Identifying Trends in Feature Attributions during Training of Neural Networks *

Elena Terzieva[1,*] (✉), Maximilian Muschalik[1,2,*] (✉), Paul Hofman[1,2], and Eyke Hüllermeier[1,2]

[1] Institute of Informatics, LMU Munich, Germany
[2] MCML Munich Center for Machine Learning, Germany
✉ enterzieva@gmail.com, maximilian.muschalik@ifi.lmu.de
* denotes equal contribution

Abstract. This study investigates the evolving dynamics of commonly used feature attribution (FA) values during training of neural networks. As models transition from a state of high uncertainty to low uncertainty, we show that the features' significance also changes, which is inline with the general learning theory of deep neural networks. During model training, we compute FA scores through Layer-wise Relevance Propagation (LRP) and Gradient-weighted Class Activation Mapping (Grad-CAM), which are selected for their efficiency and speed of computation. We summarize the attribution scores in terms of the sum of the absolute values of FA scores and their entropy. We further analyze these summary scores in relation to the models' generalization capabilities. The analysis identifies trends where FA values increase in magnitude while entropy decreases during the training process, regardless of model generalization, suggesting independence of overfitting. This research offers a unique view on the application of FA methods in explainable artificial intelligence (XAI) and raises intriguing questions about their behavior across varying model architectures and datasets, which may have implications for future work combining XAI and uncertainty estimation in machine learning.

Keywords: Feature Attribution · Layer-wise Relevance Propagation · Gradient-weighted Class Activation Mapping · Model Performance Evaluation · Explainable Artificial Intelligence · Uncertainty in Artificial Intelligence

1 Introduction

Explainable artificial intelligence (XAI) techniques play a crucial role in unraveling the inner workings of opaque machine learning models [1,9]. One widely used category of XAI methods is feature attribution (FA), which quantifies the significance of specific inputs (features) in a model's predictions [9]. However, XAI measures could also be used to explain how a model evolves during training as it transitions from an initial state associated with high uncertainty to a

* We gratefully acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): TRR 318/1 2021 – 438445824.

state of lower uncertainty. In uncertainty quantification, aleatoric and epistemic uncertainty are usually distinguished [4]. The former describing the irreducible uncertainty caused by the stochastic nature of the data and the latter describing the uncertainty in the model, which is reduced by more data. Recently, epistemic uncertainty has been further decomposed into model uncertainty, which is uncertainty regarding the choice of model or the hypothesis space, and approximation uncertainty, which is the distance between the current model and the best model in the hypothesis space [3]. One can think of empirical risk minimization as a proxy of reducing approximation uncertainty. However, as the empirical risk is minimized, the parameters of a model are changed, which means the explanations of the model could also change. Thus, the question arises how the explanations of XAI methods change as the approximation uncertainty of a model is reduced and how this change relates to the generalization performance of the underlying models. To this end, we study feature attribution methods as a function of training time.

Contribution. In this context, our main contribution include

1. a descriptive analysis[3] of the behavior of FA values during the training stage of a neural network and
2. an investigation of the effect of overfitting on this evolution of FA scores showing that the generalization capabilities of a model do *not* effect the trends in the underlying FA scores.

2 Backpropagation-based Feature Attribution Methods

One of the most widely used model-specific FA approaches are *backpropagation-based methods*, which compute importance scores by backpropagating some evaluation of the gradient signals from the output to the input layer. Since the gradients have to be computed to train the model, they are a suitable choice for producing explanations during the training process. An FA explanation can be computed through one pass through the neural network. Different backpropagation-based approaches exist [2,9] and two well-established techniques are Layer-wise Relevance Propagation [6] and Gradient-weighted Class Activation Mapping [10].

Layer-wise Relevance Propagation (LRP). LRP [6] performs a forward pass to compute the activation at each layer, where the activation at the last layer is the prediction. Afterward the output relevance score is backpropagated through the network. The key property of this method is *relevance conservation*, which ensures that no relevance is lost or added. The backward pass employs distinctive rules for different layers to account for the entanglements between the target concepts and the spurious variations that stem from weight sharing.

[3] The source code of the descriptive analysis conducted in Section 3 and Section 4 is publicly available at https://github.com/EliTerzieva1995/Identifying-Trends-in-Feature-Attributions-during-Training-of-Neural-Networks. This repository also contains the appendix and further supplementary material of this work.

Gradient-weighted Class Activation Mapping (Grad-CAM). Grad-CAM [10] is a method used to generate visual explanations for convolutional neural networks. It involves computing the gradients of the output score for a specific class with respect to the activations of feature maps. These gradients are then globally averaged and used to obtain the neuron importance weights. The relevance score map is created by combining the activation maps weighted by the neuron importance score. This map highlights regions of positive influence, resulting in a coarse heatmap with dimensions matching the convolutional feature maps.

3 Explaining the Model Training

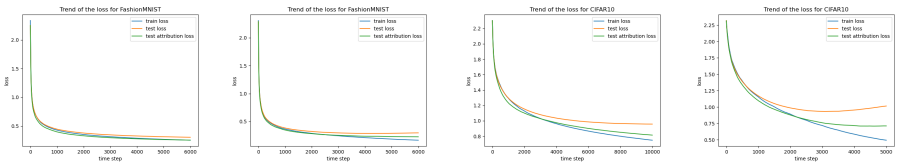
Both of the aforementioned methods, LRP and Grad-CAM, produce a local explanation of a given image instance in the form of a matrix of attribution values. We select LRP and Grad-CAM for our investigation, since they can be efficiently computed throughout the training procedure of neural networks. To investigate the behavior of these FA methods while the model is learning, the attribution values are computed at different time steps during the training process. These attribution maps are then described by two summary statistics. Firstly, the Frobenius *norm* is computed by $\|A\|_F = \sqrt{\sum_i^m \sum_j^n |a_{ij}|^2}$ where a_{ij} represents the attribution value of a particular input (i.e. feature). Secondly, we represent the attribution scores as probability distributions by normalizing them by their sum and we compute the Shannon *entropy* $H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$. A higher entropy corresponds to a more uniform distribution of the importance scores, whereas a lower entropy signifies a more peaked distribution over the attribution values, which would suggest that certain pixels are associated with higher attribution values, while other pixels have a lower attribution value. Note that this does not entail any spatial information about the attribution values.

We compute these summary statistics jointly on portions of the train and test subsets of the datasets used. We furthermore perform a correlation analysis using Spearman's rank coefficient. We use this method, because it does not make assumptions about the type of monotonic function that describes the relationship between two variables.

4 Experiments

We experimentally evaluate different neural network-based image classifiers to explore the relationship between FA scores, different model performance metrics and the models' generalization capabilities.

