Explainable AI: Combining Introspective & Justification Explanation Systems

Samarth Bhargav (11859032) **Daniel Daza** (11660201) **Gulfaraz Rahman** (11441941)
Christina Winkler (11902256)[1]

Abstract

Explaining a classification decision can decipher why a model does (or does not) work as intended. Even when a model performs well, it is beneficial to understand why it works, so a user gains trust. As an alternative to using bounding boxes to ground visual explanations, we employ a Gradient-weighted Class Activation Mapping (Grad-CAM) to an explanation model, thereby combining both the 'justification' and the 'introspective' aspects of an explainable system. Grad-CAM uses the gradients flowing into the final convolutional layer to produce a coarse localization map indicating the degree which regions contributed to the classification decision. We show the results of this technique on a fine-grained birds classification task. Additionally, we test the robustness of the explanation model by running adversarial attack experiments, as well as generating counter-factual explanations.

1. Introduction

Explaining decisions are a major part of human communication, understanding and learning. Modern neural networks already successfully solve the tasks of localizing an object, predicting its category and describing the object with natural language. However, an AI agent should also be capable of justifying decisions and pointing to evidence explaining their decisions visually as well as textually. The decisions of neural networks are often hidden from the user and it is an important task to provide explanatory text grounded in an image for the user to gain trust in an AI agent. (Hendricks et al., 2016) explores a model which produces sentences that explain why a predicted label is appropriate for a given image, and is an example of a 'justification' explanation system which produces sentences that explain why a certain classification decision was made. In a follow up work (Hendricks et al., 2017) the authors explore a new model which generates similar explanations, but utilizes localized grounding of constituent phrases, which ensures that the generated explanations are more image specific. In contrast to the two methods described above, Grad-CAM (Selvaraju et al., 2016), an 'introspective' method, uses gradient information to produce a coarse localization map which highlights which part of the image led to a class decision.

We propose to employ Grad-CAM instead of the methods described in (Hendricks et al., 2017) to produce localized maps which highlight why a class decision was made. In particular, we examine which regions of the image correspond to image-specific attributes ex. If the model produces an explanation 'This is a *Laysan Albatross* because this bird has a large wingspan, hooked yellow beak, and white belly", corresponding regions of the image (i.e 'hooked yellow beak', 'white belly') should be highlighted. This method combines both 'justification' and 'introspective' methods, and allows a user to examine both the generated explanation and localized image regions.

2. Related Work

Our work mainly builds up on the model presented in (Hendricks et al., 2016) and we combine it with a gradient based approach for visualizing class-discriminative image regions. We also draw from recent work on experiments with adversarial examples presented to our network architecture.

Generating Visual Explanations. The model presented in (Hendricks et al., 2016) focuses on the discriminating properties of the visible object, jointly predicts a class label, and generates an explanation of why the predicted label is appropriate for the image. The model incorporates a loss function based on sampling and reinforcement learning to generate explanation sentences. To constrain object parts to actually be present in the image, (Hendricks et al., 2017) extends the model by utilizing localized grounding of constituent phrases in generated explanations to ensure image relevance. This is done by first generating visual explanations, then chunking them into smaller pieces and subsequently localizing each chunk with a grounding model. In the sequel, we refer to this model as GVE.

[1]Supervisor: Dr. Zeynep Akata.
Samarth Bhargav <samarth.bhargav@student.uva.nl>
Daniel Daza <daniel.dazacruz@student.uva.nl>
Gulfaraz Rahman <gulfarazyasin@gmail.com>
Christina Winkler <christina.winkler@student.uva.nl>.

Project AI 2018, MSc Artificial Intelligence, Univ. of Amsterdam

Localization of class-discriminative regions. In this work, we utilize the Grad-CAM approach as discussed in (Selvaraju et al., 2016) instead of using bounding boxes to ground object specific features in an image. Grad-CAM is a class-discriminative localization technique that can generate visual explanations from any network using convolutional neural networks (CNN) without requiring architectural changes or re-training. Grad-CAM uses the gradient information flowing into the last convolutional layer of the CNN to decipher the importance of each neuron for a decision of interest. This enables highlighting image regions which explain decisions the network can possibly make. Previous works have evaluated the results of introspection via a qualitative evaluation of a few results, custom metrics or human subjects (Hendricks et al., 2017; Springenberg et al., 2014; Vondrick et al., 2013; Zeiler & Fergus, 2014).

