A Robust Unsupervised Ensemble of Feature-Based Explanations using Restricted Boltzmann Machines

Vadim Borisov[*]
University of Tübingen

Johannes Meier[†]
University of Tübingen

Johan van den Heuvel[†]
University of Tübingen

Hamed Jalali
University of Tübingen

Gjergji Kasneci
University of Tübingen

Abstract

Understanding the results of deep neural networks is an essential step towards wider acceptance of deep learning algorithms. Many approaches address the issue of interpreting artificial neural networks, but often provide divergent explanations. Moreover, different hyperparameters of an explanatory method can lead to conflicting interpretations. In this paper, we propose a technique for aggregating the feature attributions of different explanatory algorithms using Restricted Boltzmann Machines (RBMs) to achieve a more reliable and robust interpretation of deep neural networks. Several challenging experiments on real-world datasets show that the proposed RBM method outperforms popular feature attribution methods and basic ensemble techniques.

1 Introduction

As the applications of deep neural networks (DNNs) continue to grow, the black-box nature of DNNs creates potential trust issues [1]. Moreover, numerous life-critical (such as medical, automotive, or financial) applications utilize DNNs for various estimation tasks. In such applications, and especially for the long-term acceptance of artificial intelligence (AI) solutions, a deeper understanding and trust in the produced results is crucial. Furthermore, feature attribution methods are important tools for deep model debugging and diagnosis [2].

Explaining how the input influences the output for a given DNN is one form to interpret the black-box nature of the DNN and bring trust to a system. These so-called feature-based explanation methods received a lot of attention in recent years [1, 3–5]. They can be grouped into three broad categories, (1) approaches based on gradient information [6, 7], (2) perturbation-based approaches [8, 9], and (3) attribution-based approaches [3, 10]. Interestingly, different feature-based explanation approaches regularly produce mixed views on the main attributes (areas of an image or variables), and in the absence of the ground truth, it is still a challenge to verify which explanation method is the most trustworthy. Moreover, in the AI community, there are no yet accepted quality measures for feature-based explanations. *All these difficulties resulted in a large number of different explanation methods and in a lack of consensus on which techniques are most reliable.*

Within the machine learning (ML) community, there is much work on the combination of methods that do not always agree with each other, i.e. *ensemble learning* [11, 12]. Normally ensemble models outperform the non-ensemble models and turn out to be more robust to outliers. The main idea is that if multiple methods make mistakes in different areas, combining them in an intelligent way improves

[*] Corresponding author: `vadim.borisov@uni-tuebingen.de`

[†] Equal contribution

1st Workshop on eXplainable AI approaches for debugging and diagnosis (XAI4Debugging@NeurIPS2021).

performance and reduces the effect of outliers as compared to the single method. Moreover, from statistical learning theory and practical applications, it is well understood that ensemble learning is the path of choice towards a more robust machine learning system [13], even in unsupervised learning scenarios where the target is not available [14, 15].

In this work, utilizing ideas from [15, 11] and [16, 17], we introduce a novel approach for the unsupervised ensemble learning of reliable and robust feature-based explanations for deep neural networks. To this end, we propose using a model based on Restricted Boltzmann Machines (RBMs), which achieves this goal by aggregating the results (saliency maps) of different feature-based explanation methods in a principled probabilistic fashion. Also, it has been shown that an RBM can be used in the truth discovery setting [18, 19], which is analogous to our task of finding a reliable feature importance map from different importance maps.

The main contributions of this work are:

- We introduce a novel method for a robust and reliable feature-based explanation using ensemble learning.
- We empirically and visually show the superior performance of the proposed method in comparison to state-of-the-art feature attribution baselines.
- We open-source our code and make it publicly available, as an RBM ensemble framework: (https://github.com/JohanvandenHeuvel/AggregationOfLocalExplanations), Besides, we also developed a single Python package with various evaluation metrics for feature attribution methods metrics: https://github.com/meier-johannes94/ExplainableAIImageMeasures

The paper is organized as follows: In Section 2 we discuss the related work and provide essential background information. Section 3 presents the proposed ensemble method for local feature-based explanations using an RBM. In Section 4, we present the results of various experiments. Section 5 discusses limitations of our work and ways to address them in the future. Section 6 concludes our work with a short summary.

2 Background and Related Work

This part of the manuscript provides the needed background and discusses related approaches. First, we present the basic notation used in this work and proceed by presenting two ensemble techniques for aggregating feature importance maps.

2.1 Feature Attribution Function

Formally, a feature attribution function can be seen as $\phi(f, \mathbf{x}, c_x)$, where f is a black box model and $\mathbf{x}$ is an input data point from a corresponding class c_x. The output of ϕ is an explanation vector or matrix $\mathbf{e}_{f(\mathbf{x})}$, where each element of $\mathbf{e}_{f(\mathbf{x})}$ is an importance score for the corresponding feature value in $\mathbf{x}$. A large positive or negative value in $\mathbf{e}_{f(\mathbf{x})}$ indicates that the corresponding feature (pixel) has a large influence on the outcome of the black-box model f.

