Model Interpretability

MEMORABLE AI

Varshini Reddy
15/7/22

Contents

- Introduction to model interpretation
- Standard interpretation techniques
    - LIME
    - KernelSHAP
    - GradCAM
    - Deconvolution
- Interesting recent interpretation techniques
- Work plan

Introduction to model interpretation

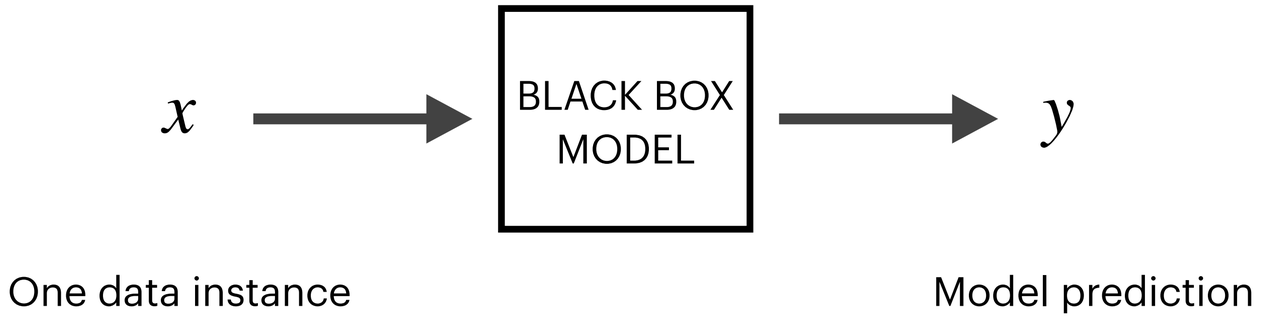
Interpretation Algorithms : Methods used to demystify the black-box nature of machine learning algorithms

Approaches;

- Interpretable models
  - Models that are intrinsically understood by humans
- Model agnostic methods
- Example based methods
  - Use an instance from the data to explain the behavior of the model

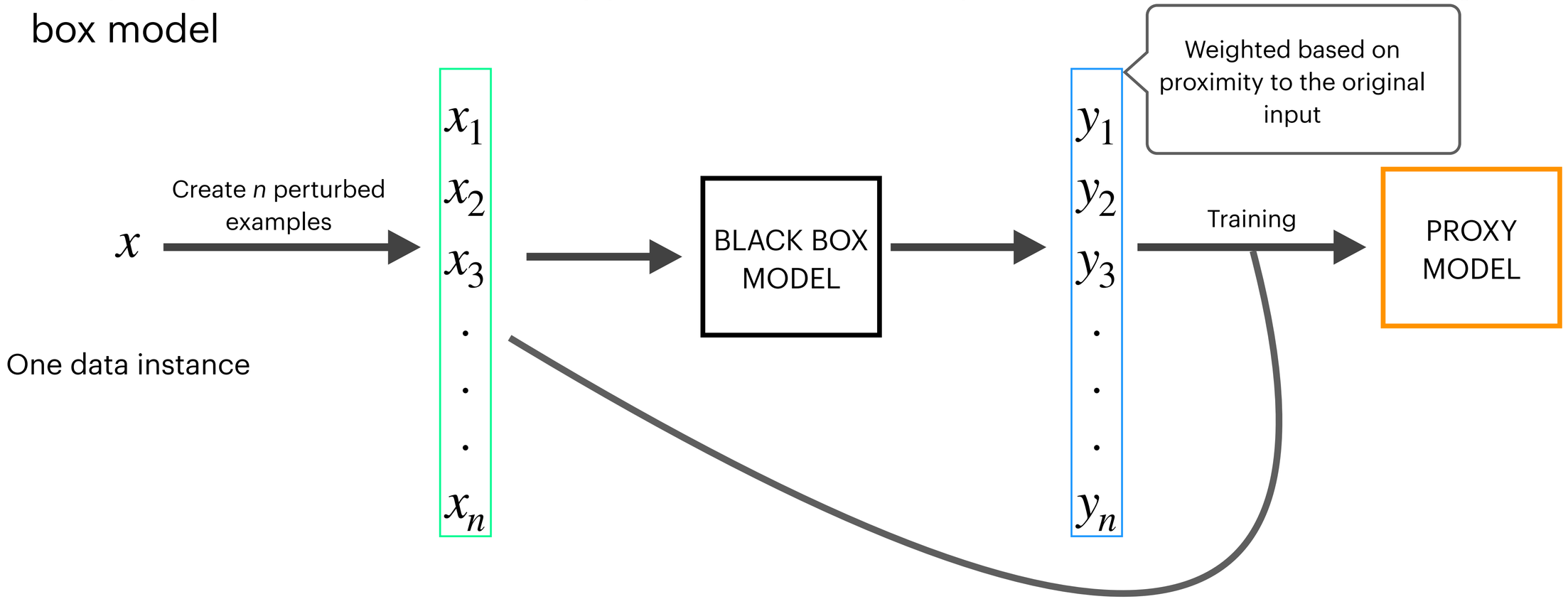
Standard interpretation techniques : LIME