Adversarial Attacks. In (Goodfellow et al., 2015) the authors argue that classifiers do not learn the true underlying concepts that determine the correct output label, but rather only provide good performance on naturally occurring data and fail to predict the correct class label when small local changes in the image are made. That is, perceptually the difference after making such a change is almost unrecognizable whereas in the feature space of the network this leads to completely different classes.

3. Generating Visual Explanations

In the GVE model[1], a deep fine-grained classifier takes an image as an input and produces visual features and a class label. It is then the task of an explanation model to use the provided label and features to generate a sentence that describes the image appropriately.

In the classifier model, a CNN is used to extract features from the input image. These features are passed through a Compact Bilinear Pooling (CBP) layer as proposed in (Gao et al., 2016). This is an efficient method to reduce the dimensionality of features from a convolutional layer that can be used in fine-grained classification tasks. The bilinear features are then passed to a fully connected layer that gives the predicted label. The explanation model then uses both the bilinear features and the predicted label as inputs to two stacked LSTMs, which generate the explanation sentence.

Architecturally, this model is based on the LRCN model (Donahue et al., 2015), although it differs on the training procedure. The GVE model is trained with two objectives to minimize: a discriminative and a relevance loss. The discriminative loss allows the model to produce explanations that truly differentiate an instance of a class from another, whereas the relevance loss allows it to generate explanations actually observed in the image, as opposed to generic descriptions of the class of the instance.

[1]We thank Stephan Alaniz for providing the source code.

4. Grad-CAM

The Grad-CAM approach works on the feature maps produced by the last convolutional layer in the classifier network of the GVE model. In particular, we are interested in the gradient of components of an explanation (probabilities of words) with respect to the feature maps, in order to determine what variations in certain regions of the image produce the largest variation on the predicted word. Since the GVE architecture is fully differentiable, it is possible to calculate these gradients. We adapt the Grad-CAM method to the model as follows: let y^c be the log-probability of the word at position c in the sentence, and let A^k_{ij} be the value at the position (i, j) of the feature map produced by the k-th filter of the last convolutional layer. An importance weight is first calculated via global average pooling as

$$\alpha^c_k = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}} \tag{1}$$

The Grad-CAM map is then obtained as

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k \alpha^c_k A^k\right) \tag{2}$$

The obtained map has the size of the feature maps resulting from the last convolutional layer. In order to visualize it on top of the input image, we upsample it using bilinear interpolation so that its resolution matches the image.

5. Adversarial attacks using the Fast Gradient Sign Method

To further bolster confidence in the model, we have to ensure that it is robust to adversarial attacks. We use the Fast Gradient Sign Method proposed by (Donahue et al., 2015) to generate adversarial images. Let the network have parameters $\boldsymbol{\theta}$, the input $\boldsymbol{x}$ and a target y. If the cost of the network is given by $J(\boldsymbol{\theta}, \boldsymbol{x}, y)$, then an adversarial sample $\boldsymbol{x'}$ can be obtained by the following equation:

$$\boldsymbol{x'} = \boldsymbol{x} + \epsilon \,\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$$

where sign operates on each element, ϵ is a parameter set by the user, and the gradient is computed by standard back-propagation. We design two adversarial attacks: The first tries to 'attack' the explanation generated by the GVE model, by setting J to the loss of the GVE model as proposed in (Hendricks et al., 2016). The second experiment tries to attack the image classifier which produces (a) the class label and (b) the bilinear features, by setting J to the cross-entropy loss corresponding to the image classifier's outputs.

6. Experimental Setup

6.1. Dataset

To train the model we used images from the CUB dataset (Wah et al., 2011), which consists of 11.788 images of 200 different classes of birds. Descriptions for these images were taken from a previoulsy developed extension (Reed et al., 2016) that used the Amazon Mechanical Turk service to provide descriptions of the images. Additionally, we resize and crop the images so they have a size of 224×224 while preserving the aspect ratio.

6.2. Model implementation

The GVE architecture requires to train a sentence classifier, an image classifier and an explanation model. All the models were implemented using the PyTorch library (Paszke et al., 2017), with the Adam optimizer, an initial learning rate of 0.001 and a batch size of 128. When training the image classifier, the pre-trained CNN was not fine-tuned.

