Data and Modeling Assumptions in Physics-Informed Operator Learning

Martin Wellenhof-Karner[1,2], Alexander Fuchs[1,2], Bernhard C. Geiger[1,3,4] Franz Pernkopf[1,2,3]
[1] Signal Processing and Speech Communication Laboratory, Graz University of Technology
[2] Christian Doppler Laboratory for Dependable Intelligent Systems in Harsh Environments
[3] Graz Center for Machine Learning
[4] Know Center Research GmbH

Abstract

Operator networks have emerged as promising surrogate models, replacing computationally expensive numerical solvers for differential equations. Beyond achieving competitive accuracy with traditional solvers, the practical viability of this approach greatly depends on its training cost, which is comprised of ground truth data acquisition and network optimization. Physics-informed machine learning seeks to reduce reliance on labeled data by embedding the governing differential equations into the loss function; however, such models are often very challenging to train using physics constraints alone.

In this paper, we study how varying amounts of labeled data and architectural choices affect convergence and final performance in operator learning. Specifically, we compare an architecture developed for operator learning, i.e., Deep Operator Network, with a simpler MLP baseline across three training regimes: purely data-driven, purely physics-informed with no labeled data, and hybrid approaches that leverage *both* data and physics information. Our experiments on the double-mass-spring-damper system indicate that the physics-informed Deep Operator Network converges faster to the same performance if small amounts of labeled data are used. For the MLP architecture, which is not as well-tailored to the underlying dynamics, a purely physics-informed approach fails. In this case, incorporating labeled data mitigates architectural deficiencies and always substantially improves convergence and performance.

1 Introduction

Most processes in science and engineering are governed by differential equations (DEs). These processes are typically solved using state-of-the-art simulation frameworks, such as Runge-Kutta-based solvers, which over the years have been developed to reliably achieve high accuracy. However, depending on the underlying DEs, and investigated system size, obtaining these solutions can be computationally expensive, which is especially detrimental in research and development processes where repeated solver runs are required. To alleviate this limitation, recent research and development have explored the use of neural networks (NNs) as surrogate models for numerical solvers of DEs. Once trained, such networks can infer the solutions at a fraction of the computational cost of traditional numerical solvers. NNs as surrogate models have successfully been applied across a wide range of domains, including aerodynamics [Sun and Wang, 2019], fluid dynamics [Cai et al., 2021], weather forecasting [Bi et al., 2023], chemical engineering [Esche et al., 2022], and the rail-vehicle industry [Zhou et al., 2023, Ye et al.].

Operator learning has recently emerged as a promising approach for surrogate modeling. In this approach, the network aims to learn the underlying solution operator of the DE directly from data

1st Workshop on Differentiable Systems and Scientific Machine Learning @ EurIPS 2025

[Boullé and Townsend, 2024], e.g., Deep Operator Networks (DeepONets) [Lu et al.]. Despite their potential, operator networks require labeled data, which can be costly to generate. This shifts computational effort from inference to training; once trained, predictions are fast, but the data-gathering and training process can be resource-intensive. A parallel line of research on physics-informed neural networks (PINNs) [Raissi et al., 2019] can help to alleviate this problem. Indeed, the physics-informed loss used for training PINNs incorporates known DEs directly into the loss function, thus allowing to solve these DEs with little to no labeled data. Wang et al. [2021] suggested using this physics-informed loss for operator learning. While such a physics-based approach reduces the dependence on labeled data, it often leads to optimization difficulties [Hao et al., 2022].

The practical viability of NNs as a surrogate model depends primarily on four major factors: accuracy, training time, generalization ability, and reliability. In scenarios where one cannot gather additional training data, one must rely heavily on the incorporated physics-constraints. Conversely, in scenarios where training data acquisition is cheap, data-driven approaches work well. However, many real-world scenarios (e.g., NNs as surrogates for computationally demanding solvers) lie between these extremes. This raises the question: how valuable is additional labeled data in physics-informed operator learning? In this work, we aim to get a better understanding of how labeled data, network architecture, and inductive biases influence physics-informed machine learning. We use the double-mass-spring damper system with external excitations as a challenging benchmark problem to investigate how varying amounts of labeled data impact the accuracy and training efficiency of DeepONets and traditional MLPs under both purely data-driven, purely physics-informed and hybrid training regimes. Furthermore, we show the importance of inductive bias through the utilization of Fourier features to encode the input time.