- Local interpretable model-agnostic explanations (LIME) is a proxy model approach
- The proxy model is trained to approximate the local predictions of the underlying black box model



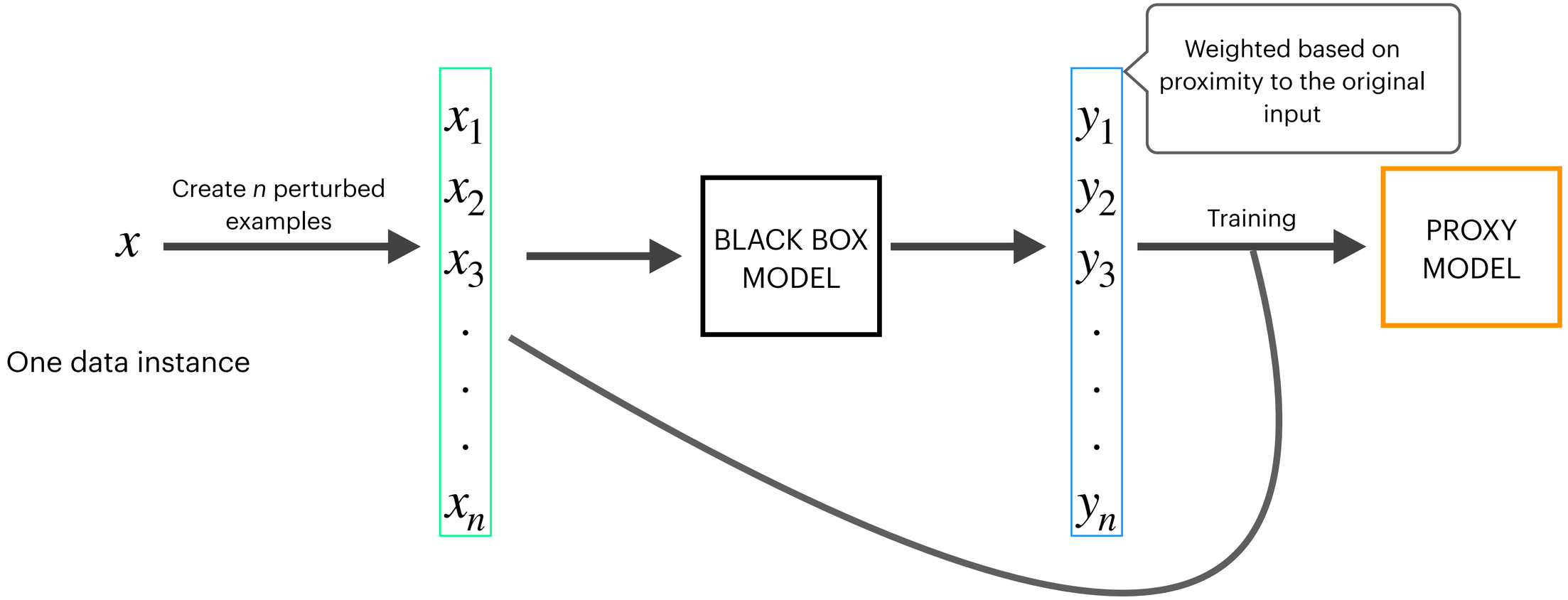
Standard interpretation techniques : LIME

- Local interpretable model-agnostic explanations (LIME) is a proxy model approach
- The proxy model is trained to approximate the local predictions of the underlying black box model



Standard interpretation techniques : LIME

- The proxy model used is an interpretable model such as linear or lasso
- The aim is for it to approximate the black box model for the data point in question



Standard interpretation techniques : KernelSHAP

Shapley Values: For each sample, take a combination of features and get the prediction and compute the difference in average prediction to this. A Shapley value of a feature is the average of the marginal contributions of features.