The sentence classifier was implemented using an LSTM stack, as proposed in (Donahue et al., 2015), and a fully connected layer. For the deep fine-grained image classifier we used a modified CNN, based on the D configuration proposed in (Simonyan & Zisserman, 2014) which had been previously trained on the ImageNet dataset (Deng et al., 2009). We remove the last fully connected layers so that only the feature maps (after a ReLU and max-pooling layer) are passed through the CBP layer. The dimension of the obtained bilinear features was set at 8192. These are then passed through a fully connected layer with 200 outputs corresponding to the number of classes.

For the explanation model the dimension of the word embeddings and the hidden states of the LSTMs was set to 1000.

6.3. Introspection

Given an input image, the GVE model generates an explanation in the form of log-probabilities of words. We used the gradient of these log-probabilities with respect to the feature maps, following equations 1 and 2, using two approaches: by adding the log-probabilities and back-propagating the gradient of the result, and by back-propagating the log-probabilities individually for chunks of attributes in the generated explanation (the chunking procedure is described below). With the first approach we obtain a single map highlighting the information used in the image to obtain the complete explanation, whereas with the second we obtain a highlight map for each attribute in the explanation.

6.4. Evaluation

In order to evaluate the generated explanations, we use METEOR and CIDEr scores, as proposed in (Hendricks et al., 2016). Both metrics attempt to measure how words in two sentences match.

To evaluate the localization maps generated by the introspection method, we provide a qualitative evaluation using some example results. In addition to this, we propose the *gradient-to-box ratio* (GBR) as a quantitative measure based on the bounding boxes provided with the CUB dataset, which enclose the body of the bird. The proposed measure is obtained by calculating the fraction (as a percentage) of the highlights that lies within the bounding box, where the highlights are calculated by by adding the log-probabilities of the explanation and back-propagating the gradient of the result. According to this, if all the highlights were inside the bounding box a GBR of 100% would be obtained, whereas if all the highlights were outside this would result in a GBR of 0.

6.5. Adversarial attacks

For testing the GVE model, we first compute adversarial samples by applying the described fast gradient sign method. In this experiment, we restrict ourselves to a qualitative analysis as evaluating the explanations generated by the adversarial samples requires expert knowledge. We focus on obvious wrong explanations (ex. 'a yellow beak' for a bird with a black beak). We also examine the effect of the ϵ parameter by visual inspection.

In addition, for evaluating the robustness of the image classifier, we compute its accuracy on the test dataset, and compare it with the accuracy computed on the generated adversarial samples. For this experiment, we fix $\epsilon = 0.1$.

6.6. Attribute Chunker

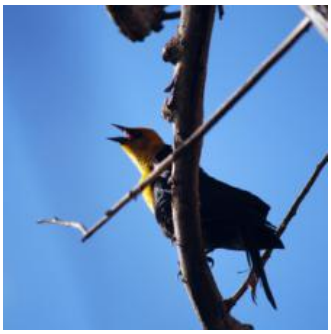
We used `spaCy`'s Dependency Parser (Honnibal & Johnson, 2015) to parse the explanation. We then find all nouns and their `amod` (Adjective Modifier) dependencies. For the sentence 'this bird has a yellow crown ...', `spaCy` detects 'crown' as a noun with adjective modifier 'yellow'. This method has high precision but low recall: the detected attributes are almost certain to be correct, but some attributes might be missing.

6.7. Generating counter-factual explanations

Given two instances from two distinct classes, a counterfactual explanation states why one image does not belong to the opposed class. Ideally, we would like to generate such an explanation from a model, however due to time constraints we built a rule based generator. We use the at-

(a) This is a *Summer Tanager* because this bird is red with black on its wings and has a long pointy beak

(b) This is a *Yellow breasted Chat* because this bird has a yellow belly and breast with a black superciliary and white wingbars



(c) This is a *Yellow headed Blackbird* because this bird has a grey crown a yellow breast and a white belly

Figure 1. Explanations generated by the model

tribute chunker to identify attributes in the explanation of both instances. Then, the following rules are used: (i) If there are overlapping attributes with different `amod` dependencies, add them to the counter-factual explanation. (ii) if an attribute is present in the true instance and not in the false instance, add it. A few examples of counter-factual explanations generated by this approach are presented in Figure 5.

