Quantus: An Explainable AI Toolkit for Responsible Evaluation of Neural Network Explanations

Anna Hedström[1,†] Leander Weber[2] Dilyara Bareeva[1] Franz Motzkus[2]
Wojciech Samek[2,3] Sebastian Lapuschkin[2,†] Marina M.-C. Höhne[1,3,†]

[1] *Understandable Machine Intelligence Lab, Technische Universität Berlin, 10587 Berlin, Germany*
[2] *Department of Artificial Intelligence, Fraunhofer Heinrich-Hertz-Institute, 10587 Berlin, Germany*
[3] *BIFOLD – Berlin Institute for the Foundations of Learning and Data, 10587 Berlin, Germany*

Abstract

The evaluation of explanation methods is a research topic that has not yet been explored deeply, however, since explainability is supposed to strengthen trust in artificial intelligence, it is necessary to systematically review and compare explanation methods in order to confirm their correctness. Until now, no tool exists that exhaustively and speedily allows researchers to *quantitatively* evaluate explanations of neural network predictions. To increase transparency and reproducibility in the field, we therefore built `Quantus` — a comprehensive, open-source toolkit in Python that includes a growing, well-organised collection of evaluation metrics and tutorials for evaluating explainable methods. The toolkit has been thoroughly tested and is available under open source license on PyPi (or on `https://github.com/understandable-machine-intelligence-lab/quantus/`).

Keywords: explainability, responsible AI, reproducibility, open source, python

1 Introduction

Despite much excitement and activity in the field of eXplainable Artificial Intelligence (XAI) [1, 2, 3, 4, 5], the evaluation of explainable methods still remains an unsolved problem [6, 7, 8, 9, 10]. Unlike in traditional Machine Learning (ML), the task of *explaining* inherently lacks "ground-truth" data — there is no universally accepted definition of what constitutes a "correct" explanation and less so, which properties an explanation ought to fulfill [11]. Due to this lack of standardised evaluation procedures in XAI, researchers frequently conceive new ways to experimentally examine explanation methods [6, 12, 13, 11, 14], oftentimes employing different parameterisations and various kinds of preprocessing and normalisations, each leading to different or even contrasting results, making evaluation outcomes difficult to interpret and compare. Critically, we note that it is common for XAI papers tend to base their conclusions on one-sided, sometimes methodologically questionable evaluation procedures — which we fear is hindering access to the current State-of-the-art (SOTA) in XAI and potentially may hurt the perceived credibility of the field over time.

For these reasons, researchers often rely on a qualitative evaluation of explanation methods e.g., [15, 16, 17], assuming that humans know what an "accurate" explanation would look like (or rather *should* look like, often disregarding the role that the explained model plays in the explanation process). However, the assumption that humans are able to recognise a correct explanation is generally not justified: not only does the notion of an "accurate" explanation often depend on the specifics of the task at hand, humans are also questionable judges of quality [18, 19]. To make matters more challenging, recent studies suggest that even quantitative evaluation of explainable methods is far from fault-proof [9, 20, 21, 22].

[†] ✉ `anna.hedstroem@tu-berlin.de, marina.hoehne@tu-berlin.de, sebastian.lapuschkin@hhi.fraunhofer.de`

In response to these issues, we developed `Quantus`, to provide the community with a versatile and comprehensive toolkit that collects, organises, and explains a wide range of evaluation metrics proposed for explanation methods. The library is designed to help automate the process of *XAI quantification* — by delivering speedy, easily digestible, and at the same time holistic summaries of the quality of the given explanations. As we see it, `Quantus` concludes an important, still missing contribution in today's XAI research by filling the gap between what the community produces and what it currently needs: a more quantitative, systematic and standardised evaluation of XAI methods.

2 Toolkit overview

`Quantus` provides its intended users — practitioners and researchers interested in the domains of ML and XAI — with a steadily expanding list of 25+ reference metrics to evaluate explanations of ML predictions. Moreover, it offers comprehensive guidance on how to use these metrics, including information about potential pitfalls in their application.

The library is thoroughly documented and includes in-depth tutorials covering multiple use-cases and tasks — from a comparative analysis of XAI methods and attributions, to quantifying to what extent evaluation outcomes are dependent on metrics' parameterisations. In Figure 1, we demonstrate some example analysis that can be produced with `Quantus`[1]. Moreover, the library provides an abstract layer between APIs of deep learning frameworks e.g. `PyTorch` [23] and `tensorflow` [24] and can be employed iteratively both during- and after model training in the ML lifecycle. Code quality is ensured by thorough testing, using `pytest` and continuous integration (CI), where every new contribution is automatically checked for sufficient test coverage. We employ syntax formatting with `flake8` under various Python versions.