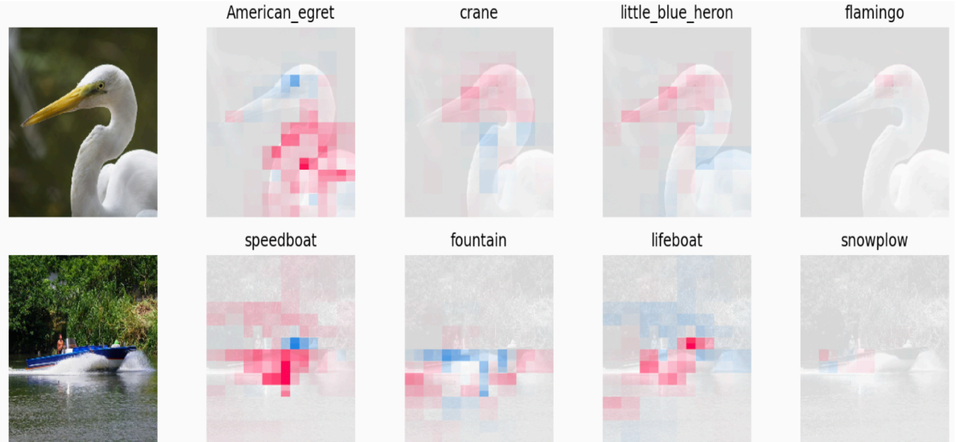
When a feature is not present in the combination, it is replaced by a random value sampled from that column.

In case of images, groups of pixels are considered as features (called super-pixels). A feature is removed by greying that group of pixels and passing the original pixels in the remaining part.

Standard interpretation techniques : KernelSHAP

KernelSHAP

- Create a dataset of feature combinations and the prediction of the black box model on this dataset
- Weight the sampled instances according to the weight the combination would get in the Shapley value estimation
    - The difference in how this weight is computed results in various research works
- Fit a weighted linear model
- The required values are the coefficients from the linear model



Standard interpretation techniques : GradCAM

It is a Class Activation Mapping technique

CLASS ACTIVATION MAPPING

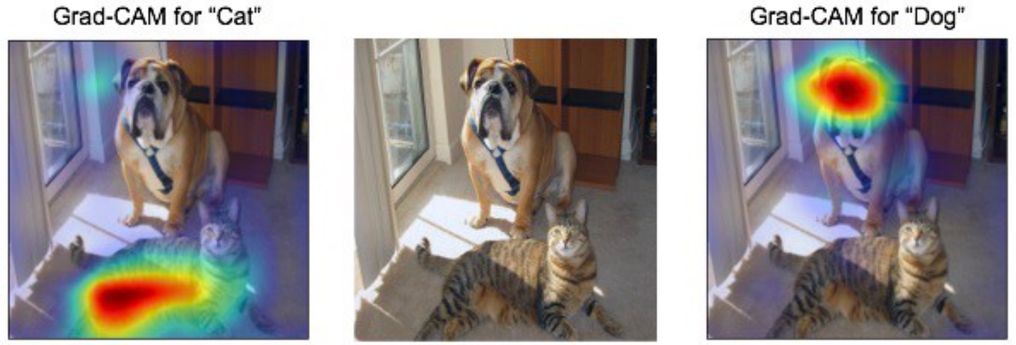
- The convolution architecture should end with a Global Average Pooling (GAP) layer followed by the output layer
- The weighted sum of the feature map of the last convolution layer is computed.
    - It is weighted by the weights connecting the GAP with the predicted class or class in question
- The final map is upsampled to the same size of the input image