7. Results

7.1. Experiments

Generating explanations. After training the GVE model, we obtained a METEOR score of 27.4 and a CIDEr score of 49.7 on a test set with 5.297 images. These values are slightly lower than those reported in (Hendricks et al., 2016). We attribute the differences to variations in the implementation and the fact that we did not fine-tune the CNN which had been pre-trained on the Imagenet dataset. In spite of this we note that the results are sensible, as can be seen in some of the examples shown in Fig. 1.

Introspection. When calculating Grad-CAM with respect to the complete explanation, we obtained a GBR of 71% with a standard deviation of 10% on the test set, thus suggesting that most of the highlights provided by Grad-CAM lie within the bounding box containing the bird. An examination of these highlights (see Figure 4) shows that this is indeed the case.

In the second approach, where we obtain highlights per attribute obtained by the chunker, we observed that while for some of the attributes the highlights would appear in the corresponding parts of the bird, others would often appear in non-related regions of the image (see Figure 3). We attribute this to the fact that sequential predictions made by the LSTM stack modify gradient computations so that attributes cannot be directly related to regions in the image when using Grad-CAM.

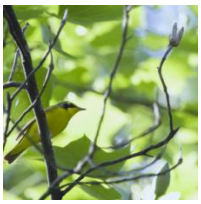
Adversarial attacks. We first observe that the adversarial samples generated do change the model's explanation, sometimes drastically, depending on the value of ϵ. Some explanations are obviously wrong, so the model is not very robust to adversarial attacks. However, we note that unless we apply a really high ϵ to the model, the explanations of the adversarial examples are relevant to the untrained eye. A brief qualitative analysis is presented in Figure 2.

For the second experiment, we first note that the image classifier achieves a reasonable accuracy of **70.27%** on the test set. On the adversarial samples however, it only achieves an accuracy of **0.44%**, a huge decrease in performance. This indicates that the image classifier, and the GVE model (which uses both the features and the output of the image classifier), is clearly not robust.

7.2. Demo

We have created a website with a demonstration that integrates the ideas treated in the present work (see Figure 6):

- **Generating explanations.** After selecting an image, the explanation is shown and the highlights produced by Grad-CAM are superimposed on the image.
- **Adversarial attacks.** The input image is modified so that changes are not visually perceptible and the modified explanation is shown.
- **Counter-factual explanations.** An explanation is given and one of two images must be selected. When doing so, a factual explanation is given on the correct choice, and a counter-factual explanation is given for the incorrect one.



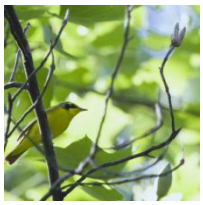
(1a) this bird has a yellow belly and breast with a short pointy bill

(2a) this bird has a white belly and breast with a black crown and long pointy bill

(3a) this bird is black with red eyes and has a long pointy beak

(4a) this bird has a white belly and breast with a blue crown and wing

(5a) this bird has a white belly and breast red crown and clack coverts and white secondaries



(1b) this is a bird with a small pointed bill beautiful white eyebrow and yellow breast

(2b) this is a white bird with black crown bright wing feathers and a thick sharp bright orange bill

(3b) this is a bird with a black head back wings and tail with a bright yellow breast belly and abdomen

(4b) this is a bird with a blue head bright yellow body and orange tail

(5b) this is a bird with a yellow breast gray back and white and black rectrices

Figure 2. Explanations of the original (top row) and the adversarial images (bottom row). Note that the images appear to be practically the same for the human eye. The explanations for the adversarial images even start off different ('this is a bird ...'). Even though the explanations for (1), (2), (3) are different, they explanations seem reasonable to the untrained eye. However the explanations for (4) and (5) are clearly erroneous. All images use $\epsilon = 0.1$

8. Discussion

Algorithm Reliability. The motivation for explainable AI is guided by legal and privacy aspects. The new European General Data Protection Regulation [2] (GDPR 2016/679 and ISO/IEC 27001) enforced on May 25th 2018 forbids usage of AI systems which are not able to explain why a decision has been made. Explainable AI has become so important that the U.S. Defense Advance Research Projects Agency (DARPA) has set up an Explainable AI program on its agenda [3]. The realm of explainable AI contrasts with black box systems where even the designers can not retrace class decisions of deep neural networks. Hence, auditing decision mechanisms in AI systems are vital to ensure that the learned decision rules reflect the implicit desires of the human system designers. However, even the models described in this paper suffer from problems. They are clearly sensitive to adversarial attacks as we showed in previous sections. However, recent results (Kurakin et al., 2018) show that even models specially designed to withstand these attacks have a long way to go. However, we hope that Explainable models pave the way forward, as they are especially good at determining the failure modes of a model.