Unlike other XAI-related libraries[2], `Quantus` has its primary focus on evaluation and as such, supports a breadth of metrics, spanning various different categories (see Table 1). Detailed descriptions of the different evaluation categories are documented in the repository. The first iteration of the library mainly focuses on attribution-based explanation techniques[3] for (but not limited to) image classification. In planned future releases, we are working towards extending the applicability of the library further e.g., by developing additional metrics and functionality that will enable users to perform checks, verifications and sensitivity analyses on top of the metrics.

Table 1: Comparison of four XAI libraries — (`AIX360` [2], `captum` [29], `TorchRay` [30] and `Quantus`) in terms of the number of XAI evaluation methods for six different evaluation categories, as implemented in each library.

Library	Faithfulness	Robustness	Localisation	Complexity	Axiomatic	Randomisation
`Captum` (2)	1	1	0	0	0	0
`AIX360` (2)	2	0	0	0	0	0
`TorchRay` (1)	0	0	1	0	0	0
`Quantus` (27)	**9**	**4**	**6**	**3**	**3**	**2**

3 Library design

The user-facing API of `Quantus` is designed with the aim of replacing an oftentimes lengthy and open-ended evaluation procedure with structure and speed — with a single line of code, the user

[1]The full experiment can be reproduced (and obtained) at the repository, under the `tutorials` folder.

[2]Related libraries were selected with respect to the XAI evaluation capabilities. Packages including no metrics for evaluation of explanation methods, e.g., `Alibi` [25], `iNNvestigate` [26], `dalex` [27] and `zennit` [28] were excluded.

[3]This category of explainable methods aims to assign an importance value to the model features and arguably, is the most studied group of explanation.

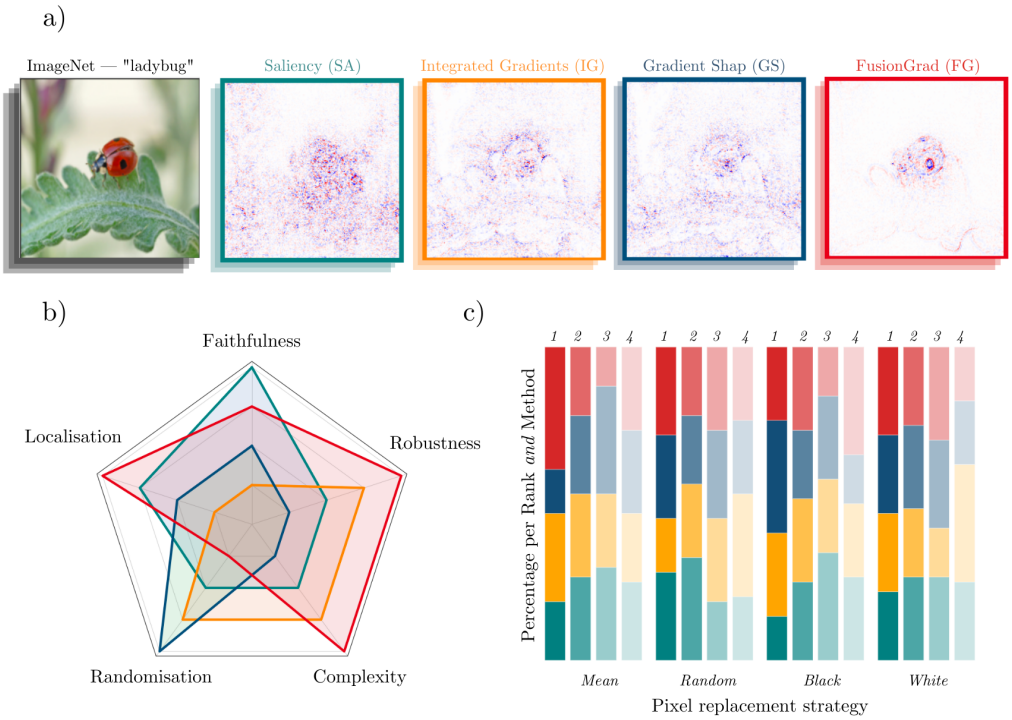

Figure 1: *a)* Simple *qualitative* comparison of XAI methods is often not sufficient to distinguish which gradient-based method — Saliency [17], Integrated Gradients [31], GradientShap [32] or FusionGrad [33] is preferred. With `Quantus`, we can obtain richer insights on how the methods compare *b)* by holistic quantification on several evaluation criteria and *c)* by providing sensitivity analysis of how a single parameter e.g. pixel replacement strategy of a faithfulness test influences the ranking of explanation methods.

