A Dashboard for Interactive Convolutional Neural Network Training And Validation Through Saliency Maps

Tim Cech, Furkan Simsek, Willy Scheibel and Jürgen Döllner

University of Potsdam, Digital Engineering Faculty, Germany

Abstract

Quali-quantitative methods provide ways for interrogating Convolutional Neural Networks (CNN) [1]. For it, we propose a dashboard using a quali-quantitative method based on quantitative metrics and saliency maps. By those means, a user can discover patterns during the training of a CNN. With this, they can adapt the training hyperparameters of the model, obtaining a CNN that learned patterns desired by the user. Furthermore, they neglect CNNs which learned undesirable patterns. This improves users' agency over the model training process.

The CUB200-2011 dataset

We present SalienCNN on the example of the CUB200-2011 dataset [2] because this dataset was used in previous studies on model architecture understanding (e.g. [3]) but SalienCNN is applicable on other datasets. The CUB200-2011 dataset contains images of examples for bird species. Then, the task is to train a CNN that discerns between the species.

Dashboard

Our dashboard SalienCNN contains three major views. In particular:

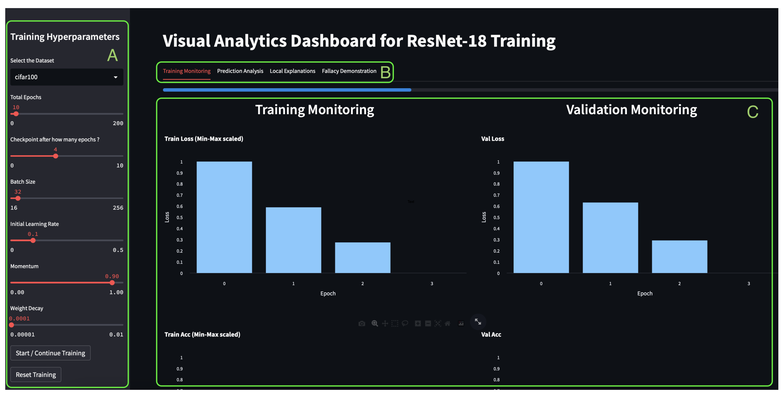
- A default view containing visualizations of standard quantitative measurements
- A side-by-side view comparing different Saliency Map techniques
- A fallacy view exemplifying one particular weakness of some Saliency Map techniques

We utilize three different Saliency Map techniques:

- Class Activation Mapping (CAM) [4]
- Gradient CAM (GradCAM) [5]
- Smoothed GradCAM++ (SmoothGradCAMpp) [6]

The default view

The default view contains visualizations of standard quantitative measurements, e.g., a bar chart showing the accuracy development over epochs or a confusion matrix [7].



The side-by-side view

In the side-by-side view, as shown in Figure 1, SMs based on CAM, GradCAM, and SmoothGradCAMpp per epoch are shown side by side to allow users to assess the model quality in terms of the more complex patterns. For it, a user chooses a class they wish to investigate. Then, we recommend to the user an SM progression over time for each of the following cases according to the training result of the most current epoch: (1) A correctly classified image, (2) an image wrongly attributed to the chosen class, and (3) an image of the chosen class that was misclassified. This allows the user to assess whether, in one of three cases, the CNN learned an abstraction that is not applicable to the intended use-case. Furthermore, the user is also able to obtain the SM progression for a custom image. Using SMs, users are encouraged to identify patterns themselves as they would in a qualitative interview where the interviewer would be free to focus on any part of the response of the interviewee. As showcased in Figure 1, the user may observe that the CNN slowly learns to focus on the neck and head of the bird like humans would usually do to identify a bird.

