EXPERIMENTATION

[20] Olayiwola Arowolo, Jochen Stiasny, Jochen Cremer, “Exploring the Extrapolation Performance of Machine Learning Models for Power System Time Domain Simulations”, *12th Bulk Power System Dynamics and Control Symposium and Sustainable Energy, Grids and Networks Journal, 2025*

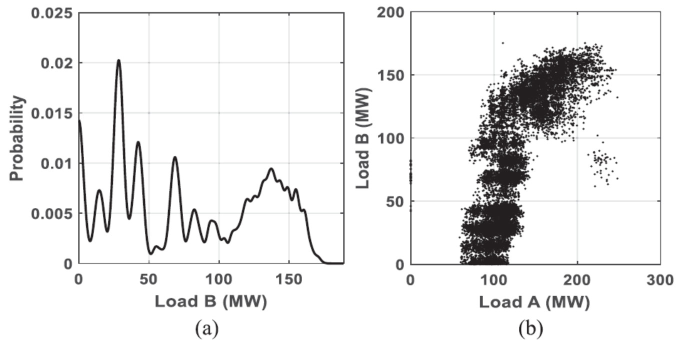
Challenge: Data-efficiency

- **Data efficiency is critical**
- Embedding **inductive bias** and learning **task-aware representations** helps supervised models generalise better — even with limited labels.

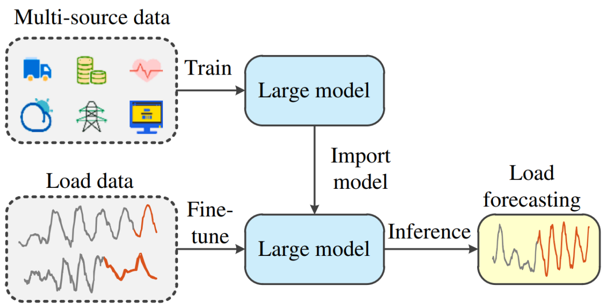


Sampling synthetic data & use real-data

Snapshot sampling



Time-series foundational models



[21] https://sgfin.github.io/2020/06/22/Induction-Intro/
[22] Konstantelos, I., Sun, M., Tindemans, S. H., Issad, S., Panciatici, P., & Strbac, G. (2018). Using vine copulas to generate representative system states for machine learning. IEEE Transactions on Power Systems, 34(1), 225-235.
[23] Al-Amin Bugaje, Jochen L. Cremer, Goran Strbac, “Split-based Sequential Sampling for Realtime Security Assessment”, *International Journal of Electrical Power & Energy Systems, 2022*
[24] A. Venzke, D.K. Molzahn, S. Chatzivasileiadis,(2019). Efficient Creation of Datasets for Data-Driven Power System Applications, arXiv:1910.01794
[25] Liao, W., Wang, S., Yang, D., Yang, Z., Fang, J., Rehtanz, C., & Porté-Agel, F. (2025). TimeGPT in load forecasting: A large time series model perspective. *Applied Energy*, *379*, 124973.

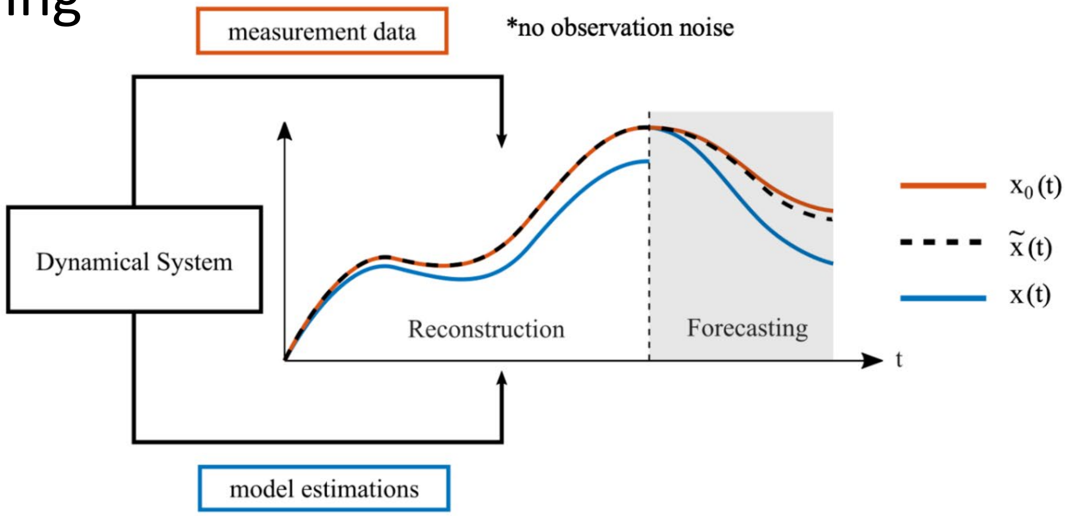
Model inaccuracy $s \neq m$ (data quality issues)