Setup. We conduct our experiments on the train and test subsets of Fashion-MNIST [12] and CIFAR10 [5]. Additionally, we create a separate *attribution* subset. The attribution set consists of 10 randomly selected images from the training set and 10 images from the test split from each class in the dataset. For each dataset, we fit two neural network architectures on the training data. Note



FashionMNIST no overfit (a) | FashionMNIST slight overfit (b) | CIFAR10 overfit (c) | CIFAR10 strong overfit (d)

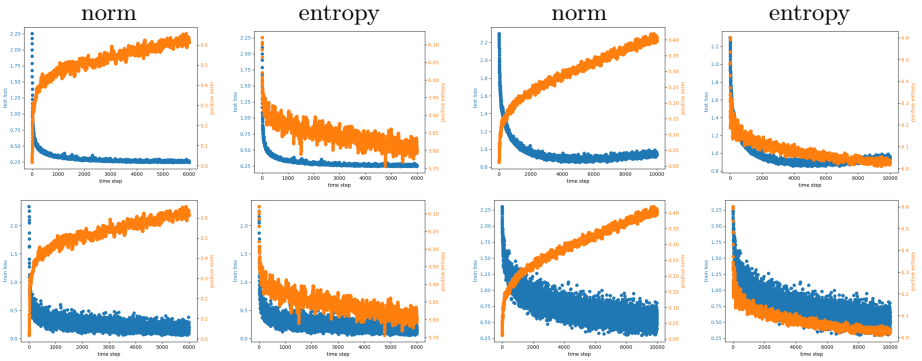
Fig. 1: Trend of the loss during training, testing and on a mixed subset with 50% train and 50% test observations (attribution set) on FashionMNIST given the non-overfitting model (a) and the slightly overfitting model (b). Figures (c) and (d) show the overfitting and the strong overfitting models on CIFAR10, respectively.

that we, intentionally, let the networks overfit on the datasets to investigate how the FA values behave in scenarios with limited generalization performance (c.f. Fig. 1 and Appendix, Fig. 8 and Fig. 9. We observe the predictive performance on all three subsets in terms of accuracy and cross-entropy loss after each model update (i.e. after each batch iteration).[4] During the training of the networks, we further observe the FA scores by computing the LRP and Grad-CAM values on all instances of the attribution dataset. Likewise to the performance metrics, we compute the FA values after each model update. The LRP procedure follows the strategy as described by Montavon et al. [8], where different rules are applied at different layers in the model. Grad-CAM is applied at the last convolutional layer as recommended by Selvaraju et al. [10]. Lastly, we summarize the attribution vectors with their norm and entropy (as presented in Section 3). Since we are interested in the attribution scores that contribute to the correct prediction of the model, we only consider the positive proportion of attribution values.[5]

Results. We conduct a descriptive analysis of LRP and Grad-CAM values for the FashionMNIST and CIFAR10 datasets. Fig. 2 clearly demonstrates trends for the LRP values which do not materialize as such for the Grad-CAM values as demonstrated in Fig. 3. Consequently, we focus exclusively on the LRP experiments. As the model's train and test loss decreases, the norm of LRP values increases. This indicates that as the model learns, the attribution values become more prominent, signifying an increase in feature importance. Moreover, as the model's training progresses, the entropy of FA values decreases. This coincides with the norm of the attribution values, but additionally shows that the uncertainty regarding the attribution values decreases. Fewer features receive high attribution values while the remaining features receive low attribution values.

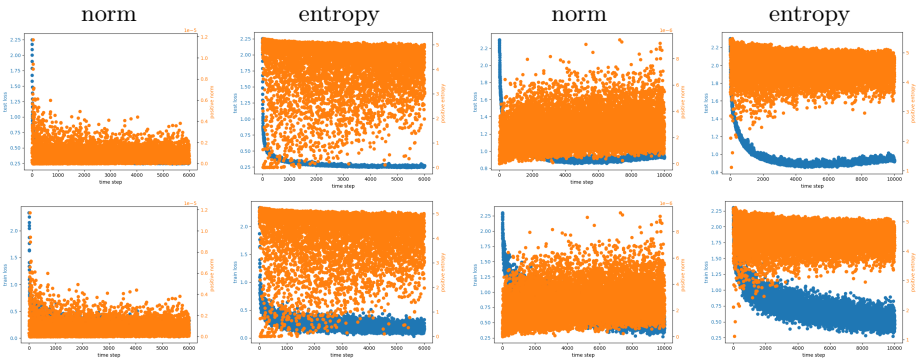
[4] For a detailed description of the models used and the datasets we refer to the supplementary material (Section A.1 and Section A.2.

[5] For details on the negative FA values we refer to the appendix.



(a) FashionMNIST (good fit) (b) CIFAR10 (slight overfit)

Fig. 2: Trend of the summary measures (orange; norm and entropy) of the pos. **LRP** values and the model performance (blue) for the test loss (top row) and the train loss (bottom row) on FashionMNIST (a) and CIFAR10 (b). The LRP-based summary measures follow a clear trend.



(a) FashionMNIST (good fit) (b) CIFAR10 (slight overfit)

Fig. 3: Trend of the summary measures (orange; norm and entropy) of the pos. **Grad-CAM** values and the model performance (blue) for the test loss (top row) and the train loss (bottom row) on FashionMNIST (a) and CIFAR10 (b). Compared to the LRP-based summary measures the Grad-CAM-based measures do not follow a clear trend and scatter throughout the training procedure.

Section. B.2 contains the evolution of FA summary measures for the different models and generalization performances. The detected trends of the summary measures appear to be independent of the model's generalization capabilities, meaning they remain consistent regardless of whether or not the model is overfitting on the data.

Table 1: Spearman's rank coefficient ρ of the LRP and Grad-CAM based summary measures and losses on FashionMNIST and CIFAR10. Coefficients with an absolute value above 0.5 are highlighted in bold.

FA	dataset	FashionMNIST				CIFAR10			
	model fit	good fit		slight overfit		overfit		strong overfit	
	summary	norm	entropy	norm	entropy	norm	entropy	norm	entropy
LRP	test loss	**-0.875**	**0.806**	0.226	-0.141	-0.363	0.391	0.345	-0.296
	train loss	**-0.657**	**0.591**	**-0.836**	**0.783**	**-0.832**	**0.808**	**-0.940**	**0.884**
Grad CAM	test loss	0.040	0.118	0.158	0.022	-0.086	0.055	0.035	-0.022
	train loss	0.129	0.075	0.261	0.198	-0.108	0.050	-0.031	-0.020

Lastly, we conduct a correlation analysis presented in Table 1. Regarding the well-fitted FashionMNIST model, we observe a significant negative correlation between the train and test loss and the norms of the FA values. As the model's performance improves and the loss decreases, the magnitude of FA values increases. Conversely, with progressing training and smaller losses, the entropy declines. In the case of an overfitting model, the correlation between the test loss and the summary measures behaves differently. The test loss correlations become considerably weaker or change directions.

5 Conclusion and Future Work

The experimental study demonstrates a clear trend of the LRP values in the progression of a model's training. Over time, the magnitude of LRP values increases while the entropy of attribution scores decreases. In contrast to LRP, Grad-CAM does not exhibit a similarly clear trend. The evaluation suggests that this trend is independent of the generalization capabilities of the models, meaning both overfitted and non-overfitted models exhibit similar behavior. However, given the limited scope of this study, future work is needed to comprehensively evaluate this behavior across various model architectures, datasets, and training times.

Uncertainty Quantification (UQ). Moreover, we consider this work to be a preliminary study into the combined field of UQ and XAI. While both UQ and XAI are well-established fields individually, the combined field is emerging. As such, further research into this topic is necessary. We argue that this field can be approached in at least two ways. Firstly, through the development of tools aimed at quantifying the uncertainty in XAI methods, thus answering the question: "How (un)certain is the model about this explanation?" [7]. Secondly, UQ and XAI can be combined by using XAI methods to explain uncertainty, thereby answering the question: "Why is the model uncertain about this prediction?" [11]. Answering both of these questions requires an already trained model, yet with this analysis, we propose a first step in merging UQ and XAI at learning time. Future research may aim to answer questions such as "How does the model's uncertainty change over time and different data?".

