POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

A Human-in-the-Loop Approach for Post-hoc Explainability of CNN-based Image Classification

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Authors: Antonio De Santis, Matteo Bianchi

Advisor: Prof. Marco Brambilla

Co-advisor: Dr. Andrea Tocchetti

Academic year: 2021-2022

---

1. Introduction

Deep Neural Networks have transformed machine learning, with Convolutional Neural Networks leading the way in image classification. However, the decision-making process of these models has become increasingly complex and less transparent. Consequently, they are referred to as *black-box* models since their internal workings are too intricate for humans to understand. This opacity poses a serious concern as it leads to an absence of trust in AI decisions. Moreover, debugging black-box models is challenging without insights into their reasoning, which is necessary to address and fix errors and biased predictions. Recognizing this need for transparent and explainable AI, a new research area called Explainable Artificial Intelligence (XAI) has emerged. State-of-the-art XAI techniques for image classification produce heatmaps that highlight the pixels of an image that contribute the most towards the output. While these heatmaps offer insight into whether the AI is looking at the "right thing", they don't explain its decision process. Furthermore, such techniques provide *local explanations* (i.e., explanations for a specific prediction) that are difficult to generalize because the highlighted pixels are meaningful only in the context of the analyzed image. Consequently, heatmaps alone are difficult to aggregate for providing *global explanations* (i.e., explanations of the model's overall behaviour). In order to address these limitations, we propose a technique named Abstract Network Visualizations (ANV). ANVs are comprehensive local explanations for CNN-based image classification that provide a detailed view of the image features and patterns extracted by the CNN at each stage of execution, thereby providing an overview of the AI decision process. Moreover, these features, presented as heatmaps, are associated with a weight (i.e., the importance) towards the output and are described by a set of labels collected by means of a gamified crowdsourcing activity. The presence of labels improves the interpretability of heatmaps while allowing the production of global explanations by aggregating similarly labeled maps across multiple images.

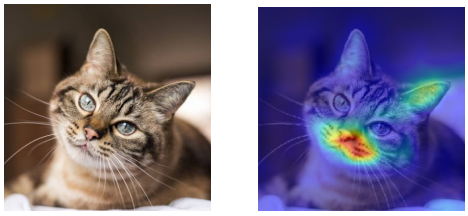
2. Related Works

First, we present a summary of the state-of-the-art explainability techniques for image classification. Following that, we provide an overview of how human knowledge has been effectively in-

corporated into this field.

2.1. Explainability

While there is no universal agreement on the definition of explainability, it can be described as a method of building an interface between humans and the AI system that can provide accurate explanations of the AI decisions that are also comprehensible to humans. There are two main categories of explainable AI: ante-hoc (i.e., redefining the architecture of a black-box model to improve its transparency) and post-hoc (i.e., providing explanations after the model has already been trained and deployed). The focus of this work is on post-hoc explainability, which can be further classified into model-agnostic and model-specific techniques. In the context of image classification, model-agnostic approaches focus on studying the input-output relationship of a model to compute an estimate of the importance of each region of the image. Therefore, they can be applied to any model regardless of its architecture or type. On the other hand, model-specific approaches are techniques for explaining the decisions made by a specific ML model. These methods try to "open the black-box" and reverse engineer its internal structure to provide insights into the model's decision-making process. Class Activation Map (CAM) is a model-specific XAI technique that can generate heatmaps highlighting the most important regions of an image for a particular class. This is achieved by replacing the fully connected layers with a Global Average Pooling (GAP) layer to reduce the feature maps into a single scalar value. Then, the class-specific weights of the GAP layer, which represent the feature maps' importance, are used to perform a weighted linear sum of the feature maps which results in a class-specific CAM (i.e., an heatmap highlighting where the network identified the class). *Selvaraju et al.* [3] later introduced a generalized version of CAM called Gradient-weighted Class Activation Mapping (Grad-CAM) which can be applied to any CNN architecture, without introducing a GAP layer. The main idea behind Grad-CAM is that the weights needed to combine the feature maps can be calculated by applying a global average pooling on the gradients of the score (before the softmax) for a given class with respect to the feature maps. The final maps for a specific class are then generated by applying a ReLU to the weighted sum of the feature maps, ensuring that only features that positively impact the prediction are taken into consideration. Grad-CAM was a big step forward thanks to its computational efficiency and wide applicability. However, it lacks an explanation of why the highlighted regions are relevant since heatmaps interpretation is highly subjective. For example, if Grad-CAM highlights the region of a cat's nose as being important for the prediction "cat", this might suggest that the network is using information about the nose to make the prediction, although it is impossible to know for certain. Following on the cat's image example, it could be that the network is using information about the texture of the fur near the nose, or about the presence of whiskers, to make its prediction (see Figure 1).