can gain quantitative insights of how their explanations are behaving under various criteria. In the following code snippet, we demonstrate one way for how `Quantus` can be used to evaluate pre-computed explanations via a `PixelFlipping` experiment [12] — by simply calling the initialised metric instance. In this example, we assume to have a pre-trained model (`model`), a batch of input- and output pairs (`x_batch`, `y_batch`) and a set of attributions (`a_batch`).

```
import quantus

pixelflipping = quantus.PixelFlipping(perturb_baseline="black", normalise=False,
                                      features_in_step=28)
scores = pixelflipping(model, x_batch, y_batch, a_batch, **params)
# [0.6653, 0.4972, 0.4343, ...]

pixelflipping.plot(y_batch=y_batch, scores=scores)
```

Needless to say, XAI evaluation is intrinsically difficult and there is no one-size-fits-all metric for all tasks — evaluation of explanations must be understood and calibrated from its context: the application, data, model, and intended stakeholders [10, 34]. To this end, we designed `Quantus` to be highly customisable and easily extendable — documentation and examples on how to create new metrics as well as how to customise existing ones are included. Thanks to the API, any supporting functions of the evaluation procedure, e.g., `perturb_baseline` — that determines with what value patches of the input shall be iteratively masked — can flexibly be replaced by a user-specified function to ensure that the evaluation procedure is appropriately contextualised.

It is practically well-known but not yet publicly recognised that evaluation outcomes of explanations

can be highly sensitive to the parameterisation of metrics [20, 35] and other confounding factors introduced in the evaluation procedure [9, 36]. Therefore, to encourage a thoughtful and responsible selection and parameterisation of metrics, we added mechanisms such as warnings, checks and user guidelines, cautioning users to reflect upon their choices. Great care has to be taken when interpreting the quantification results and to this end, we provide additional functionality on potential interpretation pitfalls.

4 Broader impact

We built `Quantus` to raise the bar of *XAI quantification* — to substitute an ad-hoc and sometimes ineffective evaluation procedure with reproducibility, simplicity and transparency. From our perspective, `Quantus` contributes to the XAI development by helping researchers to speed up the development and application of explanation methods, dissolve existing ambiguities and enable more comparability. As we see it, steering efforts towards increasing objectiveness of evaluations and reproducibility in the field will prove rewarding for the community as a whole. We are convinced that a holistic, multidimensional take on XAI quantification will be imperative to the general success of (X)AI over time.

Acknowledgments

This work was partly funded by the German Ministry for Education and Research through project Explaining 4.0 (ref. 01IS20055) and BIFOLD (ref. 01IS18025A and ref. 01IS18037A), the Investitionsbank Berlin through BerDiBA (grant no. 10174498), as well as the European Union's Horizon 2020 programme through iToBoS (grant no. 965221).

References

[1] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. "Methods for interpreting and understanding deep neural networks". In: *Digit. Signal Process.* 73 (2018), pp. 1–15.

[2] Vijay Arya et al. *One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques*. 2019.

[3] Sebastian Lapuschkin et al. "Unmasking Clever Hans Predictors and Assessing What Machines Really Learn". In: *CoRR* abs/1902.10178 (2019).

[4] Wojciech Samek et al. "Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications". In: *Proc. IEEE* 109.3 (2021), pp. 247–278.

[5] Kirill Bykov et al. "Explaining Bayesian Neural Networks". In: *CoRR* abs/2108.10346 (2021).

[6] Wojciech Samek et al. "Evaluating the Visualization of What a Deep Neural Network Has Learned". In: *IEEE Trans. Neural Networks Learn. Syst.* 28.11 (2017), pp. 2660–2673.

[7] Julius Adebayo et al. "Debugging Tests for Model Explanations". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al. 2020.

[8] Andreas Holzinger, André M. Carrington, and Heimo Müller. "Measuring the Quality of Explanations: The System Causability Scale (SCS)". In: *Künstliche Intell.* 34.2 (2020), pp. 193–198.

[9] Gal Yona and Daniel Greenfeld. "Revisiting Sanity Checks for Saliency Maps". In: *CoRR* abs/2110.14297 (2021).

[10] Leila Arras, Ahmed Osman, and Wojciech Samek. "CLEVR-XAI: A Benchmark Dataset for the Ground Truth Evaluation of Neural Network Explanations". In: *Information Fusion* 81 (2022), pp. 14–40.