Standard interpretation techniques : GradCAM

- In case of GradCAM, instead of GAP we compute the derivative of the logits with respect to the feature maps

$$\alpha_c^k = \frac{1}{Z} \sum_{i,j} \frac{\partial y^c}{\partial A_{i,j}^k}$$

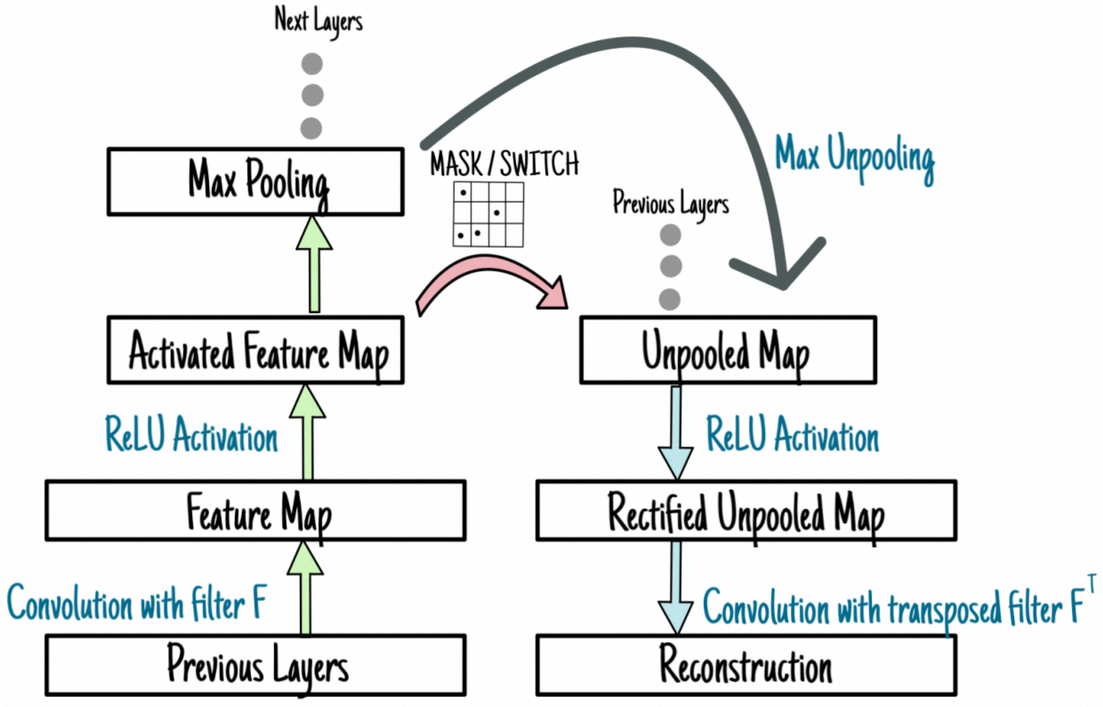
- An average over all the pixels of the feature map gives the importance α of a feature map k for a class c
- Combine the feature map as in CAM, but use the importance α rather than the weights. Apply ReLU on the output map and finally upsample



Standard interpretation techniques : Deconvolution

This technique recognizes what features in the input image an intermediate layer of the network is looking for

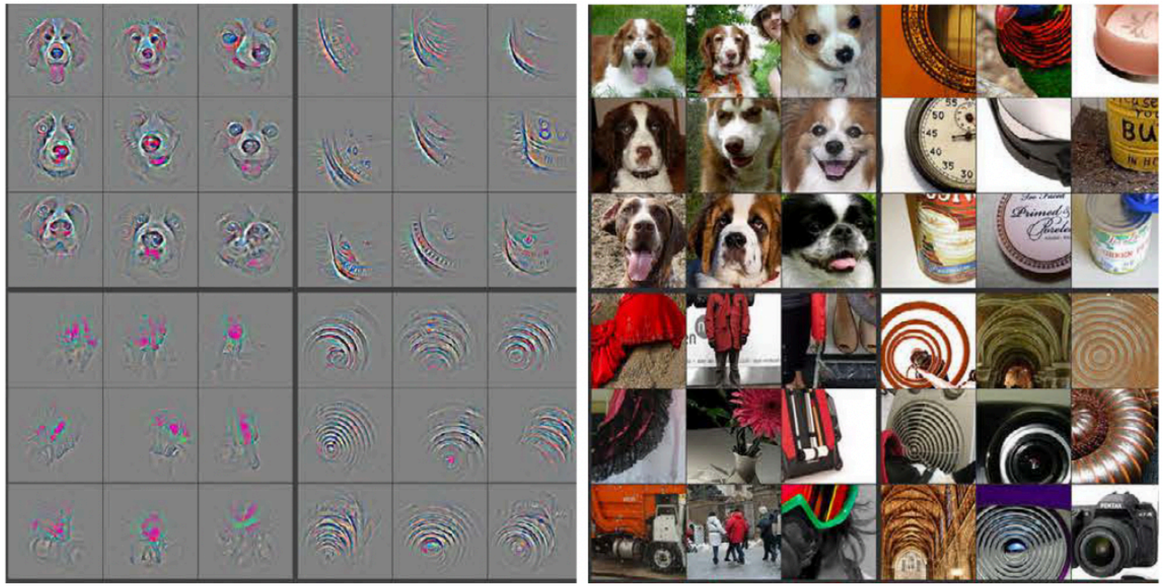
Backpropagates the derivatives of an activation map to the input