2 Related work

The concept of physics-guided NNs dates back to the 1990s. Raissi et al. [2019] introduced PINNs, popularizing the concept by leveraging automatic differentiation and increased computational power to make this approach practically viable. However, training NNs exclusively using a physics-informed loss is often challenging [Wang et al., a, Rohrhofer et al.], particularly for systems involving high-frequency dynamics due to the spectral bias of NNs [Wang et al., b, Steger et al., 2022]. Applying input encodings has proven to be beneficial for learning higher frequencies in both data-driven and physics-informed learning settings, either through Fourier features [Tancik et al., 2020], [Wang et al., 2021] or by modifying the employed activation functions [Sitzmann et al., 2020, Wong et al., 2022, Hofmann-Wellenhof et al., 2024].

Operator learning aims to learn nonlinear mappings between function spaces. Among the most widely adopted architectures in this domain is DeepONet [Lu et al.], motivated by the Universal Approximation Theorem for Operators [Chen and Chen, 1995]. The DeepONet architecture has also been extended to multiple outputs, partitioning the output neurons of the subnetworks and computing multiple dot products [Lu et al., 2022]. Originally introduced as purely data-driven models, DeepONets have since been extended to physics-informed variants [Wang et al., 2021]. Recent studies explored extensions of these architectures to improve adaption to time-dependent inputs [Liu et al., 2023, He et al., 2024].

3 Methodology

In the following experiments, we employ a double mass–spring–damper system with external excitations (the quarter-car model), as it possesses several properties that make it a compelling test case. It is the easiest multibody system that exhibits coupled dynamics, is highly oscillatory, which poses challenges for NN learning, and involves time-varying input excitations, a scenario that has been relatively less explored for (physics-informed) DeepONets. The system is given by:

$$m_1\, \ddot{y}_1 = k_1(y_2 - y_1) + c_1(\dot{x}_2 - \dot{x}_1) \tag{1a}$$

$$m_2\, \ddot{y}_2 = k_2(u - y_2) + c_2(\dot{u} - \dot{x}_2) - k_1(y_2 - y_1) - c_1(\dot{x}_2 - \dot{x}_1), \tag{1b}$$

where $u(t)$ is the time varying excitation, m_1, m_2 being the weight of the masses, k_1, k_2 being the first and second spring constant, c_1, c_2 are the damping constants, and y_1, y_2 are the displacements of the first and second mass. The task is to train a surrogate model that takes the excitation signal $u(t)$ and time t as input and responds with the displacements $y_1(t)$ and $y_2(t)$, as shown in Figure 1.

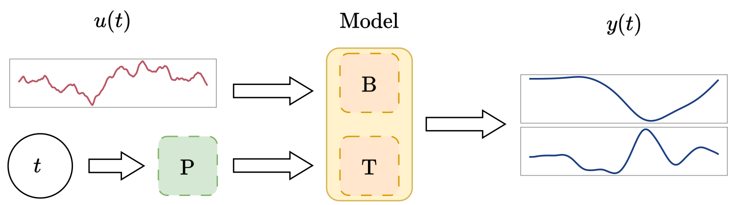

Figure 1: Overview of the learning task. The DeepONet takes the excitation $u(t)$ and projected time t as inputs to its branch and trunk networks, respectively, while the MLP uses their concatenation. Both models output the predicted displacements of the two masses, $y_1(t)$ and $y_2(t)$.