Assumption 2.1 In the following, we assume that a true feature attribution $\bar{\mathbf{e}}_f(\mathbf{x})$ for a given model f and input $\mathbf{x}$ exists and can be constructed by adequately aggregating available attributions $\mathbf{e}_{f(\mathbf{x}),i}, i \in \{1, ..., N\}$, where N is the number of baseline explanations (from N baseline methods).

For better readability and simplicity, from here we omit the index $f(\mathbf{x})$.

The goal of any explanation method ϕ is to obtain an attribution $\mathbf{e}$ that is as close as possible to $\bar{\mathbf{e}}$. Note that our method naturally generalizes to probabilistic local explanation methods [20]. Given the before-mentioned assumption, we can say that there is a joint probability distribution of the pair $(\mathbf{e}, \bar{\mathbf{e}})$ parametrized by θ.

$$p_\theta(\mathbf{e}, \bar{\mathbf{e}}) = p_\theta(\bar{\mathbf{e}}) p_\theta(\mathbf{e}|\bar{\mathbf{e}}).$$

The joint distribution $p_\theta(\mathbf{e}, \bar{\mathbf{e}})$ is not known, and neither are the marginals $p_\theta(\mathbf{e})$, $p_\theta(\bar{\mathbf{e}})$.

For the following theoretical results we require that the explanation methods give independent explanations when conditioned on the true explanation. However, as with Naive Bayes methods, for practical purposes, this assumption can be violated without negatively impacting the aggregation quality [15]. Also note that we do assume some consistency between the explanations, following the assumption that feature attributions reflect the underlying (but unknown) importance distributions of the feature values [21].

Assuming conditional independence between the provided baseline explanations given the (unknown) true explanation, we have

$$p_\theta(\mathbf{e}|\bar{\mathbf{e}}) = \prod_{n=1}^{N} p_\theta(\mathbf{e}_n|\bar{\mathbf{e}}),$$

where $\mathbf{e}_n$ is a baseline explanation in the ensemble involving N different baseline explanations.

2.2 Ensemble Learning

As we state in the introduction, ensemble learning is a well-studied approach for improving the performance of an ML system. One of the most basic ensemble methods employs the mean of results of base learners [13], where a *base learner* is a single algorithm from the ensemble.

$$\mathbf{e}_{mean} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{e}_n. \tag{1}$$

A significant drawback of the *mean ensemble approach* is that it still is sensitive to outliers or noisy estimations. Furthermore, data scaling may strongly influence the aggregation. In Section 4, these weaknesses of the *mean ensemble approach* are also seen in the experimental evaluation.

To mitigate these weaknesses, the authors of [22] propose to take the local uncertainty into account. To this end, they divide the mean by the local variance plus a constant ϵ for stability reasons, which results in the *variance ensemble approach*:

$$\mathbf{e}_{var} = \frac{1}{N}\sum_{n=1}^{N} \frac{\mathbf{e}_n}{\sigma_*(\mathbf{e}_{i\in\{1,\ldots,N\}}) + \epsilon},$$

where $\sigma_*(\mathbf{e}_{i\in\{1,\ldots,N\}})$ is the point-wise standard deviation over all the available explanations $\mathbf{e}_i, i \in \{1, ..., N\}$. This method assigns less relevance to explanations that have high disagreement with the remaining explanations.

Also, the authors of [23] proposed a novel method to aggregate Shapley values through an explanation function that minimizes sensitivity.

3 Ensemble Learning using Restricted Boltzmann Machines

In this section, we present an unsupervised aggregation of feature attribution maps using a Restricted Boltzmann Machine (RBM). Similar aggregation techniques have been proposed in other contexts, e.g., in [15, 18].

3.1 The Restricted Boltzmann Machine

An RBM is an undirected bipartite graph that can be parametrized by a neural network. It is a variant of the Boltzmann Machine, with the additional property that there are no connections within both the group of visible nodes or the group of hidden nodes. The advantage of this property is that nodes in one group are conditionally independent of each other, given that we know the state of the nodes in the other group. One of the main characteristics of an RBM is that it can learn a probability distribution over its set of inputs. A graphical representation of an example RBM is shown in Figure 1.

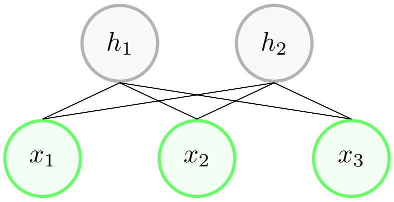

Figure 1: An RBM with three visible and two hidden units. In our work, we use an RBM with a single hidden node.

The formal definition of an RBM is as follows. There is a set X of n visible binary random variables and a set H of m hidden binary random variables. The RBM has parameters $\lambda = (\mathbf{W}, \mathbf{a}, \mathbf{b})$. $\mathbf{W}$ is the weight matrix of the connections between the nodes, $\mathbf{a}$ is the bias of the visible layer and $\mathbf{b}$ is the bias of the hidden layer. Each possible state of the RBM, i.e. the particular values of (X, H), is associated with the following energy function (in matrix notation):

$$E_\lambda(\mathbf{x}, \mathbf{h}) = -(\mathbf{a}^T\mathbf{x} + \mathbf{b}^T\mathbf{h} + \mathbf{x}^T\mathbf{W}\mathbf{h}),$$

which then can also be used to define the joint probability distribution for the visible and hidden vectors is defined in terms of the energy function:

$$P_\lambda(\mathbf{x}, \mathbf{h}) = \frac{1}{Z}e^{-E(\mathbf{x},\mathbf{h})},$$

where Z is the sum over $e^{-E(\mathbf{x},\mathbf{h})}$ for all possible configurations $\mathbf{x}, \mathbf{h}$, which can be seen as a normalization constant to ensure that all probabilities sum to 1, also known as the partition function.