The network architecture as described in this research could also be applied to medical images. The decisions made by the AI system then have an impact on human health and should be able to align with the decisions made by experts in the field.

Data Governance. Data governance covers the maintenance and management of data to avoid the existence of poor data. The models described in this paper can contribute significantly to improved data governance. For instance, they can be used as in the data auditing process: If a regulatory body requires a justification of a classification decision, or if there are ethical concerns regarding use of A.I in certain areas (e.g. use in policy making or health care), then an explanation model can be used to validate decisions.

Security. Security is crucial for many applications, especially so for the models described in this paper. We discuss it from two perspectives: whether the model can be manipulated, and the security of the interface itself.

We have already shown that the model itself can easily be manipulated to provide differing explanations. However, the method we have used is an example of a non-targeted 'white box' attack: we have access to the model itself and its

[2] https://www.eugdpr.org/

[3] https://www.darpa.mil/program/explainable-artificial-intelligence

internal representation. An attacker would not have access to this, therefore, it is protected against white box attacks. We note that it is still sensitive to other attacks, specifically 'black box attacks with probing' (Kurakin et al., 2018) which do not require internal details, but allow an attacker to observe outputs for a given input (which we provide via a REST API). This requires multiple calls to the API, as described next.

To make sure the data being sent to the client does not overlap we use separate instances of the explainer to generate explanations. The application is designed using REST architecture which allows state of the art security mechanisms such as `OAuth`. As our application is built solely for demonstration purposes and as we have no individual user profiles we do not require any authentication or authorization methods.

Privacy. Since the dataset we used for the model does not contain personal information, privacy is not a direct issue. However, such an explainable model could be used in applications that, if the data is not fully anonymized, could lead to a violation of privacy, and it could be used to target specific individuals according to the explanations given by the model when applying it in industries such as health care, and the financial, governmental and telecommunication sectors.

9. Conclusion

We have reproduced the GVE architecture in order to obtain an explanation model to which we can apply introspection using the Grad-CAM technique. The highlights obtained for a complete explanation were validated using the fraction of highlights falling within the ground-truth bounding boxes and showed that the GVE model is using most of the body of the birds to produce an explanation. We also applied Grad-CAM to individual attributes. In some cases the highlights would lie on the corresponding attributes, although in other cases parts of the environment would be highlighted. As future work we propose further tuning of the visual model, since we used a model pre-trained for the ImageNet dataset but did not fine tune it for the bird classification task. This could improve the performance of the explanation model and the information provided by introspection. Additionally, we propose further work on the attribute chunker so that it can be effectively used with Grad-CAM to obtain better insight on the generated explanations.

We generated adversarial attacks for the GVE architecture and found that it is fairly robust to such attacks when using the fast gradient sign method, as the explanations would change but still remained sensible, although in other cases errors were introduced. However, we saw that the image classifier part of model itself is extremely sensitive to adversarial attacks. Clearly, the robustness of this model needs to be improved. This is of special importance for the GVE model since the explanations should be secure to such attacks before it can be used with its intended purpose. These results are yet to be validated on a larger scale, as this was evaluated by examining particular examples. Different attack procedures can be implemented as well as targeted attacks, so that particular parts of an explanation could be changed.

For counter-factual explanations, the rule based function based on the attribute chunker gives good results. Improving the behavior of the attribute chunker will also improve the quality of the counter-factual explanations. An alternative approach is to provide pairs of images as input to train a language model. Higher computation will be required as the training complexity is n^2 for pairs of input images.

Our experiments in Explainable A.I. shed light onto how existing methods perform, with and without adversarial attacks, for end-user oriented tasks such as generating counter factual explanations.

References

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634, 2015.

Gao, Y., Beijbom, O., Zhang, N., and Darrell, T. Compact bilinear pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 317–326, 2016.