Table 1: Relative L_2 test error for the different architectures with and without input Fourier encoding. The data-driven and hybrid training scheme was trained on 270 data pairs.

	data-driven	hybrid	physics-informed
DeepONet + input proj.	0.072 ± 0.007	$\mathbf{0.032 \pm 0.004}$	$\mathbf{0.031 \pm 0.001}$
DeepONet	0.112 ± 0.004	0.159 ± 0.056	0.211 ± 0.038
MLP + input proj.	$\mathbf{0.048 \pm 0.001}$	0.043 ± 0.003	0.091 ± 0.018
MLP	0.081 ± 0.005	0.101 ± 0.020	0.179 ± 0.017

We compare two model architectures: a baseline MLP and a DeepONet. In most DeepONet applications, an initial condition is provided and propagated forward in time to obtain the solution. Our benchmark problem differs, as it additionally requires time-dependent excitation signals as inputs. DeepONet consists of two sub-networks whose outputs are combined via a dot product. Here, one sub-network, i.e., the trunk net, encodes the coordinates of the solution functions (i.e., the time variable t), while the other, the branch net, encodes the input function (i.e., the excitation signal $u(t)$). For the trunk net, we also analyze the effectiveness of linearly spaced Fourier feature encodings, projecting the input time t to a vector $\mathbf{v}(t)$ consisting of 200 sine and cosine pairs. As the system response can be expected to be an oscillating signal, this introduces an inductive bias. All model variants are trained under three regimes: purely data-driven, purely physics-informed, and a hybrid approach that combines both.

4 Experimental setup and results

Both model types have a depth of four and are optimized using Adam [Kingma and Ba, 2017]. The MLP has a width of 64 (70,274 parameters), and the two subnets of the DeepONet have a width of 106 (in total 70,702 parameters). The used excitations are artificially generated signals developed for modeling rail roughness. This data is then split into training, validation, and test sets. Excitations $u(t)$ are two-second snippets, assuming a sampling rate of 250 Hz. The physics residual of the DE (1) is evaluated at $N = 500$ collocation points. Initial displacement and velocity are set to zero for both masses, enforced through the data loss. During training, early stopping was employed. Our main performance metric was the relative L_2 error calculated snipped-wise,

$$L_{2,r} = \frac{1}{N} \sum_{i=1}^{N} \frac{\sqrt{\sum_{t=1}^{T} \left(y_t^{(i)} - \hat{y}_t^{(i)} \right)^2}}{\sqrt{\sum_{t=1}^{T} \left(y_t^{(i)} \right)^2}},$$

which is independent of excitation amplitudes. All reported results are averaged over three runs.

Table 1 highlights the importance of inductive biases, introduced through the Fourier encodings, especially for the hybrid and physics-informed training schemes. Across all models, performance improves noticeably if Fourier encodings are introduced. We see that DeepONet performs approximately $7\times$ better with the encodings in the physics-informed case, and about $1.5\times$ better for the data-driven variant. Consequently, all subsequent experiments were conducted using Fourier encodings.

Figure 2 shows $L_{2,r}$ as a function of the number of training data points, i.e., excitation-solution pairs. In the data-driven regime, more data generally improves accuracy. In contrast, for the hybrid and physics-informed DeepONet models, additional data does not yield further performance gains. This indicates that the physics-informed DeepONet can solve the problem solely using the

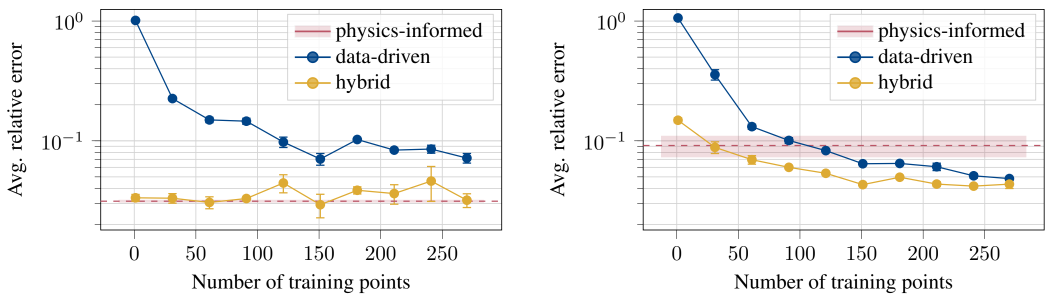

Figure 2: Comparison of data-driven, hybrid, and physics-informed model performance with standard deviation, for the DeepONet (left) and MLP (right) architectures over the number of used training data points (excitation-solution pairs).

physics constraint. Across all dataset sizes, a clear performance gap persists between the data-driven DeepONet and the models incorporating a physics loss. For the MLP, however, the physics-informed variant performs worse than the other paradigms, in all but the very low-data regime. The hybrid MLP outperforms the purely data-driven MLP, although this advantage diminishes as the training set grows. Notably, the MLP performs better than the DeepONet in the purely data-driven setting.