"All models are wrong, but some are useful", George E. P. Box

Sim-to-Real Domain Adaptation

- Example challenges
  - Distribution: Inaccurate transformer-tap positions
  - Transmission: Converter-based control models are unkown

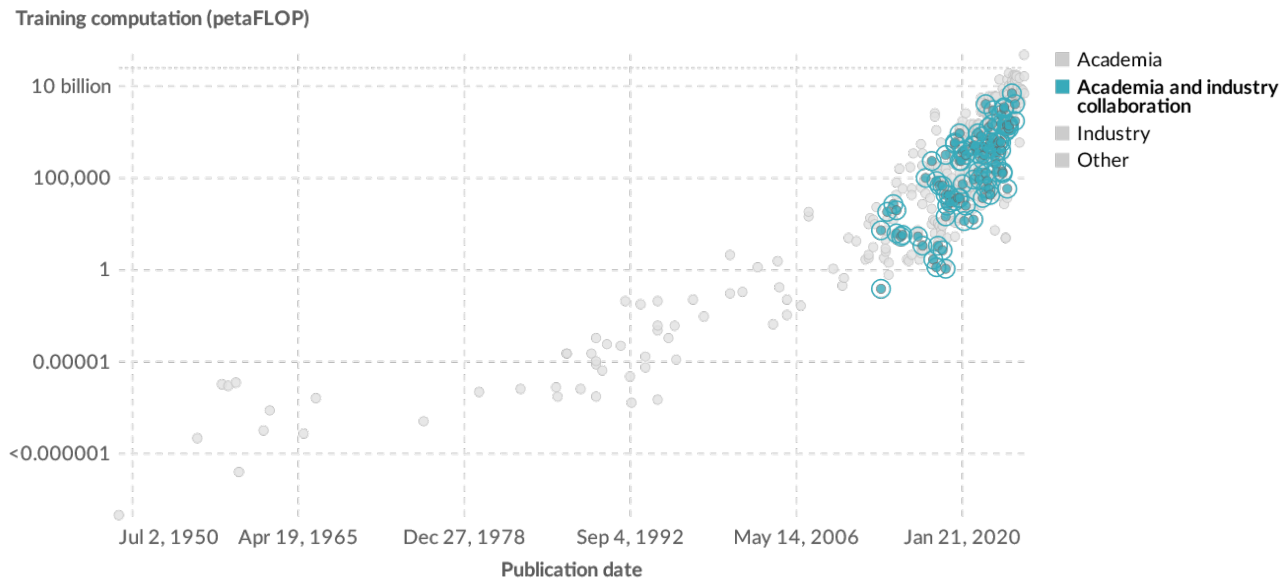
- Possible techniques: Parameter estimation to develop probabilistic and deterministic models, discrepancy learning



[26] Ebers, M. R., Steele, K. M., & Kutz, J. N. (2024). Discrepancy modeling framework: Learning missing physics, modeling systematic residuals, and disambiguating between deterministic and random effects. *SIAM Journal on Applied Dynamical Systems*, *23*(1), 440-469.

Conclusions

- Let’s work together to realise the potential of AI-based methods



Data4Grids project

- Let’s develop good representations to learn for grids

GenAI EU Horizon Projects with Canada…

- How to train data-efficiently models across system operators?
- Know when your model works and when it does not work (generalisation)

TUDelft

Thank you

Speaker

Jochen Cremer

Associate Professor IEPG, TU Delft

Web: https://www.tudelft.nl/ai/delft-ai-energy-lab

Personal www.jochen-cremer.com

Email: j.l.cremer@tudelft.nl

Code: https://github.com/TU-Delft-AI-Energy-Lab

TU Delft