Goodfellow, I., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. URL http://arxiv.org/abs/1412.6572.

Hendricks, L. A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., and Darrell, T. Generating visual explanations. *CoRR*, abs/1603.08507, 2016. URL http://arxiv.org/abs/1603.08507.

Hendricks, L. A., Hu, R., Darrell, T., and Akata, Z. Grounding visual explanations (extended abstract). *CoRR*, abs/1711.06465, 2017. URL http://arxiv.org/abs/1711.06465.

Honnibal, M. and Johnson, M. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1373–1378, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL https://aclweb.org/anthology/D/D15/D15-1162.

Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Xie, C., et al. Adversarial attacks and defences competition. *arXiv preprint arXiv:1804.00097*, 2018.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Reed, S., Akata, Z., Lee, H., and Schiele, B. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 49–58, 2016.

Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., and Batra, D. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

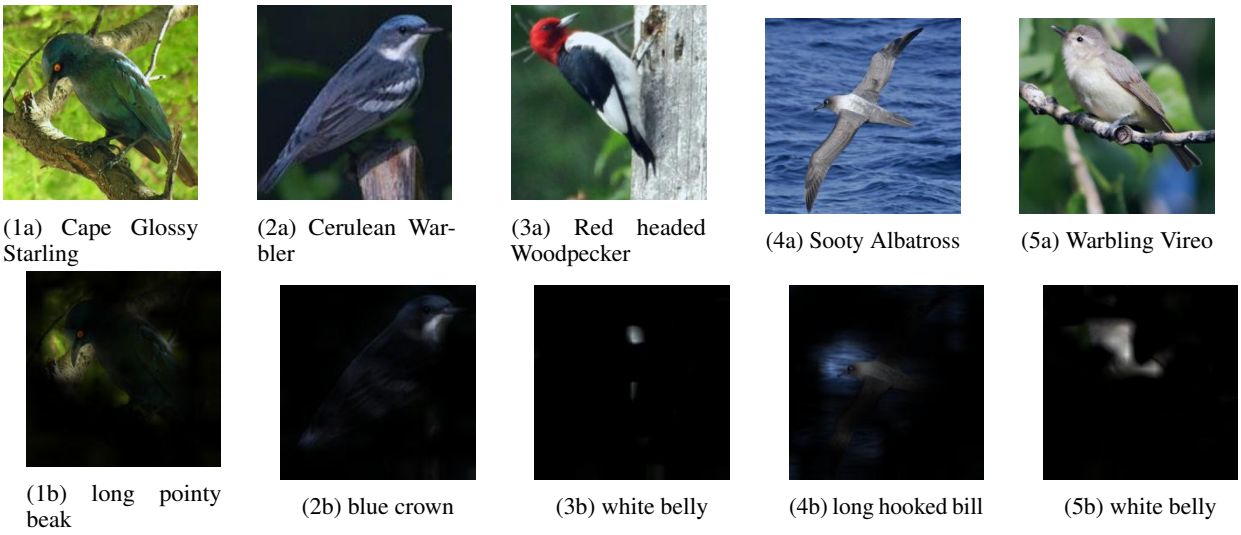
Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Vondrick, C., Khosla, A., Malisiewicz, T., and Torralba, A. Hoggles: Visualizing object detection features. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 1–8. IEEE, 2013.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

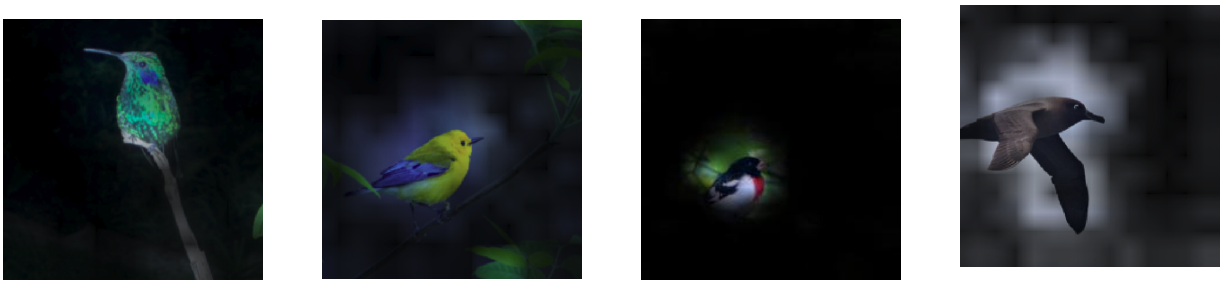
Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