References

1. Adadi, A., Berrada, M.: Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). IEEE Access **6**, 52138–52160 (2018). https://doi.org/10.1109/ACCESS.2018.2870052
2. Ancona, M., Ceolini, E., Öztireli, C., Gross, M.: "Gradient-Based Attribution Methods", pp. 169–191. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-28954-6_9
3. Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. Machine Learning **110**(3), 457–506 (2021). https://doi.org/10.1007/s10994-021-05946-3
4. Kendall, A., Gal, Y.: What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? CoRR **abs/1703.04977** (2017)
5. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech Report (2009)
6. Lapuschkin, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. PLoS ONE **10**, e0130140 (07 2015). https://doi.org/10.1371/journal.pone.0130140
7. Löfström, H., Löfström, T., Johansson, U., Sönströd, C.: Calibrated Explanations: with Uncertainty Information and Counterfactuals. CoRR **abs/2305.02305** (2023)
8. Montavon, G., Binder, A., Lapuschkin, S., Samek, W., Müller, K.: Layer-Wise Relevance Propagation: An Overview. In: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Lecture Notes in Computer Science, vol. 11700, pp. 193–209. Springer (2019). https://doi.org/10.1007/978-3-030-28954-6_10
9. Schwalbe, G., Finzel, B.: A comprehensive taxonomy for explainable artificial intelligence: a systematic survey of surveys on methods and concepts. Data Mining and Knowledge Discovery (2023). https://doi.org/10.1007/s10618-022-00867-8
10. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In: IEEE International Conference on Computer Vision, ICCV. pp. 618–626. IEEE Computer Society (2017). https://doi.org/10.1109/ICCV.2017.74
11. Watson, D.S., O'Hara, J., Tax, N., Mudd, R., Guy, I.: Explaining Predictive Uncertainty with Information Theoretic Shapley Values. CoRR **abs/2306.05724** (2023)
12. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. CoRR **abs/1708.07747** (2017)

A Summary of Datasets and Models

A.1 Dataset Description

FashionMNIST is a dataset of Zalando's article images. It consists of 60,000 training samples and a test set of 10,000 samples. Each image is a 28x28 grayscale image, associated with a label from 10 classes. The different labels are one of the following:

- 0 T-shirt/top
- 1 Trouser
- 2 Pullover
- 3 Dress
- 4 Coat
- 5 Sandal
- 6 Shirt
- 7 Sneaker
- 8 Bag
- 9 Ankle boot

The CIFAR10 dataset consists of 60,000 images. The train set is comprised of 50,000 samples and the test set has 10,000 samples. Each image is a 32x32 color image and can be one from 10 different classes. The labels can be one of the following:

- 0 Airplane
- 1 Automobile
- 2 Bird
- 3 Cat
- 4 Deer
- 5 Dog
- 6 Frog
- 7 Horse
- 8 Ship
- 9 Truck

A.2 Model Description

The experiments were conducted using two different architectures. The first neural network is more simple, consisting of 7 hidden layers, while the second one is more complex, consisting of 18 hidden layers. Due to the choice of approaches, both architectures include convolutional layers, followed by a ReLU activation function and max pooling layers. The shallow models for the FashionMNIST and CIFAR10 dataset are summarized in Fig 4 and Fig. 6. The deeper models are described in Fig 5 and Fig. 7.

```
Model(
  (features_conv): Sequential(
    (0): Conv2d(1, 16, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (1): ReLU()
    (2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (3): Conv2d(16, 32, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (4): ReLU()
  )
  (pool2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (fc1): Linear(in_features=1568, out_features=10, bias=True)
)
```

Fig. 4: Architecture of the more shallow neural network for FashionMNIST.

```
ComplexModel(
  (features_conv): Sequential(
    (0): Conv2d(1, 32, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (1): ReLU()
    (2): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (3): ReLU()
    (4): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (5): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (6): ReLU()
    (7): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (8): ReLU()
    (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (10): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (11): ReLU()
    (12): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (13): ReLU()
  )
  (pool2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (fc1): Linear(in_features=1152, out_features=1024, bias=True)
  (relu): ReLU()
  (fc2): Linear(in_features=1024, out_features=10, bias=True)
)
```

Fig. 5: Architecture of the deeper neural network for FashionMNIST.

```
ModelCIFAR10(
  (features_conv): Sequential(
    (0): Conv2d(3, 16, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (1): ReLU()
    (2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (3): Conv2d(16, 32, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (4): ReLU()
  )
  (pool2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (fc1): Linear(in_features=2048, out_features=10, bias=True)
)
```

Fig. 6: Architecture of the more shallow neural network for CIFAR10.

```
ComplexModelCIFAR10(
  (features_conv): Sequential(
    (0): Conv2d(3, 32, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (1): ReLU()
    (2): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (3): ReLU()
    (4): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (5): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (6): ReLU()
    (7): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (8): ReLU()
    (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (10): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (11): ReLU()
    (12): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=same)
    (13): ReLU()
  )
  (pool2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (fc1): Linear(in_features=2048, out_features=1024, bias=True)
  (relu): ReLU()
  (fc2): Linear(in_features=1024, out_features=10, bias=True)
)
```

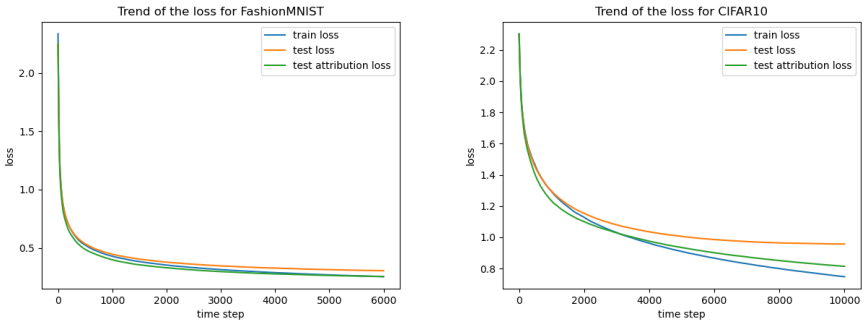
Fig. 7: Architecture of the deeper neural network for CIFAR10.

B Additional Experimental Results

This section holds details on the experimental evaluation and further results.

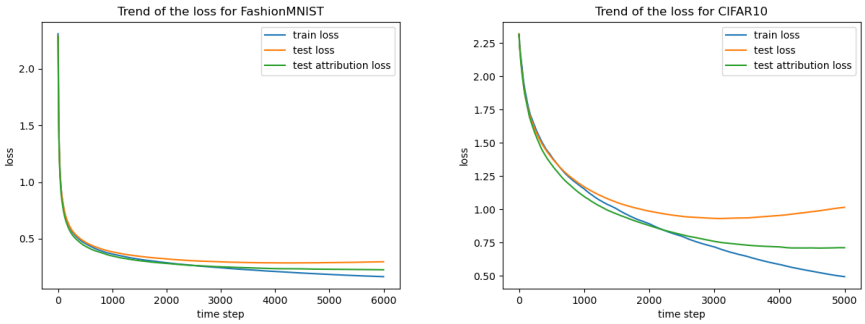
B.1 Training Summary.

Fig. 8 and Fig. 9 demonstrate the models' training progress for the shallow and the deeper models. Note that the models are overfitting on CIFAR10. The deeper model is also overfitting on FashionMNIST towards the end of training.



(a) FashionMNIST (b) CIFAR10

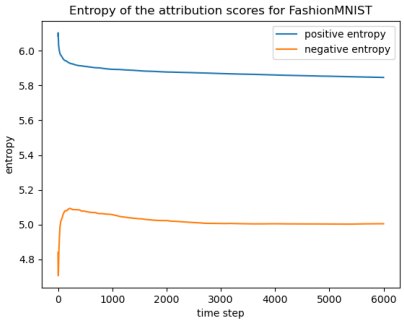
Fig. 8: Trend of the loss during training, testing and on the attribution dataset on FashionMNIST (a) and CIFAR10 (b) of the simple model.