The optimization objective of the RBM is to maximize the expected log probability of a training sample $\mathbf{x}$:

$$\begin{aligned} &\operatorname*{argmax}_\lambda \mathbb{E}[\log P_\lambda(X = \mathbf{x})] = \\ &\operatorname*{argmax}_\lambda \mathbb{E}[\log \sum_{\mathbf{h}} P_\lambda(X = \mathbf{x}, H = \mathbf{h})]. \end{aligned} \tag{2}$$

To train an RBM, a gradient-based optimization can be applied using the contrastive divergence algorithm [24, 25].

3.2 Aggregation of Local Explanations using an RBM

Given an RBM with N visible nodes and one hidden node, with input $\mathbf{x}$, where N is the number of baseline explanations in our ensemble, it can be shown that the true posterior probability of y can be efficiently estimated (Lemma 4.1, Lemma 4.2 from [15]). Furthermore, given the previously discussed mild assumptions on the input data (which are in line with those in [15]), the maximum likelihood estimate $\bar{\lambda}_{MLE}$ for the parameters of the RBM, the RBM posterior probability $P_{\bar{\lambda}_{MLE}}(H = 1|X = \mathbf{x})$ converges to true posterior $P_\theta(Y = 1|X = \mathbf{x})$.

Hence, we are able to apply the RBM to the *unsupervised aggregation* of N available feature-based explanations. We assume a joint distribution $p_\theta(\mathbf{e}, \bar{\mathbf{e}})$, and that the $\mathbf{e}_i$'s are conditionally independent from each other given $\bar{\mathbf{e}}$. By fitting the RBM we learn the parameters θ and thus obtain the relationship between our known explanations $\mathbf{e}_i$ and the true explanation $\bar{\mathbf{e}}$. The ensemble pipeline of the proposed method is depicted in Fig. 2.

In order to preserve the spatial information for visual data using the RBM-based ensemble, we do a pixel-wise aggregation. Therefore, for each pixel we train a Bernoulli RBM with a single hidden unit.

A known limitation of an RBM is the so-called *flipping issue* [15, 18, 19], which arises from the RBM parametrization symmetry. That is, the weights of the RBM can be flipped symmetrically without changing the behavior of the RBM. In order to avoid this unwanted effect, we propose two approaches: flip detection and metric optimization. The *flip detection* algorithm extends the idea from Remark 4.3 in [15], by comparing the top 5 % of most important and 5 % of less important pixels to the mean baseline. The algorithm inverts the current important scores if there is a strong disagreement between the proposed approach and the mean baseline. The *metric optimization* method utilizes the chosen metric to overcome the flipping issue. It compares two versions of the RBM ensemble results and selects the one with a better performance according to the selected metric.

4 Experiments

To demonstrate the effectiveness of the proposed ensemble algorithm we conduct various visual and quantitative experiments. First, we present the visual inspection results on the MNIST [27] and ImageNet [26] datasets in two settings, with and without noisy explanation maps in our ensemble.

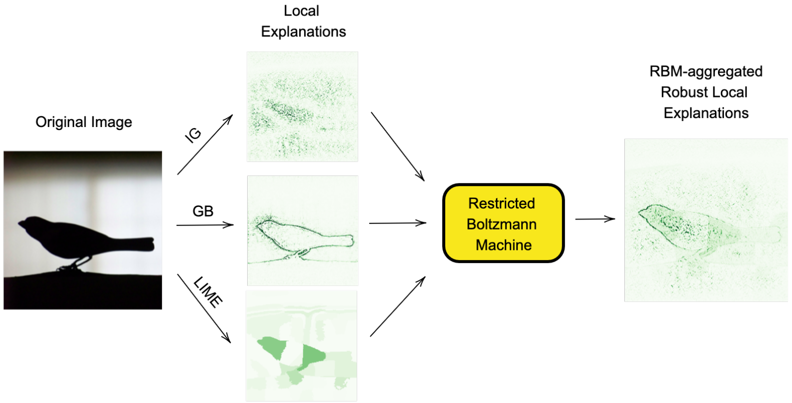
Original Image
Local Explanations
IG
GB
LIME
Restricted Boltzmann Machine
RBM-aggregated Robust Local Explanations

Figure 2: An overview on the ensemble of feature attribution maps from three different local explanation algorithms using an RBM for an image from the ImageNet dataset [26].

Despite a growing body of research focusing on explainable ML, the fair quantitative comparison of local explanation (or saliency-based) algorithms is still an open question, since the existing methods mostly utilize the pixel perturbation strategy (e.g., removing the most or least important pixels and reporting the change in recognition quality) [28, 29]. Also, such evaluations have a significant drawback, replacing image pixels with black or "mean" or any other pixel values may lead to artifacts affecting the data distribution [29, 30]. Nevertheless, since pixel perturbation analyses are employed in many related works, for our quantitative analysis we select the following approaches: the pixel perturbation for insertion (IAUC) and deletion (DAUC). Furthermore, we utilize the iterative removal of features (IROF) analysis [31]. We explain each evaluation method in detail in the corresponding subsections.