Standard interpretation techniques : Deconvolution

This technique recognizes what features in the input image an intermediate layer of the network is looking for

Backpropagates the derivatives of an activation map to the input



Recent interpretation techniques

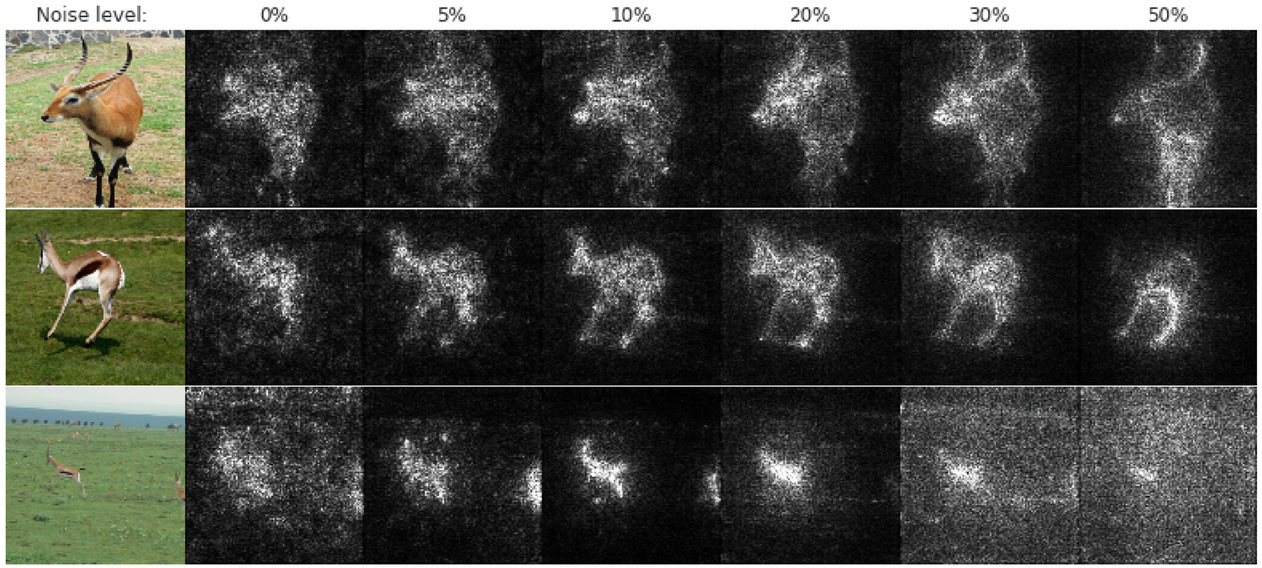
SmoothGrad: removing noise by adding noise

The core idea is to take an image of interest, sample similar images by adding noise to the image, then take the average of the resulting sensitivity maps for each sampled image.

In this approach, we differentiate the score function wrt the input to get a sensitivity map.

> This represents how much difference a tiny change in each pixel of the input image would make to the classification score.