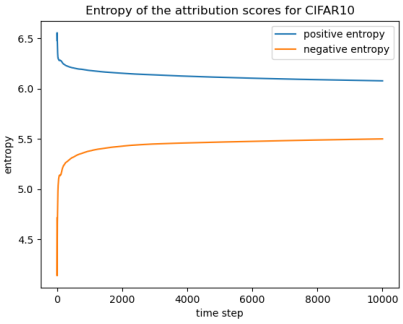
(a) FashionMNIST (b) CIFAR10

Fig. 9: Trend of the loss during training, testing and on the attribution dataset on FashionMNIST (a) and CIFAR10 (b) of the complex model.

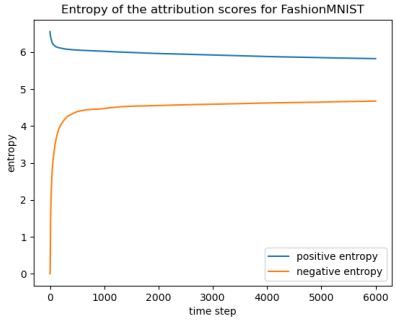
B.2 Progression of Summary Measures during Training



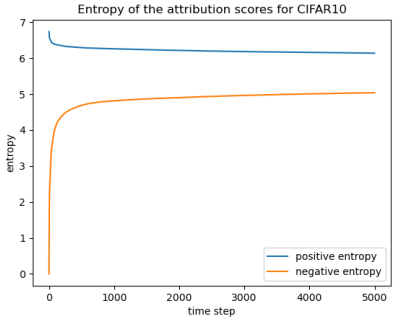
(a) FashionMNIST simple model



(b) CIFAR10 simple model

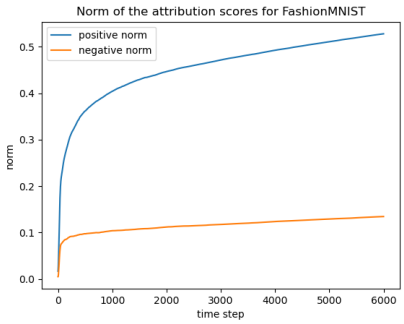


(c) FashionMNIST complex model

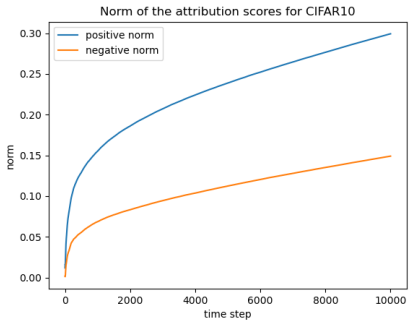


(d) CIFAR10 complex model

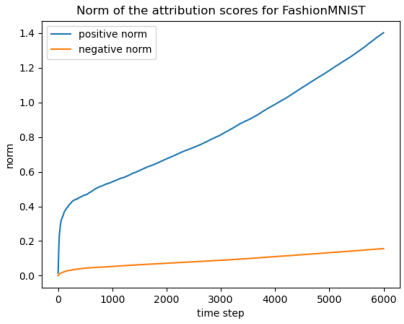
Fig. 10: Evolution of the entropy of the positive and negative LRP values on FashionMNIST and CIFAR10.



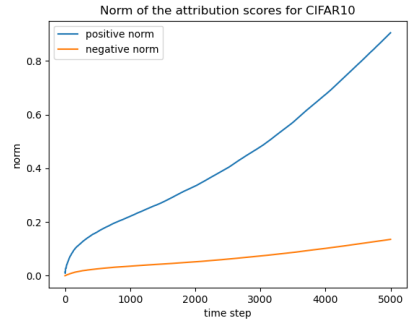
(a) FashionMNIST simple model



(b) CIFAR10 simple model

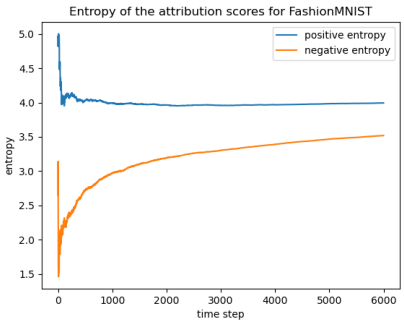


(c) FashionMNIST complex model

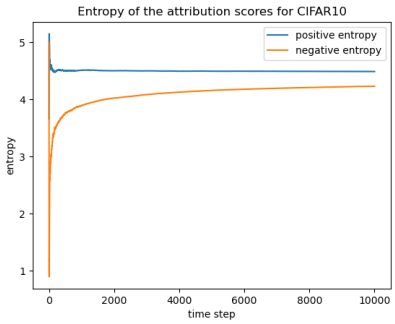


(d) CIFAR10 complex model

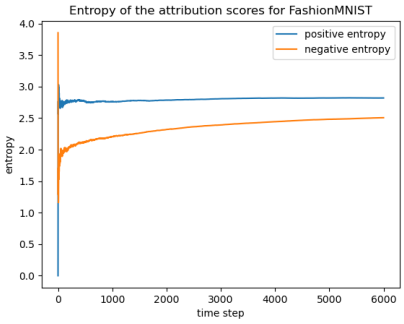
Fig. 11: Evolution of the norm of the positive and negative LRP values on FashionMNIST and CIFAR10.



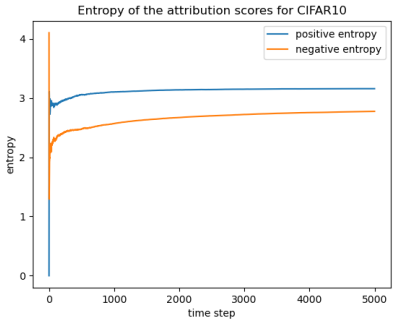
(a) FashionMNIST simple model



(b) CIFAR10 simple model

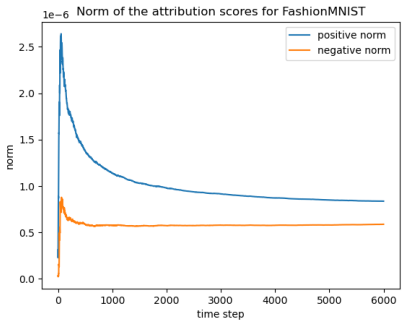


(c) FashionMNIST complex model

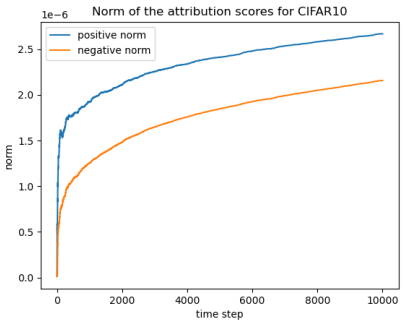


(d) CIFAR10 complex model

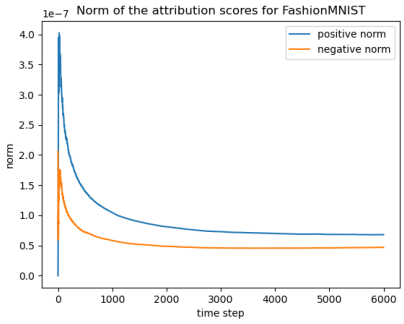
Fig. 12: Evolution of the entropy of the positive and negative Grad-CAM values on FashionMNIST and CIFAR10.



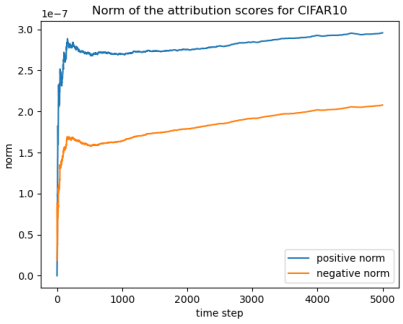
(a) FashionMNIST simple model



(b) CIFAR10 simple model



(c) FashionMNIST complex model



(d) CIFAR10 complex model

Fig. 13: Evolution of the norm of the positive and negative Grad-CAM values on FashionMNIST and CIFAR10.