In our last experiment, we demonstrate that our ensemble approach can be also used within a singe feature attribution framework to achieve more robust and stable explanations. Since, it has been shown that hyperparameters choice can significantly affect the saliency maps [32].

4.1 Visual Inspection for Image Data

In our first experiment, a visual evaluation on images from ImageNet [26] and MNIST [27] is performed for several baseline and ensemble methods. We provide benchmark outcomes for two settings, with and without fifteen noisy baseline explanations in an ensemble. The results are depicted in Table 1.

Without artificial noise in the ensemble. We select four samples from ImageNet dataset [26] and five baseline explanation methods for the ensemble models: LIME [8], Guided Backpropagation (GB) [33], Integrated Gradients (IG) [34], Gradient SHAP (GS) [9], and SmoothGrad (SG) [35]. We compare the proposed RBM ensemble strategy to simple mean and variance ensembles [22]. The results in Table 1 show that our approach produces sharp and visually appealing saliency maps in comparison to other ensemble baselines. In comparison to the baseline explanation methods, the proposed ensemble technique seems to produce more reliable and robust results by highlighting commonalities among the baseline methods and by mitigating the noise coming from the single baseline methods.

With artificial noise in the ensemble. We challenge the discussed approaches by adding fifteen baselines with random noise sampled from the standard normal distribution $\mathbf{e}_{rand} \sim \mathcal{N}(0, 1)$ to the ensemble. The results in Table 1 reveal that the proposed RBM-based aggregation method mitigates noise and hence results in more robust saliency maps in comparison to the other ensemble baselines.

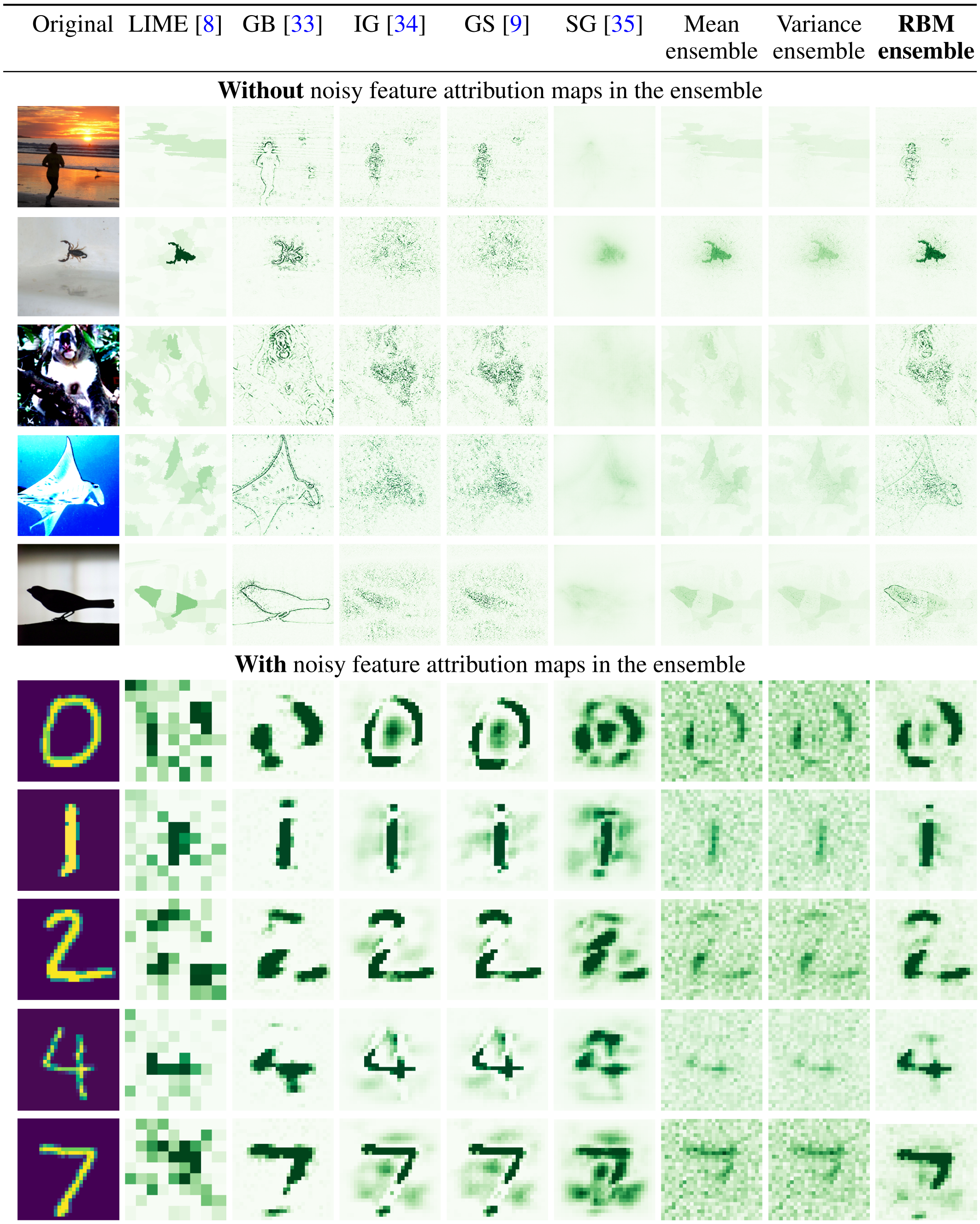
Original
LIME [8]
GB [33]
IG [34]
GS [9]
SG [35]
Mean ensemble
Variance ensemble
RBM ensemble
Without noisy feature attribution maps in the ensemble
With noisy feature attribution maps in the ensemble