These derivatives may fluctuate sharply at small scales i.e. small variations in the partial derivative and hence lead to noisy results. Hence, to every image add some Gaussian noise and then calculate the gradients.



$$M_c'(x) = \frac{1}{n}\sum_{1}^{n} M_c(x + \mathcal{N}(0,\sigma^2))$$

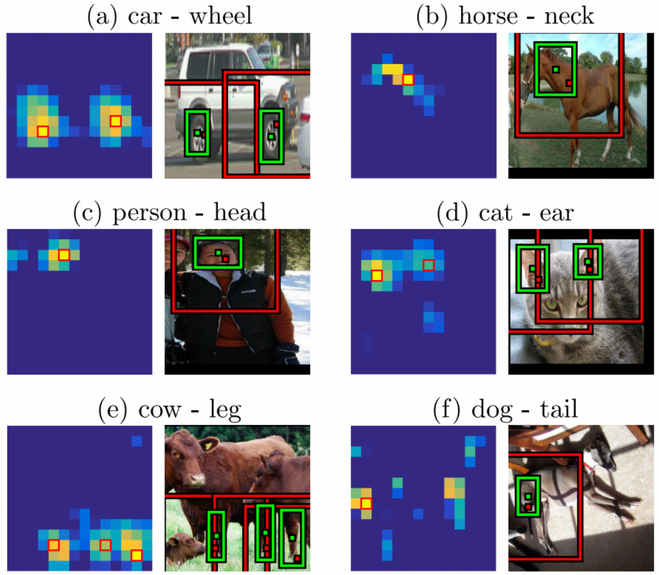
2018

Do semantic parts emerge in Convolutional Neural Networks?

Each pixel in a feature map (activation) indicates the activation of a particular position in an input image.

For each feature map, select all its local maxima activations. Each of these activations will lead to a stimulus detection in the image, regardless of its activation value (i.e. no minimum threshold).

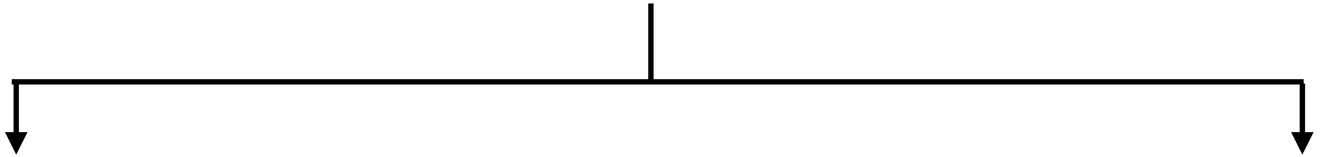
To factor out the elements of the image, a bounding box mechanism is used.



Object detectors emerge in deep scene CNNs

Shows that neural networks can localize objects during training and inference without explicitly being given a notion of object.

The basic idea is to simplify an image such that it keeps as little visual information as possible while still having a high classification score for the same category



Segmentation approach

Create a segmentation of edges and regions and remove segments from the image iteratively.

At each iteration we remove the segment that produces the smallest decrease of the correct classification score

At the end, we get a representation of the original image that contains, approximately, the minimal amount of information needed by the network to correctly recognize the scene category.

Based on this we can compute the percentage an object remains in a given dataset.

Occlusion approach

Generate a discrepancy map for each input image (This is done by creating occluded versions of the input image and and record the change in activation as compared to using the original image).

Center the discrepancy map around the spatial location of the unit (neuron) that caused the maximum activation for the given image.

Average the re-centered discrepancy maps to generate the final receptive field. Expect to see semantically meaningful objects of the input image.

Object detectors emerge in deep scene CNNs

These approaches will work for any model without any modification.

However, the results give the values for each layer and not the entire model.