B.3 Additional Results for LRP

Fig. 14, 15, 16, and 17 show all results for the FashionMNIST and CIFAR10 datasets with the shallow and deeper model and the LRP attribution values.

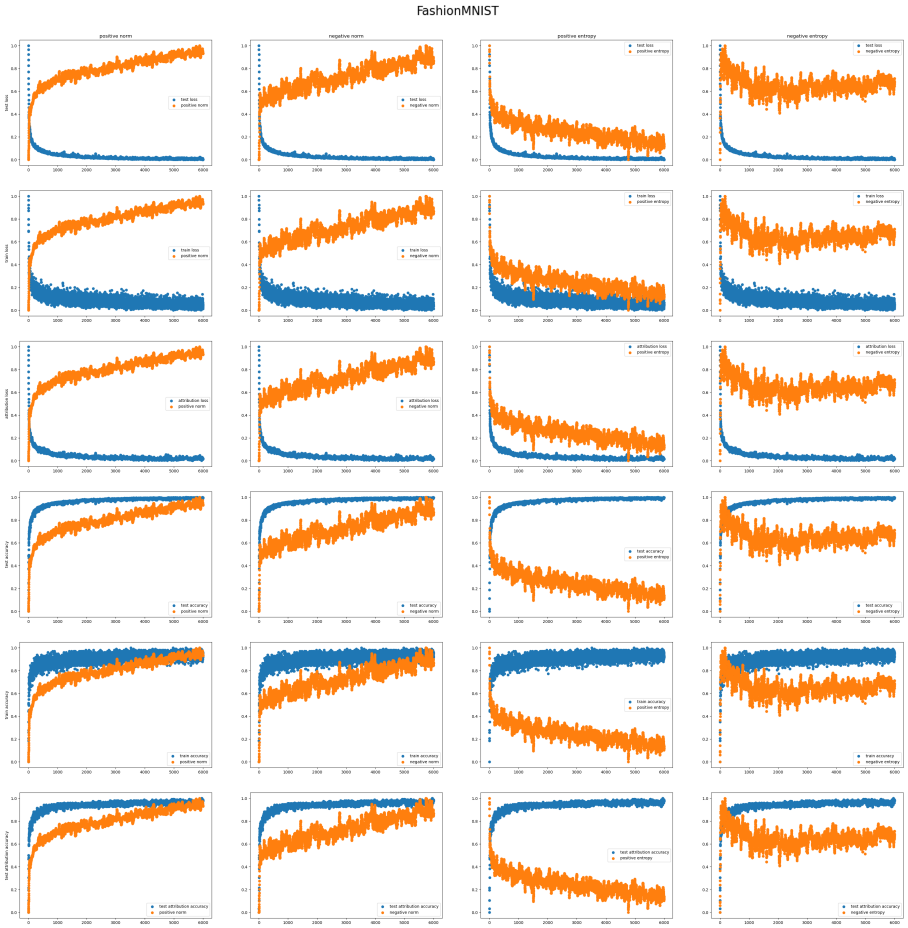


Fig. 14: Plots of the normalized LRP summary measures and normalized performance metrics evolution over time on FashionMNIST of the **shallow** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

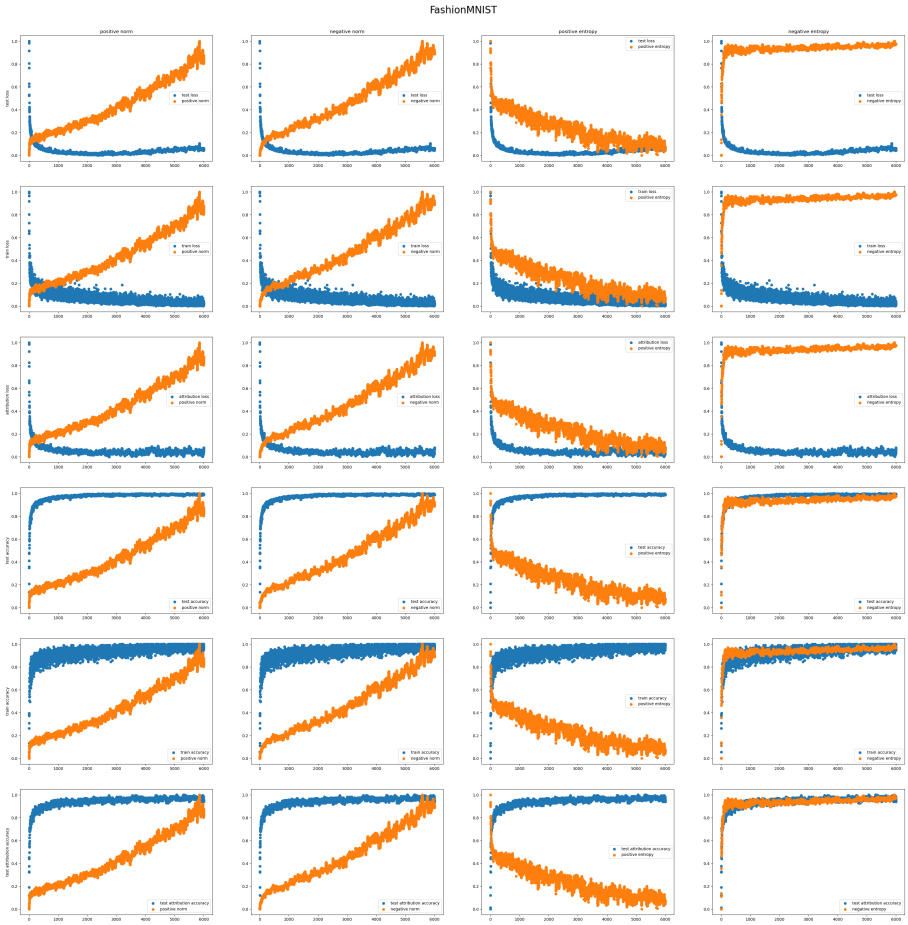


Fig. 15: Plots of the normalized LRP summary measures and normalized performance metrics evolution over time on FashionMNIST of the **deeper** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