Table 1: A visual comparison between baseline methods and ensemble methods on ImageNet [26] and MNIST [27] datasets.

Method	Insertion (IAUC)	Deletion (DAUC)	IROF [31]
Gradient SHAP [9]	0.61 ± 0.42	0.22 ± 0.29	0.73 ± 0.24
DeepLIFT [3]	0.62 ± 0.42	0.23 ± 0.30	0.73 ± 0.23
LIME [8]	**0.80** ± **0.31**	0.23 ± 0.23	**0.76** ± **0.22**
Saliency map [38]	0.50 ± 0.35	0.37 ± 0.32	0.65 ± 0.25
SmoothGrad [35]	0.60 ± 0.26	0.38 ± 0.29	0.63 ± 0.26
Integrated Gradients [34]	0.66 ± 0.42	**0.19** ± **0.27**	0.75 ± 0.23
Guided Backpropagation [33]	0.54 ± 0.38	0.49 ± 0.36	0.65 ± 0.25
Original Image	0.52 ± 0.32	0.53 ± 0.34	0.47 ± 0.30
Mean Ensemble	**0.79** ± **0.33**	0.25 ± 0.28	0.70 ± 0.26
Variance Ensemble [22]	0.62 ± 0.36	0.39 ± 0.31	0.71 ± 0.26
RBM ensemble with the flip detection	0.76 ± 0.38	0.19 ± 0.26	0.76 ± 0.22
RBM ensemble with the metric optimization	0.77 ± 0.37	**0.18** ± **0.24**	**0.76** ± **0.22**

Table 2: A quantitative comparison between single and ensemble methods for the pixel perturbation: IAUC (higher is better), DAUC (lower is better), and IROF (higher is better) experiments on 10,000 samples from the CIFAR10 validation dataset [36].

4.2 Pixel Perturbation Experiment

In the first quantitative experiments, we compare multiple baseline models and ensemble methods on the CIFAR10 dataset [36] by removing the most important pixels (according to a scoring function) and reporting the area under a curve score (DAUC). In addition, we also follow the approach of inserting the most important pixels into an empty image and again report the area under a curve (IAUC). Thus, an ideal feature scoring function has a large IAUC and low DAUC. These benchmark methods well accepted by the research community [37]. For this experiment, we select the following algorithms as baseline explanation methods: Gradient SHAP [9], DeepLIFT [3], LIME [8], Saliency maps [38], SmoothGrad [35], Integrated Gradients [34], Guided Backpropagation [33]. As suggested in [11], we add the original image as a baseline to the ensemble. However, according to our experiments, adding th original image to the ensemble does improve the overall ensemble performance. We report all scores for the baseline approaches and the ensemble methods in Table 2.

4.3 IROF Experiment

In [22] the authors propose the IROF measure as an extension to the work [39]. The main idea of the IROF benchmark is as follows: the image is divided into superpixels using the SLIC algorithm [40]. Superpixels are regional blocks of pixels within an image where the contained pixels share a high similarity measure among each other. The relevancy for each superpixel is calculated by averaging over the attribution scores over all contained pixels (inside the superpixel). After, the superpixels are sorted descending by their relevancy. The entire superpixels are gradually replaced by a baseline and sent through the network again to measure the new recognition quality for the modified image wrt. to the target label. For more accurate attribution methods, the recognition quality decreases faster, and thus the area under the curve is lower. The IROF score is defined as the area over the curve (AOC): $AOC = 1 - AUC$. Higher values, therefore, indicate a better attribution. We use the same baseline methods as in the pixel perturbation experiment (Sec. 4.2). The results are listed in Table 2.

4.4 An RBM Ensemble Within a Singe Explanation Framework

In this experiment, we demonstrate that even for multiple baseline explanations of the same explanation method, the unsupervised ensemble with an RBM can lead to an improvement. To this end, we select the LIME [8] method with a different hyperparameter - the number of superpixels in the image. For the baselines (LIME-0, LIME-1, and LIME-2), we used 10, 100, and 1000 superpixels per image, respectively. The results can be seen in Fig. 4. The main idea is that each lime method has a different granularity level, thus highlighting distinct detail levels, and the proposed method's aggregation may help improve the reliability and robustness of feature attributions.