(a) Cat input image. (b) Grad-CAM.

Figure 1: Grad-CAM highlights the region near the nose, but it's hard to understand whether the network learnt to recognize the nose specifically or possibly the fur or whiskers.

A different approach for identifying features learnt by a neural network is Testing with Concept Activation Vectors (TCAV). The method works by inputting example images containing only one specific feature and observing the network's predictions. Both Grad-CAM and TCAV are very effective in detecting biases and explaining AI decisions but, at the same time, they may introduce bias in the explanations. For Grad-CAM, this happens when humans misinterpret why a region is highlighted due to biased assumptions, while for TCAV the bias can be introduced by the images that are selected for representing a feature.

2.2. Human-in-the-Loop

Despite the multitude of techniques and advancements in AI explainability, there are still

limitations in ensuring that explanations are fully accurate and understandable from a human perspective. For this reason, many researchers have turned to human-centered (i.e., human-in-the-loop) techniques that utilize human insight and reasoning to improve explanations of machine-learning models. Among these, *Lu et al.* [2] employed human knowledge to evaluate visual explanations generated by different XAI techniques such as Grad-CAM. They proposed a gamified crowdsourcing activity based on the game "Peek-a-Boom". In their implementation, only a small part of an image is initially shown, starting from the region deemed the most important by an explainability method. If the player cannot guess the image, more pixels are revealed. The number of pixels needed for the player to guess correctly determines a score for each explainability method. Another approach that focuses on global explainability was introduced by *Balayn et al.* [1]. They suggested augmenting the heatmaps generated by explainability methods by incorporating semantic concepts through crowdsourcing annotations. The main advantage of their approach is that the annotations can be aggregated, allowing the use of different statistical mining techniques to generate global explanations about the model behaviour. Overall, their method demonstrated the value of incorporating human knowledge in the explainability of ML models, hence foreseeing a promising direction for advancing the field of XAI.

3. Methodology

State-of-the-art XAI methods can provide explanations for image classification predictions by identifying the most important region of the image that contributed to the prediction, but they do not offer a complete understanding of the machine rationale. Hence, we need to examine the feature extraction process that happens through multiple layers in order to understand the entire AI decision-making process. Striving to cover such a need, we developed a process to generate post-hoc local explanations in the form of Abstract Network Visualizations (ANV) which provide a detailed view of the image features and patterns the CNN identifies at each stage of its execution. An ANV is composed of layers, each consisting of heatmaps that represent the areas of the input image where important features were identified. These heatmaps represent groups of feature maps clustered by similarity (i.e., feature maps focusing on the same region of the image are grouped together). For each heatmap, the final visualization also includes the relative importance of its corresponding group of feature maps with respect to the final predicted class and a set of crowdsourced labels that indicate the human concepts it represents. The ANV can be built considering all layers of the CNN, as well as a selected subgroup of interest. In particular, shallow layers usually focus on detecting basic shape information (e.g., edges, outlines, corners, etc.). Hence, it might be more efficient to focus on deeper layers which should contain more semantic concepts as their receptive fields are bigger. A significant advantage of using crowdsourced labels to describe the extracted features is the ability to aggregate multiple local explanations, thereby extending the explanation from a local to a global perspective. By analyzing the features extracted by a CNN to recognize a particular class across multiple images, we can develop a global explanation of how the network generally identifies that class. Building ANVs requires three steps. The first step is *Feature Maps Analysis*, in which feature maps are clustered and merged. The second is *Human Knowledge Collection* in which labels are collected through crowdsourcing and the last is *Label Analysis* in which the collected labels are post-processed to make them structured and free of errors.

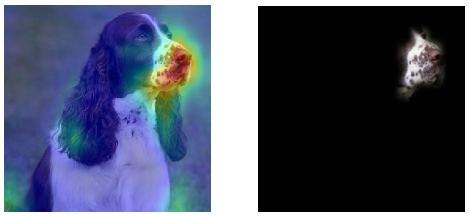
3.1. Feature Maps Analysis

Feature maps are extracted after applying the activation function for each convolutional layer. Next, feature maps are associated with their corresponding class-specific weights towards the predicted class, computed using a local explainability method. For this purpose, we used Grad-CAM as it is a straightforward approach that works with any CNN architecture. We performed unit normalization to enhance the interpretability of these weights. This technique guarantees that the total weight for each layer sums up to one, allowing for the importance of feature maps to be visualized as percentage per layer. Since the number of feature maps per layer can often be in the order of hundreds or more, which can be an overwhelming amount

of information for humans to handle, we decided to cluster and merge them together to generate representative heatmaps that we refer to as cluster maps. However, some pre-processing steps are necessary before proceeding with the clustering process. More specifically these steps consist of applying min-max normalization and dimensionality reduction. For the latter, we use a combination of two techniques: Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE). After pre-processing, we apply a clustering algorithm for each layer. In our method, we use Agglomerative Clustering, a widely employed type of Hierarchical Clustering. The number of clusters can be different for each layer and is selected by computing the average silhouette score for a range of 3 to 8 clusters. Once the clustering process is complete, each cluster is merged using a weighted average approach. This produces cluster maps representing an entire cluster of feature maps. Each cluster map is assigned a weight value that indicates its significance towards the predicted class. This value is computed by summing the weights of all feature maps belonging to that cluster.