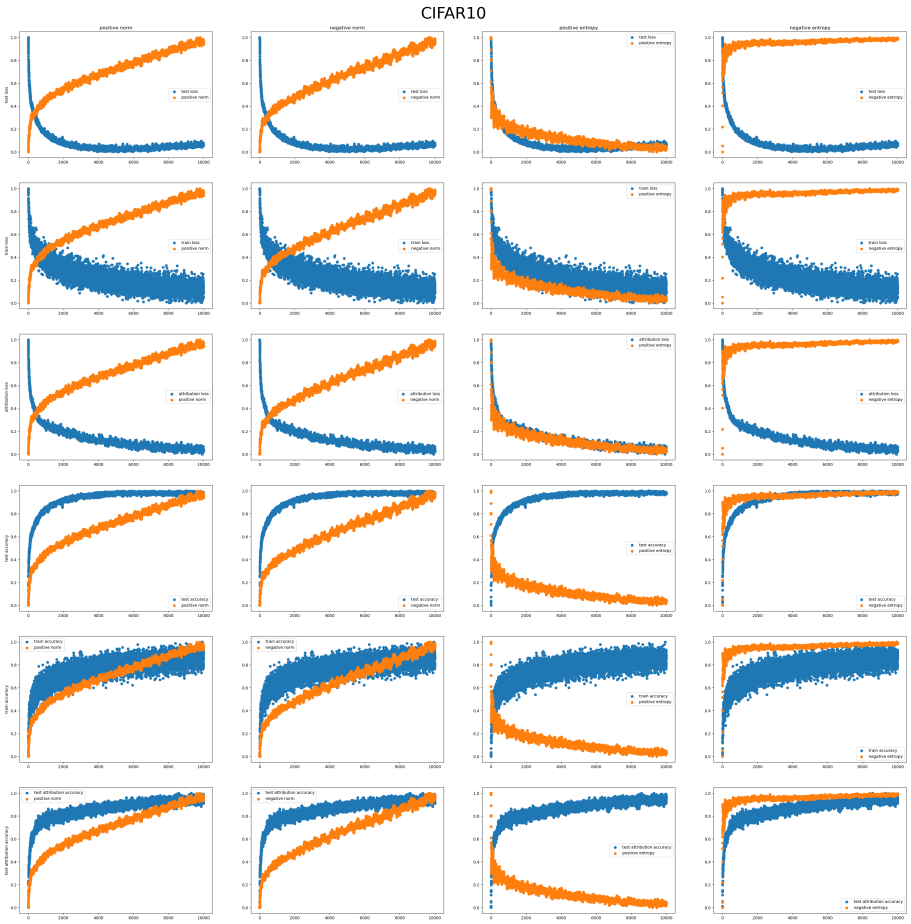


Fig. 16: Plots of the normalized LRP summary measures and normalized performance metrics evolution over time on CIFAR10 of the **shallow** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

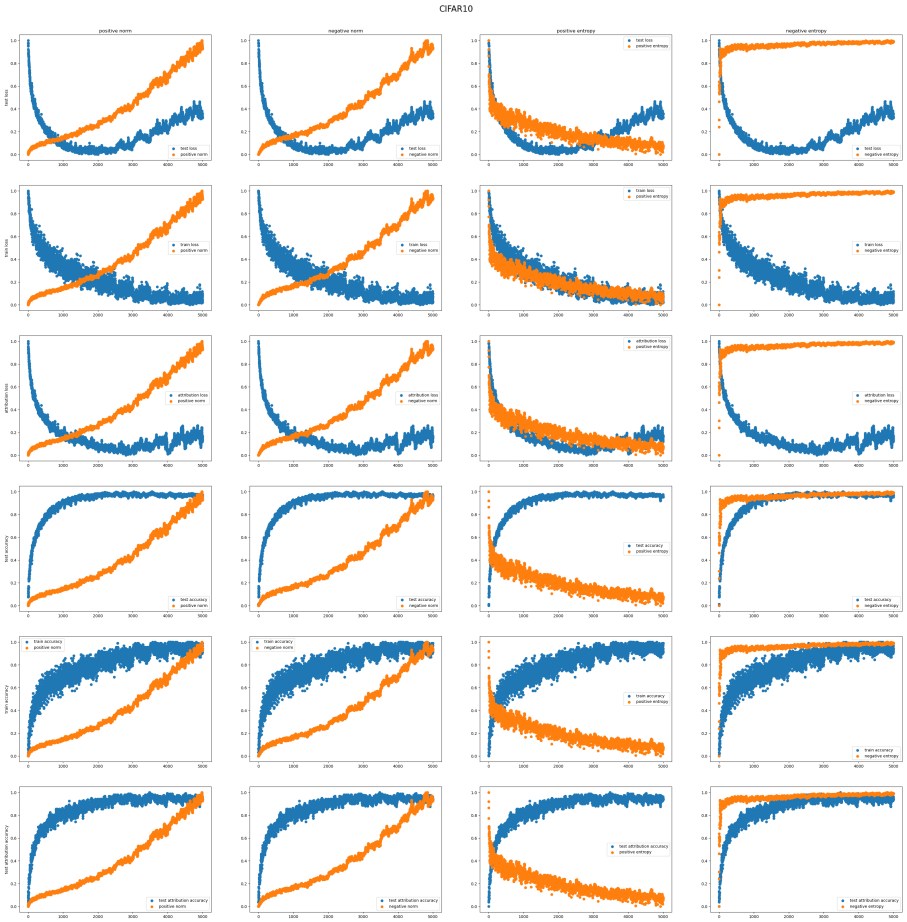


Fig. 17: Plots of the normalized LRP summary measures and normalized performance metrics evolution over time on CIFAR10 of the **deeper** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

B.4 Additional Results for Grad-CAM

Fig. 18, 19, 20, and 21 show all results for the FashionMNIST and CIFAR10 datasets with the shallow and deeper model and the Grad-CAM attribution values.

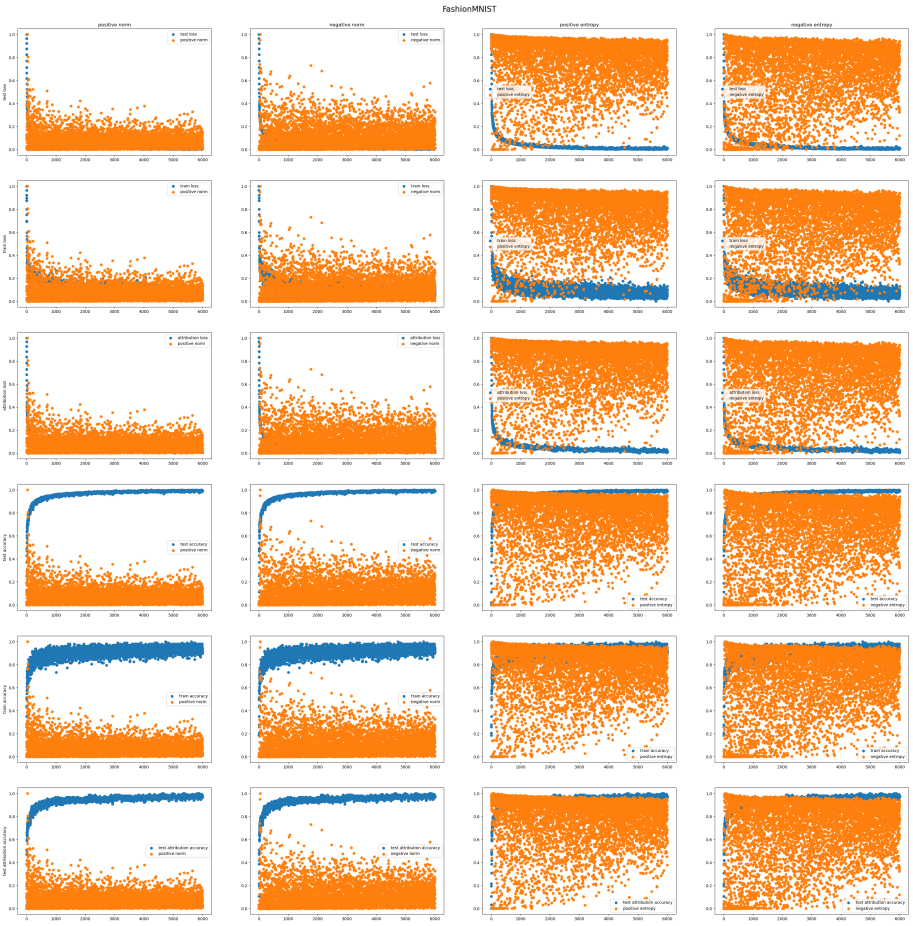


Fig. 18: Plots of the normalized Grad-CAM summary measures and normalized performance metrics evolution over time on FashionMNIST of the **shallow** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