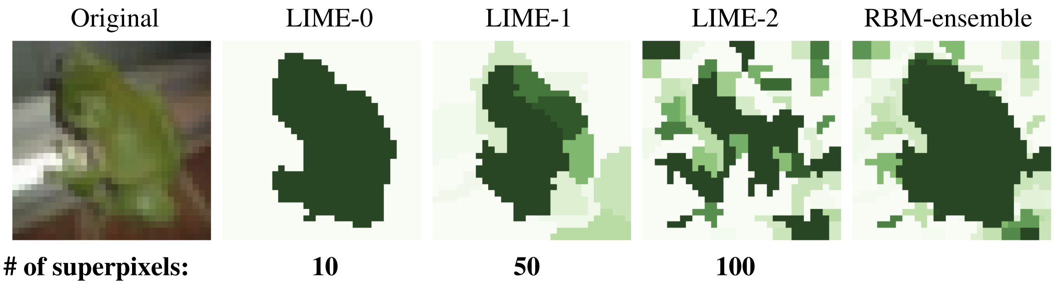

Figure 3: Three local feature attribution maps for a single data sample from CIFAR10 dataset [36] using the LIME algorithm [8] with different number of superpixels and the proposed RBM ensemble of the selected feature attribution maps.

4.5 Reproducibility

For reproducibility reasons, we describe the data preprocessing step used for all the experiments and provide information about packages used in this work. Also, the code for every experiment is publicly available online (see the links provided in Section 1).

To achieve a fair comparison, the image data from all datasets was preprocessed in the same way for each baseline. We performed a per saliency map normalization before the aggregation. In every experiment, we used the ResNet18 neural network architecture [41], except for the experiment on the MNIST dataset where we utilized a simple five layers convolutional neural network. We use a pre-trained model for ImageNet dataset [26] from torchvision library [42].

For the experiments we used the Bernoulli RBM implementation from the Scikit-Learn library [43] with following hyperparameters for each experiment: for the MNIST dataset we set the batch size to 5, the learning rate to 0.01, and the number of iterations to 100. For CIFAR10 and ImageNet datasets we use the following hyperparameters: a batch size of 35, a learning rate of 0.001, and a number of iterations is 250. The rest of hyperparameters are default to the scikit-learn package. For all baseline explanation techniques we use the publicly available open-source implementations from the captum library [44] with their default hyperparameters.

5 Discussion and Future Work

The results of multiple experiments with the proposed RBM ensemble show its competitive performance compared to base explanation techniques and other ensemble approaches. We hypothesize moderate performance of the RBM ensemble on the insertion (IAUC) benchmark is connected to our data preparation step since we filter the negative values for every saliency map in the ensemble.

The computational complexity of an ensemble method primarily depends on the base learner. In our case, the base explanation techniques are relatively fast, especially on specialized hardware (GPU or TPU), where an RBM has low computational complexity.

The gradient-based methods frequently produce noisy explanations. We empirically demonstrated that our approach reduces the noise in the final ensemble (Tab. 1). Therefore, we believe that the RBM aggregation of multiple saliency maps from gradient-based feature attributions is a powerful tool for improving the overall reliability of local explanations.

As part of our future work, we aim to evaluate our aggregation approach on larger datasets. Furthermore, methods for selecting a few quite reliable base explanations for aggregation might lead to efficient explanations ensembles for larger datasets.

Finally we expect that the proposed approach can be easily adapted to handle local explanations over structured tabular data, where the explanation of deep neural networks is an essential task for many crucial applications such as healthcare and finance [45].

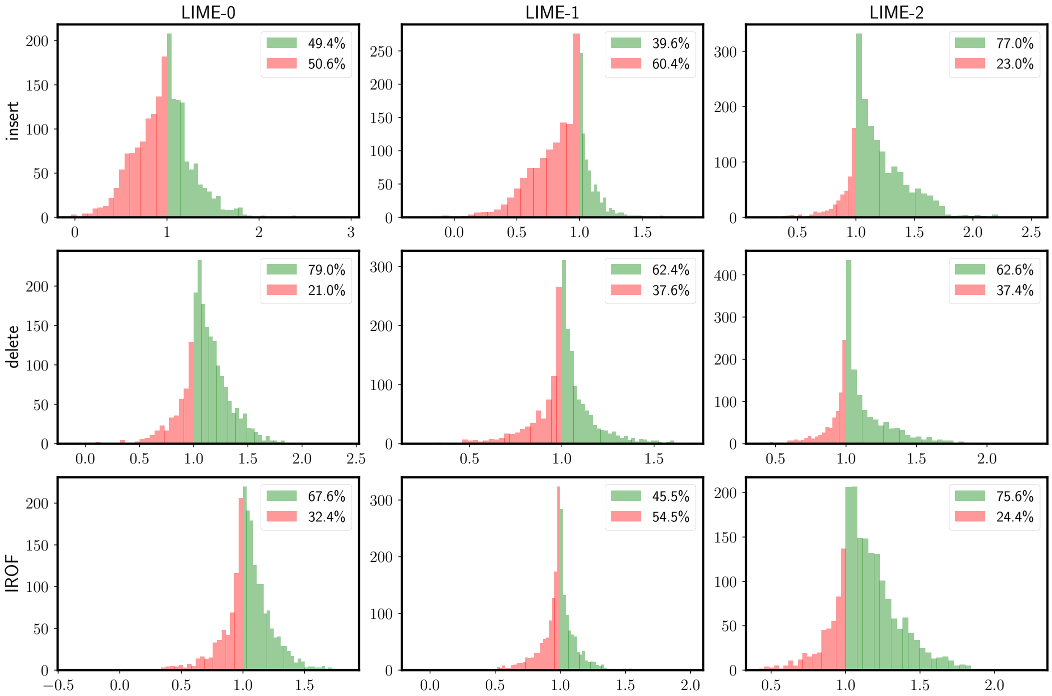

Figure 4: Distributions of differences where the proposed RBM ensemble shows better (green) and inferior (red) results in comparison to a baseline explanation technique according to insertion, deletion, and IROF metrics (score 1 means them being equal). The ensemble consists of the feature attributions from the same algorithm - LIME, but different hyperparameters. We randomly sampled 2000 images from CIFAR10 [36] for this experiment.