[11] Mengjiao Yang and Been Kim. "Benchmarking Attribution Methods with Relative Feature Importance". In: *CoRR* abs/1907.09701 (2019).

[12] Sebastian Bach et al. "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation". In: *PloS one* 10.7 (2015).

[13] Julius Adebayo et al. "Sanity Checks for Saliency Maps". In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al. 2018, pp. 9525–9536.

[14] Pieter-Jan Kindermans et al. "The (Un)reliability of Saliency Methods". In: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Ed. by Wojciech Samek et al. Vol. 11700. Lecture Notes in Computer Science. Springer, 2019, pp. 267–280.

[15] Matthew D. Zeiler and Rob Fergus. "Visualizing and Understanding Convolutional Networks". In: *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*. Ed. by David J. Fleet et al. Vol. 8689. Lecture Notes in Computer Science. Springer, 2014, pp. 818–833.

[16] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. Ed. by Balaji Krishnapuram et al. ACM, 2016, pp. 1135–1144.

[17] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. "Learning Important Features Through Propagating Activation Differences". In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 3145–3153.

[18] Danding Wang et al. "Designing Theory-Driven User-Centric Explainable AI". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*. Ed. by Stephen A. Brewster et al. ACM, 2019, p. 601.

[19] Avi Rosenfeld. "Better Metrics for Evaluating Explainable Artificial Intelligence". In: *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*. Ed. by Frank Dignum et al. ACM, 2021, pp. 45–50.

[20] Naman Bansal, Chirag Agarwal, and Anh Nguyen. "SAM: The Sensitivity of Attribution Methods to Hyperparameters". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 8670–8680.

[21] Céline Budding et al. "Evaluating saliency methods on artificial data with different background types". In: *CoRR* abs/2112.04882 (2021).

[22] Peter Hase and Mohit Bansal. "Evaluating Explainable AI: Which Algorithmic Explanations Help Users Predict Model Behavior?" In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by Dan Jurafsky et al. Association for Computational Linguistics, 2020, pp. 5540–5552.

[23] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach et al. 2019, pp. 8024–8035.

[24] Martin Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2016.

[25] Janis Klaise et al. "Alibi Explain: Algorithms for Explaining Machine Learning Models". In: *J. Mach. Learn. Res.* 22 (2021), 181:1–181:7.

[26] Maximilian Alber et al. "iNNvestigate Neural Networks!" In: *J. Mach. Learn. Res.* 20 (2019), 93:1–93:8.

[27] Hubert Baniecki et al. "dalex: Responsible Machine Learning with Interactive Explainability and Fairness in Python". In: *J. Mach. Learn. Res.* 22 (2021), 214:1–214:7.

[28] Christopher J. Anders et al. *Software for Dataset-wide XAI: From Local Explanations to Global Insights with Zennit, CoRelAy, and ViRelAy*. 2021. arXiv: `2106.13200 [cs.LG]`.

[29] Narine Kokhlikyan et al. *Captum: A unified and generic model interpretability library for PyTorch*. 2020.

[30] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. *Understanding Deep Networks via Extremal Perturbations and Smooth Masks*. 2019.

[31] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. "Axiomatic Attribution for Deep Networks". In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 3319–3328.

[32] Scott M. Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon et al. 2017, pp. 4765–4774.

[33] Kirill Bykov et al. "NoiseGrad: enhancing explanations by introducing stochasticity to model weights". In: *CoRR* abs/2106.10185 (2021).

[34] Ajay Chander and Ramya Srinivasan. "Evaluating Explanations by Cognitive Value". In: *Machine Learning and Knowledge Extraction - Second IFIP TC 5, TC 8/WG 8.4, 8.9, TC 12/WG 12.9 International Cross-Domain Conference, CD-MAKE 2018, Hamburg, Germany, August 27-30, 2018, Proceedings*. Ed. by Andreas Holzinger et al. Vol. 11015. Lecture Notes in Computer Science. Springer, 2018, pp. 314–328.

[35] Chirag Agarwal and Anh Nguyen. "Explaining Image Classifiers by Removing Input Features Using Generative Models". In: *Computer Vision - ACCV 2020 - 15th Asian Conference on Computer Vision, Kyoto, Japan, November 30 - December 4, 2020, Revised Selected Papers, Part VI*. Ed. by Hiroshi Ishikawa et al. Vol. 12627. Lecture Notes in Computer Science. Springer, 2020, pp. 101–118.

[36] Peter Hase, Harry Xie, and Mohit Bansal. "The Out-of-Distribution Problem in Explainability and Search Methods for Feature Importance Explanations". In: *Advances in Neural Information Processing Systems* 34 (2021).