Correcting Classification: A Bayesian Framework Using Explanation Feedback to Improve Classification Abilities

Yanzhe Bekkemoen and Helge Langseth

NorwAI Norwegian Research Center for AI Innovation

NTNU

Introduction

Neural networks have shown high predictive performance, however, with shortcomings:

- Firstly, the reasons behind the classifications are not fully understood. Several explanation methods have been developed, but they do not provide mechanisms for users to interact with the explanations. Explanations are "social", meaning they are a transfer of knowledge through interactions. Nonetheless, current explanation methods contribute only to one-way communication.
- Secondly, neural networks tend to be overconfident, providing unreasonable uncertainty estimates on out-of-distribution observations.

We overcome these difficulties (incorrect explanations and model overconfidence) by training a Bayesian convolutional neural network that uses explanation feedback.

Methodology

Explanation Feedback

- A model presents explanations of training sample classifications to an annotator after training. Based on the provided information, the annotator can accept or reject the explanations by giving feedback. Our proposed method utilizes this feedback for fine-tuning to correct the model such that the explanations and classifications improve.

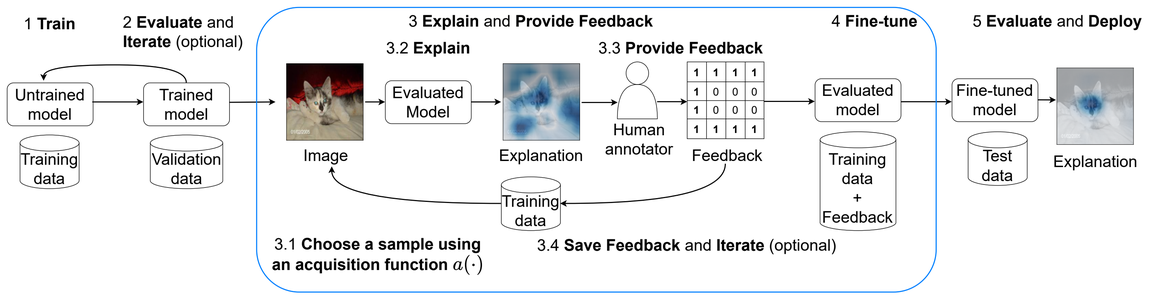


Figure 1: A "standard" ML pipeline with steps for annotating explanations and correcting a model. During Step 3, a model explains training sample classifications to a user who gives feedback on those explanations. A feedback $\boldsymbol{F}^{(i)}$ for a sample i is a matrix of the same width and height as the image. If a feature k, j is irrelevant $\boldsymbol{F}^{(i)}_{k,j} = 1$, otherwise $\boldsymbol{F}^{(i)}_{k,j} = 0$. In Step 4, a model is fine-tuned with training data and feedback. The goal is to improve the reasons behind the classifications (explanation in Step 3 vs. Step 5) and predictive performance.

Bayesian Neural Network

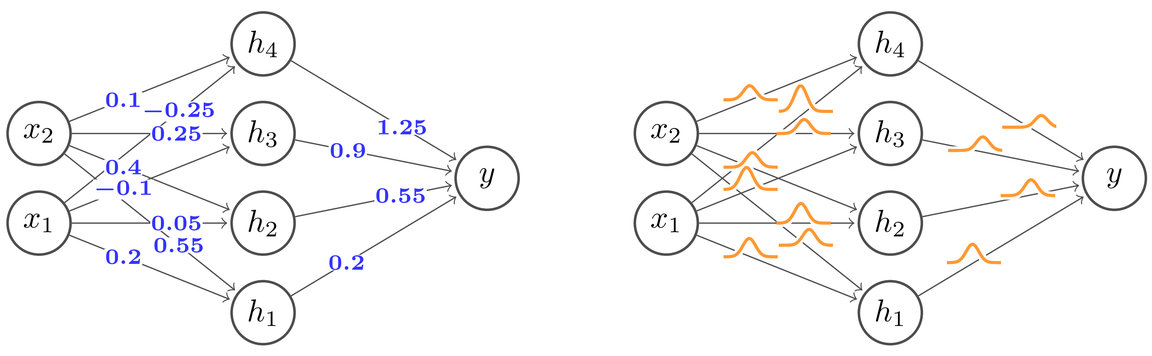


Figure 2: The left side depicts a regular feed-forward neural network with weights represented as scalar values. The right side shows a Bayesian neural network with weights represented as probability distributions.

- We use variational inference [2, 3] to train a Bayesian neural network with explanation feedback as additional evidence by using the following objective function:

$$\mathcal{L}(\boldsymbol{\theta}) = \underbrace{D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\boldsymbol{w}) \| P(\boldsymbol{w}))}_{\text{Complexity}} - \underbrace{\mathbb{E}_{q_{\boldsymbol{\theta}}(\boldsymbol{w})}[\log P(\mathcal{D}|\boldsymbol{w})]}_{\text{Likelihood}} \tag{1}$$

- To compute the likelihood of the explanation feedback, we use the local reparameterization trick (LRT) [3]. LRT lets us compute the activation distribution of the last convolution layer (forward direction) that captures high-level semantics and detailed spatial information.