6 Conclusion

In this work, we presented a novel approach to unsupervised aggregation of feature-based explanations using Restricted Boltzmann Machines with the aim of reliably interpreting the influence of inputs on the output of deep neural networks. In addition to explanatory reasons, the latter is also essential for debugging and diagnostic purposes and serves the long-term acceptance of deep learning in real-world applications.

Using the proposed approach, we demonstrated through visual and quantitative experiments its ability to obtain more robust and reliable explanations than other existing ensemble methods. In a setting with noisy attribution maps in an ensemble, the proposed approach successfully selects only the valuable information, mitigating noise. Moreover, our work illuminates and mitigates the problem of possible contradictory results that may be obtained by different explanation and evaluation methods. Finally, we note that our approach can also be used within a single interpretability framework to reduce the sensitivity of a feature-based explanatory approach to its hyperparameters.

References

[1] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A survey on neural network interpretability. *arXiv preprint arXiv:2012.14261*, 2020.

[2] Mattia Setzu, Riccardo Guidotti, Anna Monreale, Franco Turini, Dino Pedreschi, and Fosca Giannotti. Glocalx-from local to global explanations of black box ai models. *Artificial Intelligence*, 294:103457, 2021.

[3] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pages 3145–3153. PMLR, 2017.

[4] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE, 2018.

[5] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*, 2018.

[6] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

[7] Gjergji Kasneci and Thomas Gottron. Licon: A linear weighting scheme for the contribution ofinput variables in deep artificial neural networks. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 45–54, 2016.

[8] Marco Tulio 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*, pages 1135–1144, 2016.

[9] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pages 4768–4777, 2017.

[10] Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. Layer-wise relevance propagation: an overview. *Explainable AI: interpreting, explaining and visualizing deep learning*, pages 193–209, 2019.

[11] Chun-Xia Zhang, Jiang-She Zhang, Nan-Nan Ji, and Gao Guo. Learning ensemble classifiers via restricted boltzmann machines. *Pattern Recognition Letters*, 36:161–170, 2014.

[12] Zhi-Hua Zhou. *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC, 2019.

[13] Ludmila I Kuncheva and Christopher J Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

[14] Ariel Jaffe, Ethan Fetaya, Boaz Nadler, Tingting Jiang, and Yuval Kluger. Unsupervised ensemble learning with dependent classifiers. In *Artificial Intelligence and Statistics*, pages 351–360. PMLR, 2016.

[15] Uri Shaham, Xiuyuan Cheng, Omer Dror, Ariel Jaffe, Boaz Nadler, Joseph Chang, and Yuval Kluger. A deep learning approach to unsupervised ensemble learning. In *International conference on machine learning*, pages 30–39. PMLR, 2016.

[16] Gjergji Kasneci, Jurgen Van Gael, David Stern, and Thore Graepel. Cobayes: bayesian knowledge corroboration with assessors of unknown areas of expertise. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 465–474, 2011.

[17] Gjergji Kasneci, Jurgen Van Gael, Ralf Herbrich, and Thore Graepel. Bayesian knowledge corroboration with logical rules and user feedback. In *Joint european conference on machine learning and knowledge discovery in databases*, pages 1–18. Springer, 2010.

[18] Klaus Broelemann, Thomas Gottron, and Gjergji Kasneci. Restricted boltzmann machines for robust and fast latent truth discovery. *arXiv preprint arXiv:1801.00283*, 2017.

[19] Klaus Broelemann and Gjergji Kasneci. A gradient-based split criterion for highly accurate and transparent model trees. *CoRR*, abs/1809.09703, 2018.

[20] Xingyu Zhao, Wei Huang, Xiaowei Huang, Valentin Robu, and David Flynn. Baylime: Bayesian local interpretable model-agnostic explanations, 2021.

[21] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. *Distill*, 5(1):e22, 2020.

[22] Laura Rieger and Lars Kai Hansen. Aggregating explanation methods for stable and robust explainability, 2020.

[23] Umang Bhatt, Adrian Weller, and José MF Moura. Evaluating and aggregating feature-based model explanations. *arXiv preprint arXiv:2005.00631*, 2020.

[24] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[25] Yoshua Bengio. *Learning deep architectures for AI*. Now Publishers Inc, 2009.

[26] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[27] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[28] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks, 2018.

[29] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. Evaluating feature importance estimates. *CoRR*, abs/1806.10758, 2018.