A. Appendix



(1a) Cape Glossy Starling
(2a) Cerulean Warbler
(3a) Red headed Woodpecker
(4a) Sooty Albatross
(5a) Warbling Vireo
(1b) long pointy beak
(2b) blue crown
(3b) white belly
(4b) long hooked bill
(5b) white belly

Figure 3. Grad-CAM results. Given an input image (top row), we obtain highlights (bottom row) for attributes in the generated explanation. While some attributes are effectively highlighted, others are not, highlighting non-relevant regions of the image.



(a) Kentucky Warbler
(b) Blue Winged Warbler
(c) Rose Breasted Grosbeak
(d) White Breasted Kingfisher

Figure 4. Grad-CAM highlights obtained by backpropagating the complete explanation. The fraction of highlights that falls within the ground truth bounding box is measured with the GBR (see section 6.4).

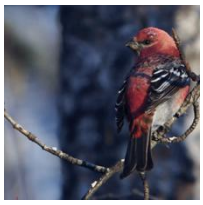
(1a) this bird has a bright blue head and a white belly

(2a) this bird has a black crown a black breast and a brown belly

(3a) this bird has a yellow belly and breast with a short pointy bill

(4a) this bird has a white belly and breast black and white wings and a red head with a long black bill

(5a) this bird is yellow with brown spots and a small beak



(1b) this bird has a red head and not a bright blue head

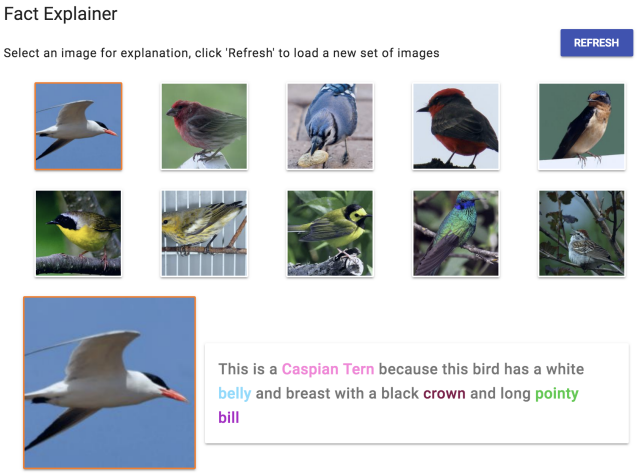
(2b) this bird has a white breast and not a black breast and has a blue crown and not a black crown

(3b) this bird has a long bill and not a short bill and doesn't have a yellow belly

(4b) this bird has a speckled belly and not a white belly and has a short bill and not a long black bill

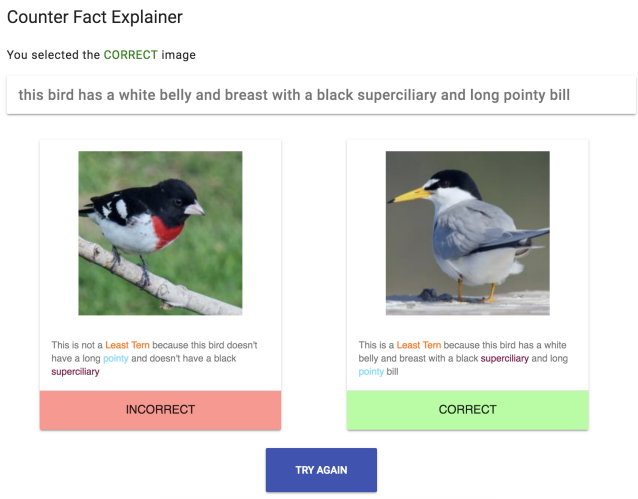
(5b) this bird doesn't have brown spots and doesn't have a small beak

Figure 5. Counter Factual Explanations: The true class' explanations (a) are compared with explanations of another class (b) to generate counter-factual explanations which tell why it wasn't chosen as the true class



(a) Fact Explainer

(b) Adversarial Explainer



(c) Counterfactual Explainer

Figure 6. Demo Interfaces