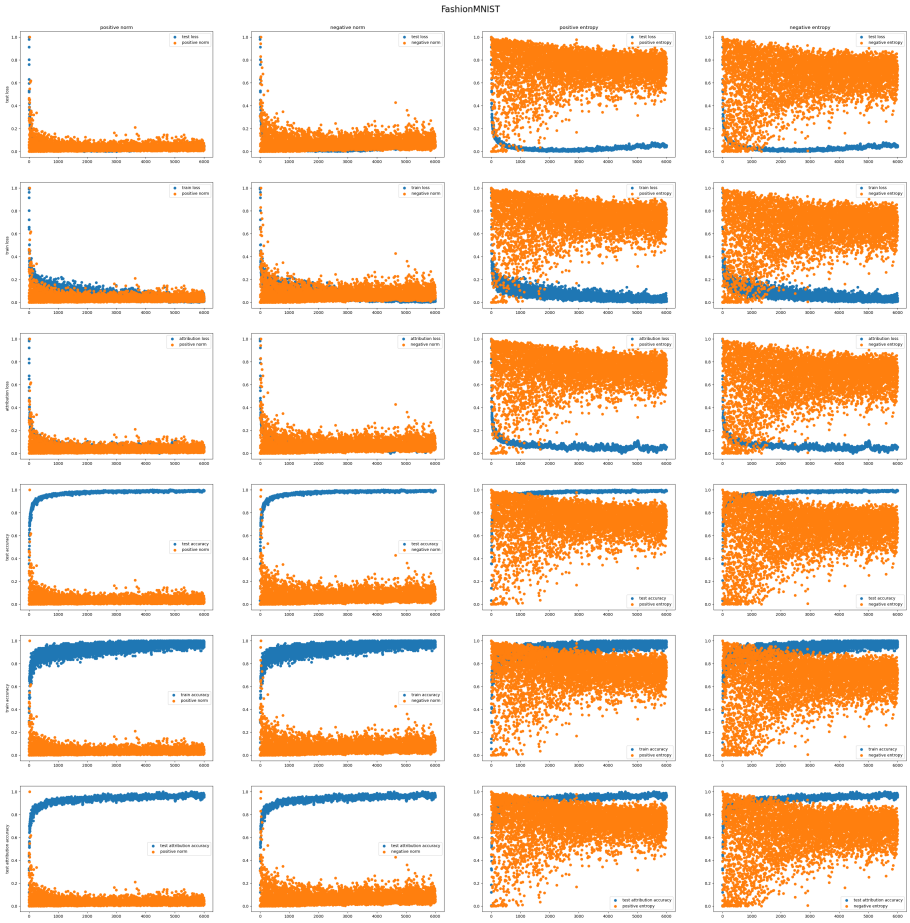


Fig. 19: Plots of the normalized Grad-CAM summary measures and normalized performance metrics evolution over time on FashionMNIST of the **deeper** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

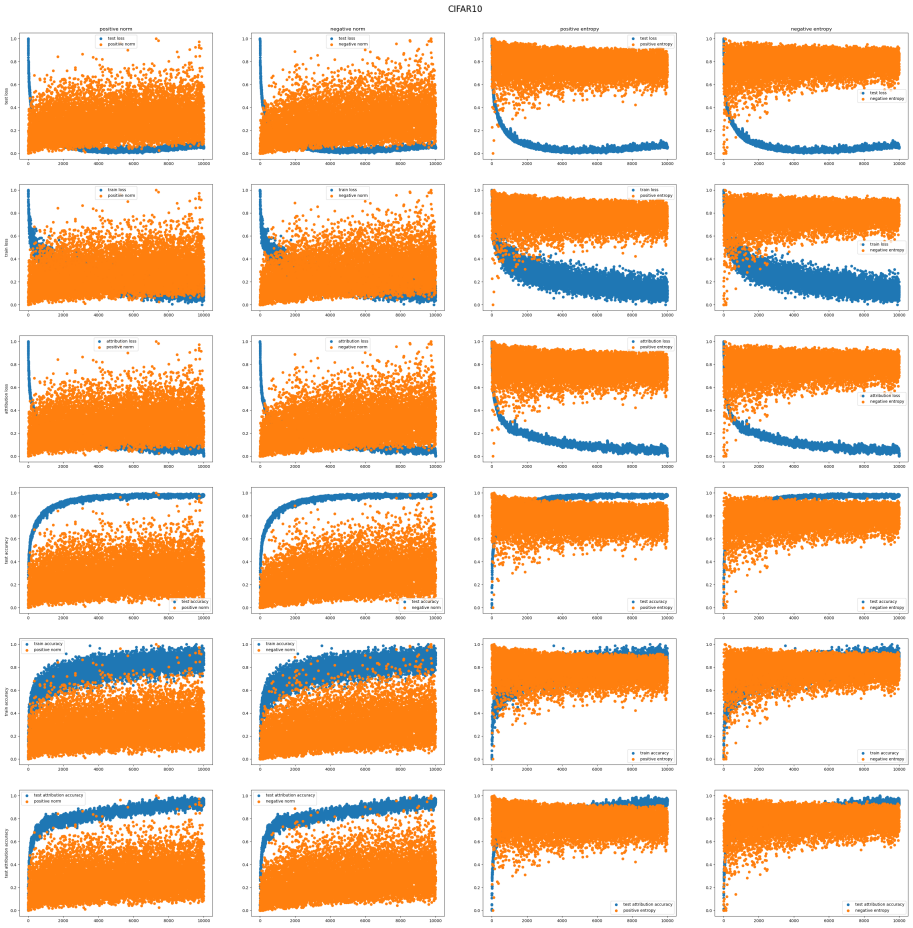


Fig. 20: Plots of the normalized Grad-CAM summary measures and normalized performance metrics evolution over time on CIFAR10 of the **shallow** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

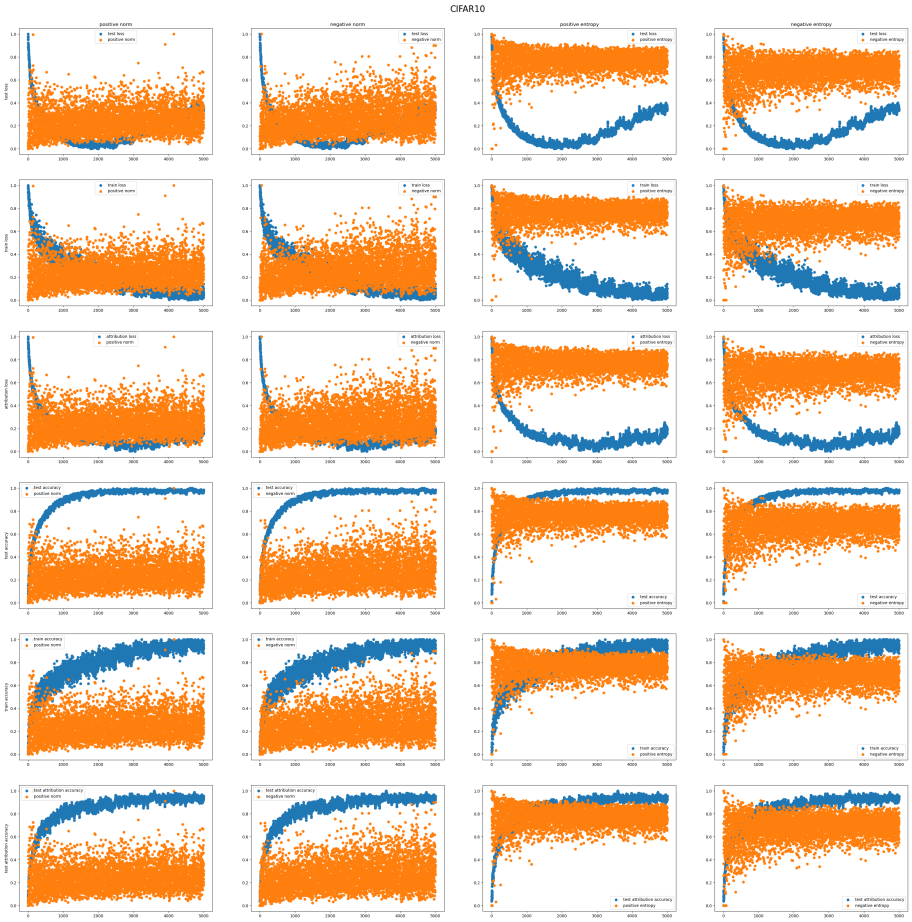


Fig. 21: Plots of the normalized Grad-CAM summary measures and normalized performance metrics evolution over time on CIFAR10 of the **deeper** model. Each column corresponds to one of the summary measures and each row corresponds to one of the performance metrics.