Figure 3 relates $L_{2,r}$ to the number of training epochs, confirming the intuition that the data-driven approach converges fastest, being the easiest to optimize. Additionally, hybrid models converge faster than those relying solely on the physics. Using the physics residual, on the other hand, improves accuracy considerably, albeit at the cost of increased convergence time. One surprising finding is that the amount of data does *not* have a significant impact on convergence speed. Similarly, for the hybrid DeepONet, the amount of used data has little impact on accuracy. For the hybrid MLP, however, accuracy increases with larger datasets, allowing it to surpass the purely physics-informed approach.

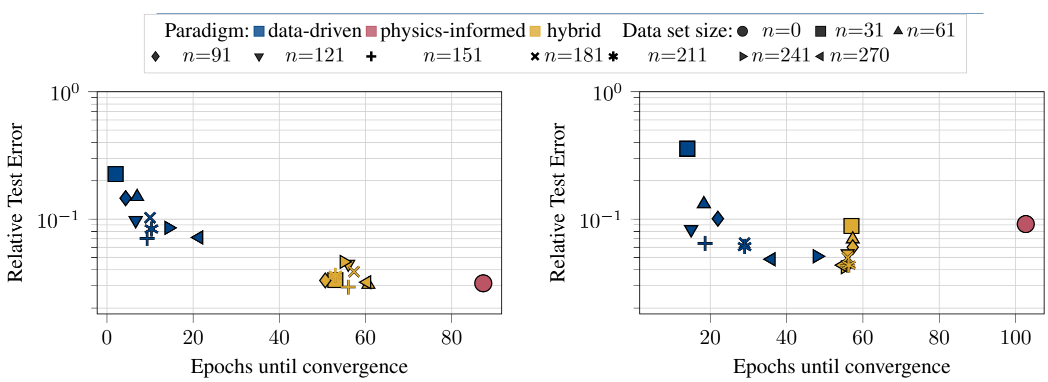

Figure 3: Pareto front comparison of $L_{2,r}$ errors at the early stopping epochs for the DeepONet (left) and MLP (right) architectures. The plots illustrate trade-offs between model training complexity and performance across configurations.

5 Conclusion

In this work, we analyzed how varying amounts of training data and inductive biases influence the convergence behavior and performance of operator networks. Our results show that incorporating training data into a physics-informed training regime can significantly reduce training time. However, this effect does not scale with the amount of data used. Conversely, adding physics-based residuals can substantially enhance performance, if the network architecture is well-suited to exploit the underlying physical behavior. Architectures specifically designed for operator learning, such as DeepONet, combined with suitable inductive biases, such as Fourier feature embeddings, play a crucial role in leveraging the potential of the physics loss. These findings motivate further research into developing tailored modeling approaches for specific differential equations. Otherwise, training data is required to compensate for suboptimal design choices to achieve satisfactory accuracy.

Acknowledgments and Disclosure of Funding

The financial support by the Austrian Federal Ministry of Labour and Economy, the National Foundation for Research, Technology and Development, and the Christian Doppler Research Association is gratefully acknowledged. Furthermore, the research was funded by Siemens Mobility.

The work of Bernhard C. Geiger has been partly supported by European Union's HORIZON Research and Innovation Programme under grant agreement No 101120657, project ENFIELD (European Lighthouse to Manifest Trustworthy and Green AI).

References

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, 619(7970):533–538, 2023.

Nicolas Boullé and Alex Townsend. A mathematical guide to operator learning. In *Handbook of Numerical Analysis*, volume 25, pages 83–125. Elsevier, 2024.

Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechanica Sinica*, 37(12):1727–1738, 2021.

Tianping Chen and Hong Chen. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE transactions on neural networks*, 6(4):911–917, 1995.

Erik Esche, Joris Weigert, Gerardo Brand Rihm, Jan Göbel, and Jens-Uwe Repke. Architectures for neural networks as surrogates for dynamic systems in chemical engineering. *Chemical Engineering Research and Design*, 177:184–199, 2022.

Zhongkai Hao, Songming Liu, Yichi Zhang, Chengyang Ying, Yao Feng, Hang Su, and Jun Zhu. Physics-informed machine learning: A survey on problems, methods and applications. *arXiv preprint arXiv:2211.08064*, 2022.

Junyan He, Shashank Kushwaha, Jaewan Park, Seid Koric, Diab Abueidda, and Iwona Jasiuk. Sequential deep operator networks (s-deeponet) for predicting full-field solutions under time-dependent loads. *Engineering Applications of Artificial Intelligence*, 127:107258, 2024.

Martin Hofmann-Wellenhof, Alexander Fuchs, and Franz Pernkopf. On training physics-informed neural networks for oscillating problems. In *ICLR 2024 Workshop on AI4DifferentialEquations In Science*, 2024.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL `https://arxiv.org/abs/1412.6980`.

Fangyu Liu, Junlin Li, and Linbing Wang. Pi-lstm: Physics-informed long short-term memory network for structural response modeling. *Engineering Structures*, 292:116500, 2023.

Lu Lu, Pengzhan Jin, and George Em Karniadakis. DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. doi: 10.48550/ARXIV.1910.03193. URL `https://arxiv.org/abs/1910.03193`. Publisher: arXiv Version Number: 3.

Lu Lu, Xuhui Meng, Shengze Cai, Zhiping Mao, Somdatta Goswami, Zhongqiang Zhang, and George Em Karniadakis. A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data. *Computer Methods in Applied Mechanics and Engineering*, 393:114778, 2022.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Franz M. Rohrhofer, Stefan Posch, Clemens Gößnitzer, and Bernhard C. Geiger. On the role of fixed points of dynamical systems in training physics-informed neural networks. URL `http://arxiv.org/abs/2203.13648`.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33: 7462–7473, 2020.

Sophie Steger, Franz M. Rohrhofer, and Bernhard Claus Geiger. How PINNs cheat: Predicting chaotic motion of a double pendulum. In *The Symbiosis of Deep Learning and Differential Equations (DLDE) @ NeurIPS*, virtual, December 2022.

Gang Sun and Shuyue Wang. A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(16):5863–5872, 2019.

Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient pathologies in physics-informed neural networks, a. URL `http://arxiv.org/abs/2001.04536`.

Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why PINNs fail to train: A neural tangent kernel perspective. 449:110768, b. ISSN 0021-9991. doi: 10.1016/j.jcp.2021.110768. URL `https://www.sciencedirect.com/science/article/pii/S002199912100663X`.

Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. *Science advances*, 7(40):eabi8605, 2021.

Jian Cheng Wong, Chin Chun Ooi, Abhishek Gupta, and Yew-Soon Ong. Learning in sinusoidal spaces with physics-informed neural networks. *IEEE Transactions on Artificial Intelligence*, 5(3):985–1000, 2022.

Yunguang Ye, Ping Huang, Yu Sun, and Dachuan Shi. MBSNet: A deep learning model for multibody dynamics simulation and its application to a vehicle-track system. 157:107716. ISSN 0888-3270. doi: 10.1016/j.ymssp.2021.107716. URL `https://www.sciencedirect.com/science/article/pii/S0888327021001114`.

Shiyang Zhou, Alexander Meierhofer, Ozan Kugu, Yuxi Xia, and Manfred Grafinger. A machine-learning-based surrogate modeling methodology for submodel integration in the holistic railway digital twin platform. *Procedia CIRP*, 119:345–350, 2023.

A Technical Appendices and Supplementary Material

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: **[TODO]**

   Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: **[TODO]**

    Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.