[30] Johannes Haug, Stefan Zürn, Peter El-Jiz, and Gjergji Kasneci. On baselines for local feature attributions. *CoRR*, abs/2101.00905, 2021.

[31] Laura Rieger and Lars Kai Hansen. Irof: a low resource evaluation metric for explanation methods. *arXiv preprint arXiv:2003.08747*, 2020.

[32] Naman Bansal, Chirag Agarwal, and Anh Nguyen. Sam: The sensitivity of attribution methods to hyperparameters. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 8673–8683, 2020.

[33] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net, 2015.

[34] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. *CoRR*, abs/1703.01365, 2017.

[35] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[36] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *citeseerx*, 2009.

[37] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018.

[38] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.

[39] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Bach, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *CoRR*, abs/1509.06321, 2015.

[40] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels. Technical Report 149300, EPFL, June 2010.

[41] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[42] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[44] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch, 2020.

[45] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. *arXiv preprint arXiv:2110.01889*, 2021.

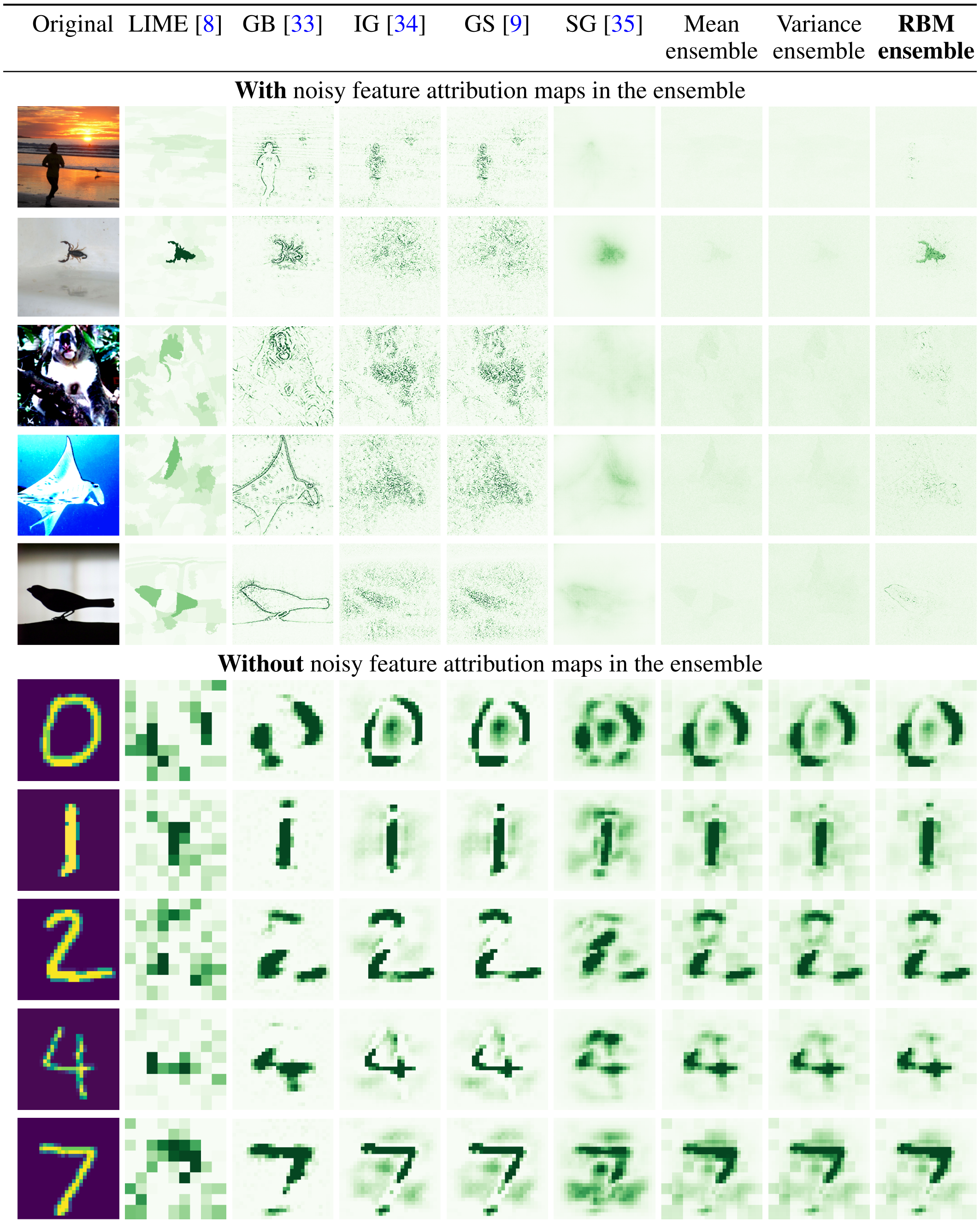
Original
LIME [8]
GB [33]
IG [34]
GS [9]
SG [35]
Mean ensemble
Variance ensemble
RBM ensemble
With noisy feature attribution maps in the ensemble
Without noisy feature attribution maps in the ensemble

Table 3: A visual comparison between base learners and ensemble methods on ImageNet [26] and MNIST [27] datasets.

A Additional Experiments

Table 3 presents extended experimental results for the compression with or without noisy feature attribution maps in the ensemble.