3.2. Human Knowledge Collection

The goal of this step is to collect labels through crowdsourcing, representing the human concepts highlighted in each cluster map. Before designing the crowdsourcing activity, we need to address what precisely participants should see during the labeling of cluster maps. The general approach to obtain an interpretable visualization of a feature map is to generate an overlay of the input image and the feature map, obtaining an image such as the one shown in Figure 2a. However, making humans aware of the image before labeling will most likely cause a loss of focus on the highlighted areas. Hence, it is necessary to hide the non-highlighted portions to prevent such behaviours. This can be achieved by computing a mask (i.e., a binary image) that defines which pixels to show. A masked image is then obtained by overlaying the mask on top of the input image, as shown in Figure 2b. For what concerns the crowdsourcing activity, we employed a gamified approach as we wanted to make participants behave similarly to the neural network, by having them observe and analyze features to guess the correct class. The actual activity consists in playing an online game we designed called *Deep Reveal* in which participants are presented with the masked image of a cluster map and are required to guess its class and specify which features they recognized that helped them guess. These inputs are then used as labels for the cluster maps. Similarly to the Peek-a-Boom game described in Section 2.2, users of *Deep Reveal* can gradually increase the displayed area up to five times, allowing them to get more clues.



(a) Cluster Map. (b) Masked overlay.

Figure 2: An example of an overlay of a cluster map that focuses on a dog's muzzle, along with its corresponding masked image that reveals only the highlighted area.

3.3. Label Analysis

After collecting sufficient labels for each cluster map, we proceed with the label analysis step. First, we perform data cleaning on the collected labels. More specifically, labels consisting of multiple words are split into different labels, and stop-words are discarded. Then, we manually map labels referring to the same feature to a single word (e.g., "column", "pillar" and "pilaster" all become "pillar") to handle synonyms and misspellings. The next step is to evaluate each label by assigning them a score that allows us to emphasise the most relevant ones within each cluster. This score takes into account the label frequency as well as the percentage of the image revealed to the humans who assigned that label. Assigning a score to each label allows us to identify the labels that best describe their respective cluster maps. However, it is possible that certain cluster maps may represent the same feature and, therefore, be labeled in a similar manner. This can happen due to imperfections in the clustering process, or because the same feature is present in different regions of the image. Cluster maps are meant to represent the different features extracted at each layer, hence

the final step before constructing the ANV is to merge clusters with the same most relevant labels. The cluster maps are merged through a weighted average and the final weight is the sum of the weights of the merged maps. The labels of the merged clusters are also combined through a weighted average of their score. Finally, the ANV of an image can be generated by organizing its cluster maps, together with their weights and labels, into one column for each layer.

4. Experiments and Results

In this section, we describe the experiments we conducted to validate our methodology and discuss the final results obtained.

4.1. Experiment Setup

For the experiment, we first selected a CNN model to analyze. More specifically, we designed a model based on the standard VGG-16 architecture, which was trained using the Imagenette[1] dataset, a small subset of ImageNet consisting of ten classes. Its classes include Cassette Player, Chainsaw, Church, English Springer, French Horn, Garbage Truck, Gas Pump, Golf Ball, Parachute, and Tench. In our experiment, we analyzed 5 images per class, resulting in a total of 50 predictions to explain. Moreover, we focused our analysis on the last 9 convolutional layers since we had limited crowdsourcing resources and the initial layers primarily extract shape information (e.g., edges and outlines). After extracting, clustering, and merging feature maps using the methods discussed in Section 3.1, we obtained a total of 1954 cluster maps to be labeled. For the labeling phase, we deployed *Deep Reveal* as a web application and shared it with 210 participants. We allowed them to insert labels both in Italian and English to collect more labels, at the cost of having to perform a translation step during data analysis. At the end of this phase, we collected 9968 raw labels evenly distributed among the cluster maps. These raw labels were then split into single words and stopwords were removed. Afterward, the translation from Italian to English was performed and manually validated. The validation step was especially important as certain labels held vastly different meanings outside of their original context. For example, the Italian word "Esso" would typically translate to "it", but in the context of a gas pump, it referred to a company name. After handling synonyms and misspellings, the result was a refined set of 12082 single-word labels. Then identical labels are grouped for each cluster and are assigned a score as mentioned in Section 3.3. Finally, we merged cluster maps within the same layer based on their maximum-scoring labels, resulting in a total of 1192 cluster maps.