Results

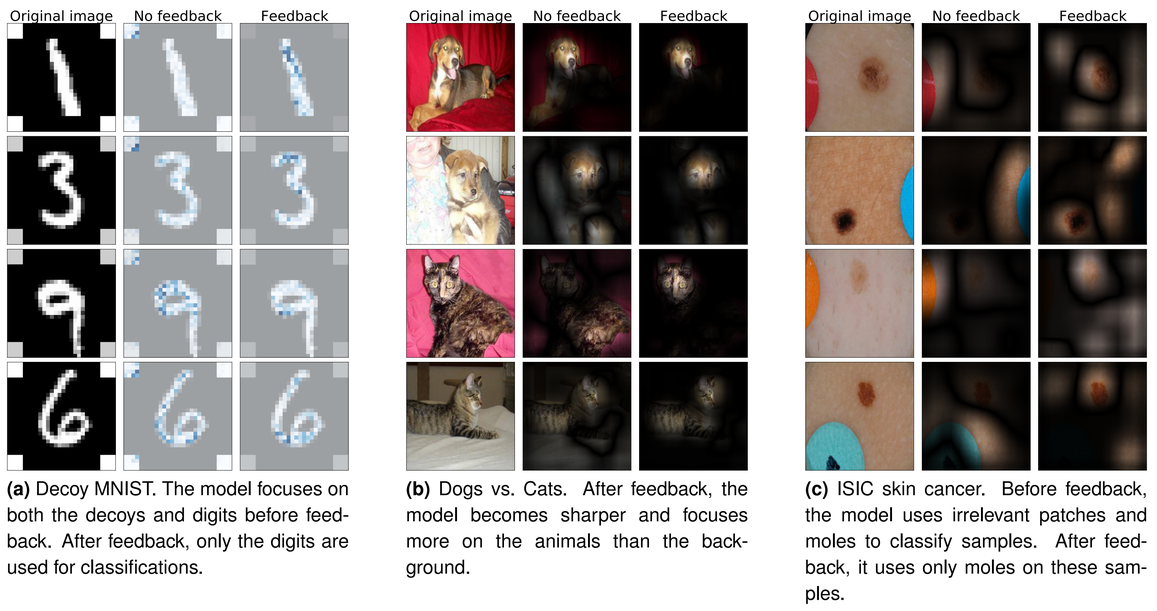
- We use existing convolutional neural network architectures (LeNet and AlexNet) to demonstrate the method's effectiveness on one toy dataset (decoy MNIST) and two real-world datasets (Dogs vs. Cats and ISIC skin cancer).
- Decoy MNIST is a modified version of MNIST where every sample in the dataset has a 4×4 square in each corner (see Figure 3a). In the training data, the decoys' colors correspond with the label of the digit y, $(255 - 25y)$ and in the test data, the colors are randomly drawn.
- Dogs vs. Cats and ISIC skin cancer (benign or malignant) are binary classification datasets. Of those benign samples in ISIC skin cancer, half of the samples have colorful patches.
- The results indicate that few annotated explanations and fine-tuning epochs are required to improve explanations justifying the classifications and the predictive performance.

Dataset	Precision		Recall		F1		Accuracy	
	NF	F	NF	F	NF	F	NF	F
Decoy MNIST	0.725	**0.970**	0.725	**0.970**	0.725	**0.970**	0.725	**0.970**
Dogs vs. Cats	0.918	**0.923**	0.857	**0.870**	0.887	**0.896**	0.886	**0.894**
Skin cancer	0.280	**0.320**	**0.904**	0.798	0.427	**0.457**	**0.815**	0.799
Skin cancer NP	0.289	**0.335**	**0.904**	0.798	0.437	**0.472**	0.702	**0.721**

Table 1: Performance metrics of the model trained with no feedback (NF) and feedback (F). For decoy MNIST, all of the metrics are calculated using micro average. The skin cancer dataset is tested with and without patch data (NP) and accuracy is computed with macro average recall, also known as balanced accuracy.

Dataset	Saliency		DeepLIFT		Grad-CAM		Occlusion	
	NF	F	NF	F	NF	F	NF	F
Decoy MNIST	0.080	**0.031**	0.078	**0.019**	0.063	**0.025**	0.181	**0.050**
Dogs vs. Cats	0.396	**0.384**	**0.405**	0.407	0.441	**0.379**	0.393	**0.380**
Skin cancer	0.154	**0.111**	0.145	**0.124**	0.221	**0.071**	0.250	**0.148**

Table 2: Attributions overlapping with irrelevant features averaged over all samples in the test dataset with annotated explanation. The attribution overlap score is bounded $[0, 1]$ and a **lower** score is better because it implies less attention is focused on irrelevant features. For Occlusion, a sliding window of size 3×3 was used for decoy MNIST and 23×23 for the other two datasets.



(a) Decoy MNIST. The model focuses on both the decoys and digits before feedback. After feedback, only the digits are used for classifications.

(b) Dogs vs. Cats. After feedback, the model becomes sharper and focuses more on the animals than the background.

(c) ISIC skin cancer. Before feedback, the model uses irrelevant patches and moles to classify samples. After feedback, it uses only moles on these samples.

Figure 3: Explanations before and after fine-tuning with feedback visualized on samples from test datasets. DeepLIFT was used for decoy MNIST and Grad-CAM for both Dogs vs. Cats and ISIC skin cancer to visualize explanations.

References

[1] Y. Bekkemoen and H. Langseth. Correcting Classification: A Bayesian Framework Using Explanation Feedback to Improve Classification Abilities. *arXiv preprint arXiv:2105.02653*, 2021.

[2] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight Uncertainty in Neural Network. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622, Lille, France, 07–09 Jul 2015. PMLR.

[3] D. P. Kingma, T. Salimans, and M. Welling. Variational Dropout and the Local Reparameterization Trick. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.