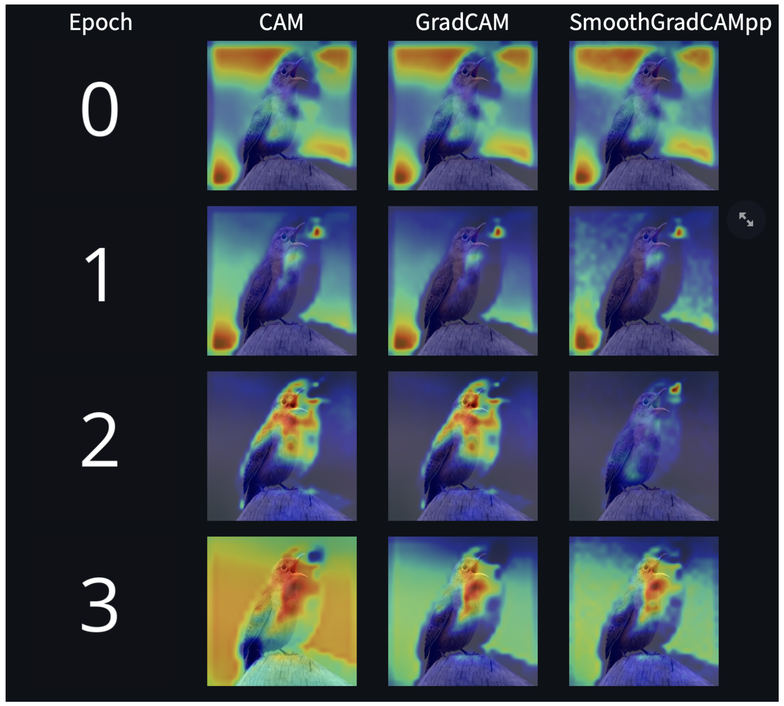


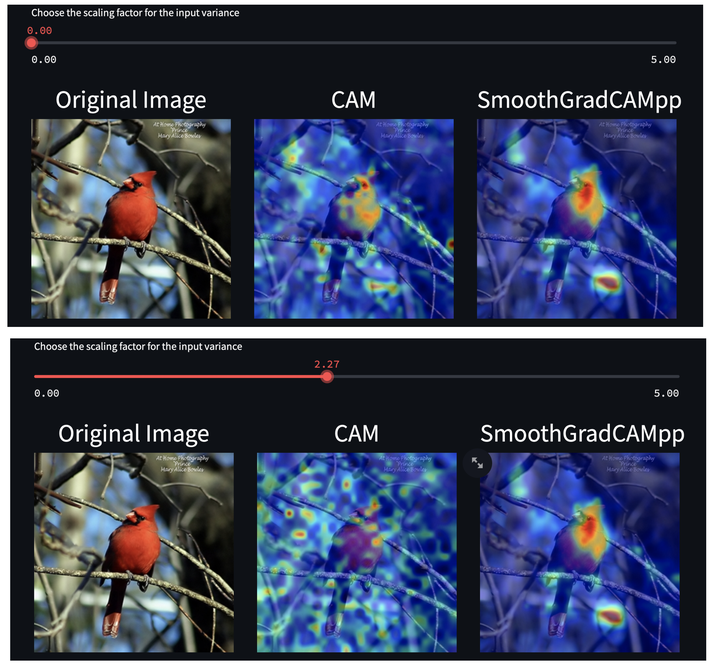
Figure 1. The side-by-side view compares different Saliency Maps techniques

Input variance fallacy

Input variance describes the phenomenon that the result of an Saliency Map algorithm may change when manipulating the CNN with a noise signal [8]. By adding the signal in the input layer and subtracting it in the first hidden layer the CNN is not changed effectively but the Saliency Map may. Therefore, mathematically identical models may result in different Saliency Maps.

The fellacy view

The fallacy view informs the user about the input variance and how strongly it affects the different Saliency Map techniques. As shown there, for the given image, the Saliency Map produced by the CAM algorithm becomes noisy when it is subject to input variance, while the Saliency Map produced by the SmoothGradCAMpp method stays relatively consistent. Hereby, the user may gain deeper insights, e.g., that they may trust the visualization provided by the SmoothGradCAMpp method more than from the basic CAM method. For it, the user obtains a more in-depth understanding of the provided visualizations.



References

[1] A. Blok and M. A. Pedersen, "Complementary social science? quali-quantitative experiments in a big data world," *Big Data & Society*, vol. 1, no. 2, pp. 1–6, 2014.

[2] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[3] C. Rudin, C. Chen, Z. Chen, H. Huang, L. Semenova, and C. Zhong, "Interpretable machine learning: Fundamental principles and 10 grand challenges," *Statistics Surveys*, vol. 16, pp. 1–85, 2022.

[4] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, IEEE Computer Society, 2016.

[5] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, 2020.

[6] D. Omeiza, S. Speakman, C. Cintas, and K. Weldermariam, "Smooth Grad-CAM++: An enhanced inference level visualization technique for deep convolutional neural network models," *CoRR cs.CV*, 2019.

[7] J. Zhou, A. H. Gandomi, F. Chen, and A. Holzinger, "Evaluating the quality of machine learning explanations: A survey on methods and metrics," *Electronics*, vol. 10, no. 5, pp. 1–19, 2021.

[8] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," in *Advances in Neural Information Processing Systems*, vol. 31, Curran Associates, Inc., 2018.