B.5 Complete Results of the Correlation Analysis

Table. 2, 3, 4, 5, 6, 7, 8, and 9 holds the correlation coefficients of all model and dataset combinations.

FashionMNIST				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = -0.875$	$\rho = -0.836$	$\rho = 0.806$	$\rho = 0.123$
train loss	$\rho = -0.657$	$\rho = -0.629$	$\rho = 0.591$	$\rho = 0.161$
attribution loss	$\rho = -0.779$	$\rho = -0.751$	$\rho = 0.690$	$\rho = 0.165$
test accuracy	$\rho = 0.903$	$\rho = 0.867$	$\rho = -0.808$	$\rho = -0.108$
train accuracy	$\rho = 0.581$	$\rho = 0.557$	$\rho = -0.535$	$\rho = -0.159$
test attribution accuracy	$\rho = 0.751$	$\rho = 0.724$	$\rho = -0.688$	$\rho = -0.171$

Table 2: Spearman's rank coefficient ρ of the LRP summary and normalized performance measures on FashionMNIST of the simple model.

CIFAR10				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = -0.363$	$\rho = -0.365$	$\rho = 0.391$	$\rho = -0.288$
train loss	$\rho = -0.832$	$\rho = -0.833$	$\rho = 0.808$	$\rho = -0.759$
attribution loss	$\rho = -0.952$	$\rho = -0.951$	$\rho = 0.929$	$\rho = -0.867$
test accuracy	$\rho = 0.839$	$\rho = 0.841$	$\rho = -0.829$	$\rho = 0.760$
train accuracy	$\rho = 0.778$	$\rho = 0.778$	$\rho = -0.758$	$\rho = 0.709$
test attribution accuracy	$\rho = 0.929$	$\rho = 0.928$	$\rho = -0.895$	$\rho = 0.822$

Table 3: Spearman's rank coefficient ρ of the LRP summary and normalized performance measures on CIFAR10 of the simple model.

FashionMNIST				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = 0.226$	$\rho = 0.220$	$\rho = -0.141$	$\rho = 0.233$
train loss	$\rho = -0.836$	$\rho = -0.835$	$\rho = 0.783$	$\rho = -0.619$
attribution loss	$\rho = -0.420$	$\rho = -0.415$	$\rho = 0.438$	$\rho = -0.275$
test accuracy	$\rho = 0.770$	$\rho = 0.774$	$\rho = -0.702$	$\rho = 0.535$
train accuracy	$\rho = 0.780$	$\rho = 0.779$	$\rho = -0.731$	$\rho = 0.576$
test attribution accuracy	$\rho = 0.812$	$\rho = 0.811$	$\rho = -0.772$	$\rho = 0.600$

Table 4: Spearman's rank coefficient ρ of the LRP summary and normalized performance measures on FashionMNIST of the deeper model.

CIFAR10				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = 0.345$	$\rho = 0.347$	$\rho = -0.296$	$\rho = 0.310$
train loss	$\rho = -0.940$	$\rho = -0.939$	$\rho = 0.884$	$\rho = -0.864$
attribution loss	$\rho = -0.383$	$\rho = -0.380$	$\rho = 0.391$	$\rho = -0.383$
test accuracy	$\rho = 0.673$	$\rho = 0.670$	$\rho = -0.635$	$\rho = 0.609$
train accuracy	$\rho = 0.925$	$\rho = 0.924$	$\rho = -0.869$	$\rho = 0.850$
test attribution accuracy	$\rho = 0.827$	$\rho = 0.827$	$\rho = -0.796$	$\rho = 0.777$

Table 5: Spearman's rank coefficient ρ of the LRP summary and normalized performance measures on CIFAR10 of the deeper model.

FashionMNIST				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = 0.040$	$\rho = -0.129$	$\rho = 0.118$	$\rho = -0.076$
train loss	$\rho = 0.129$	$\rho = 0.012$	$\rho = 0.075$	$\rho = -0.043$
attribution loss	$\rho = 0.075$	$\rho = -0.116$	$\rho = 0.161$	$\rho = -0.122$
test accuracy	$\rho = -0.047$	$\rho = 0.148$	$\rho = -0.132$	$\rho = 0.090$
train accuracy	$\rho = -0.124$	$\rho = -0.027$	$\rho = -0.064$	$\rho = 0.036$
test attribution accuracy	$\rho = -0.051$	$\rho = 0.143$	$\rho = -0.130$	$\rho = 0.091$

Table 6: Spearman's rank coefficient ρ of the Grad-CAM summary and normalized performance measures on FashionMNIST of the simple model.

CIFAR10				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = -0.086$	$\rho = -0.204$	$\rho = 0.055$	$\rho = -0.120$
train loss	$\rho = -0.108$	$\rho = -0.209$	$\rho = 0.050$	$\rho = -0.109$
attribution loss	$\rho = -0.221$	$\rho = -0.324$	$\rho = 0.064$	$\rho = -0.122$
test accuracy	$\rho = 0.168$	$\rho = 0.318$	$\rho = -0.084$	$\rho = 0.145$
train accuracy	$\rho = 0.102$	$\rho = 0.181$	$\rho = -0.036$	$\rho = 0.093$
test attribution accuracy	$\rho = 0.203$	$\rho = 0.343$	$\rho = -0.079$	$\rho = 0.135$

Table 7: Spearman's rank coefficient ρ of the Grad-CAM summary and normalized performance measures on CIFAR10 of the simple model.

FashionMNIST				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = 0.158$	$\rho = 0.111$	$\rho = 0.022$	$\rho = -0.029$
train loss	$\rho = 0.261$	$\rho = 0.004$	$\rho = 0.198$	$\rho = -0.156$
attribution loss	$\rho = 0.090$	$\rho = -0.099$	$\rho = 0.131$	$\rho = -0.096$
test accuracy	$\rho = -0.074$	$\rho = 0.102$	$\rho = -0.131$	$\rho = 0.087$
train accuracy	$\rho = -0.269$	$\rho = -0.0265$	$\rho = -0.189$	$\rho = 0.149$
test attribution accuracy	$\rho = -0.047$	$\rho = 0.116$	$\rho = -0.121$	$\rho = 0.078$

Table 8: Spearman's rank coefficient ρ of the Grad-CAM summary and normalized performance measures on FashionMNIST of the deeper model.

CIFAR10				
measures	positive norm	negative norm	positive entropy	negative entropy
test loss	$\rho = 0.035$	$\rho = 0.011$	$\rho = -0.022$	$\rho = -0.038$
train loss	$\rho = -0.031$	$\rho = -0.119$	$\rho = -0.020$	$\rho = -0.135$
attribution loss	$\rho = -0.073$	$\rho = -0.161$	$\rho = -0.009$	$\rho = -0.132$
test accuracy	$\rho = 0.043$	$\rho = 0.197$	$\rho = -0.031$	$\rho = 0.182$
train accuracy	$\rho = 0.026$	$\rho = 0.116$	$\rho = 0.019$	$\rho = 0.136$
test attribution accuracy	$\rho = 0.099$	$\rho = 0.232$	$\rho = -0.018$	$\rho = 0.172$

Table 9: Spearman's rank coefficient ρ of the Grad-CAM summary and normalized performance measures on CIFAR10 of the deeper model.