[1] https://github.com/fastai/imagenette

4.2. Results and Discussion

We conclude by discussing the resulting ANVs and their validity as a local explanation method for our CNN. Subsequently, we discuss the results obtained by aggregating the labeled heatmaps across different images to obtain global explanations. All detailed results for both the ANVs and the global explanations of the 50 images are available online[2].

4.2.1 Abstract Network Visualizations

Based on the structure of the ANV outlined in Section 3, we organized the cluster maps into a column for each layer, displaying their respective weight and highest-scoring label. Moreover, each feature has a detailed visualization which includes a plot of all labels with their respective score, the number of feature maps comprising the cluster, and additional game-related information such as the masked image, wins, losses, and resigns. From the results obtained, we observed that ANVs are capable of providing a comprehensive overview of the features that contribute to correctly predicting a class and insight about the identification process of such features. Furthermore, these visualizations also offer valuable insights into less obvious features. For instance, we found that the network was able to associate the concept of orange color with a chainsaw's motor or the concept of scales with a tench. On the other hand, we acknowledge the need to incorporate more sophisticated validation approaches to check the validity of these labels. This is because the fact that humans are able to classify an image using a certain feature suggests that the network may do the same, but it doesn't necessarily imply that it does. Moreover, we noticed that the lack of

[2] https://github.com/antonio-dee/abstract-network-visualizations

domain knowledge may oversimplify the visualization in some cases, hence a combination of expert and non-expert users could offer different levels of detail and obtain a broader view. Additionally, we showed that another advantage of ANVs was their capability of showing when the network utilizes contextual clues to help in its predictions. For example, the presence of trees to identify a chainsaw or the presence of a golf club to classify a golf ball.

4.2.2 Global Explanations

We construct our global explanations in a hierarchical way, grouping layers three by three, thereby extending the concept of the ANV to a global perspective. Features are ordered by their label's global score (i.e., the sum of the scores of a label when it appears as the highest-scoring one), which serves as a measure of label quality across multiple images. For each feature, the visualization provides a set of cluster map examples and an average weight. Moreover, we showed that it is possible to generate simple and straightforward global explanations by combining all layers together, as shown in Figure 3. However, such explanations provide a lower level of detail compared to the previously described ones.

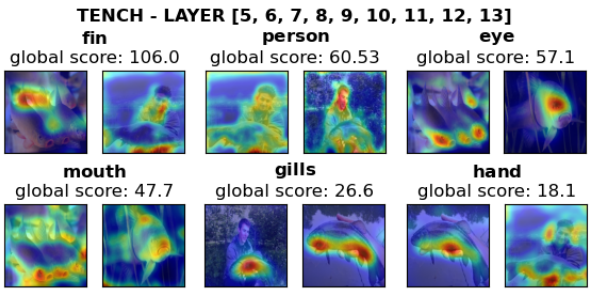


Figure 3: A straightforward global explanation for the class "Tench". Weights are not present as they may lose meaning if averaged out between shallow and deep layers.

It is important to note that our experiment was limited by the relatively small sample size of five images per class. As a result, our global explanations should be considered a preliminary approach to the problem rather than a definitive result. Due to the limited samples, there may be bias in our findings, since these few images may not be enough to represent an entire class.

5. Conclusions

We introduced a novel approach named Abstract Network Visualizations (ANV) to generate local post-hoc explanations in the context of CNN-based image classification. Our method clusters and merges feature maps to produce detailed visualizations of the extracted features at each layer. Moreover, we associated each feature with labels to facilitate human interpretation using an image-guessing game called *Deep Reveal*. Finally, we showed that aggregating these labels allows for the generation of global explanations. Our experiments demonstrated the potential of our explainability method, although open questions still remain. These include finding techniques to validate the correctness of these labels, possibly combining our method with TCAV, and exploring the possibility of associating CNN filters with labels to allow the generation of ANVs concurrently with the CNN execution, meaning that the crowdsourcing step is needed only once per model.

References

[1] Agathe Balayn, Panagiotis Soilis, Christoph Lofi, Jie Yang, and Alessandro Bozzon. What do you mean? interpreting image classification with crowdsourced concept extraction and analysis. In *Proceedings of the Web Conference 2021*, WWW '21, page 1937–1948, New York, NY, USA, 2021. Association for Computing Machinery.

[2] Xiaotian Lu, Arseny Tolmachev, Tatsuya Yamamoto, Koh Takeuchi, Seiji Okajima, Tomoyoshi Takebayashi, Koji Maruhashi, and Hisashi Kashima. Crowdsourcing evaluation of saliency-based XAI methods. *CoRR*, abs/2107.00456, 2021.

[3] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.