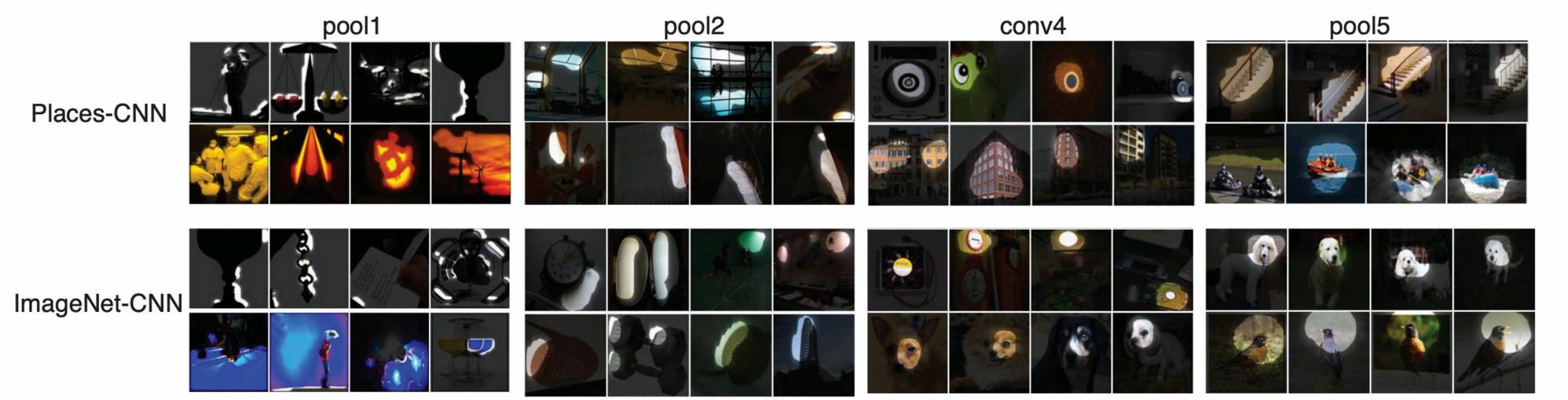
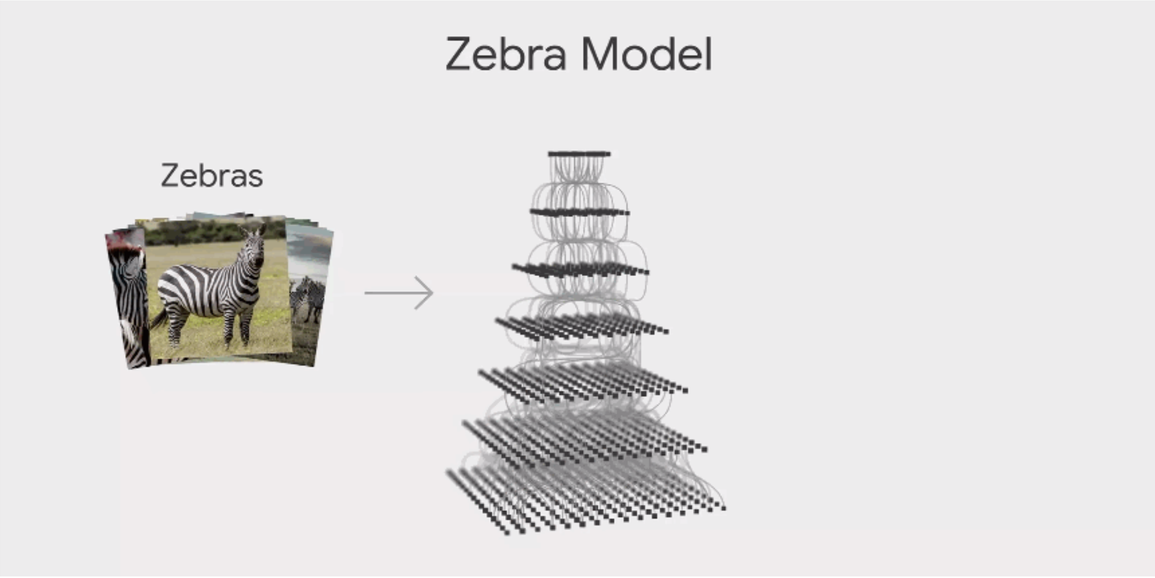


Figure 5: Segmentation based on RFs. Each row shows the 4 most confident images for some unit.

Robust Semantic Interpretability: Revisiting Concept Activation Vectors

- Quantifies the effects of semantic concepts such as textures, colors on individual model predictions
- RCAV calculates a concept gradient and takes a gradient ascent step to assess model sensitivity to the given concept
- CAV is generated by training a logistic regression on an intermediate layer to classify a given concept relative to the union of other concepts. It is the weight matrix of the trained logistic regression.



Example from Google's TCAV

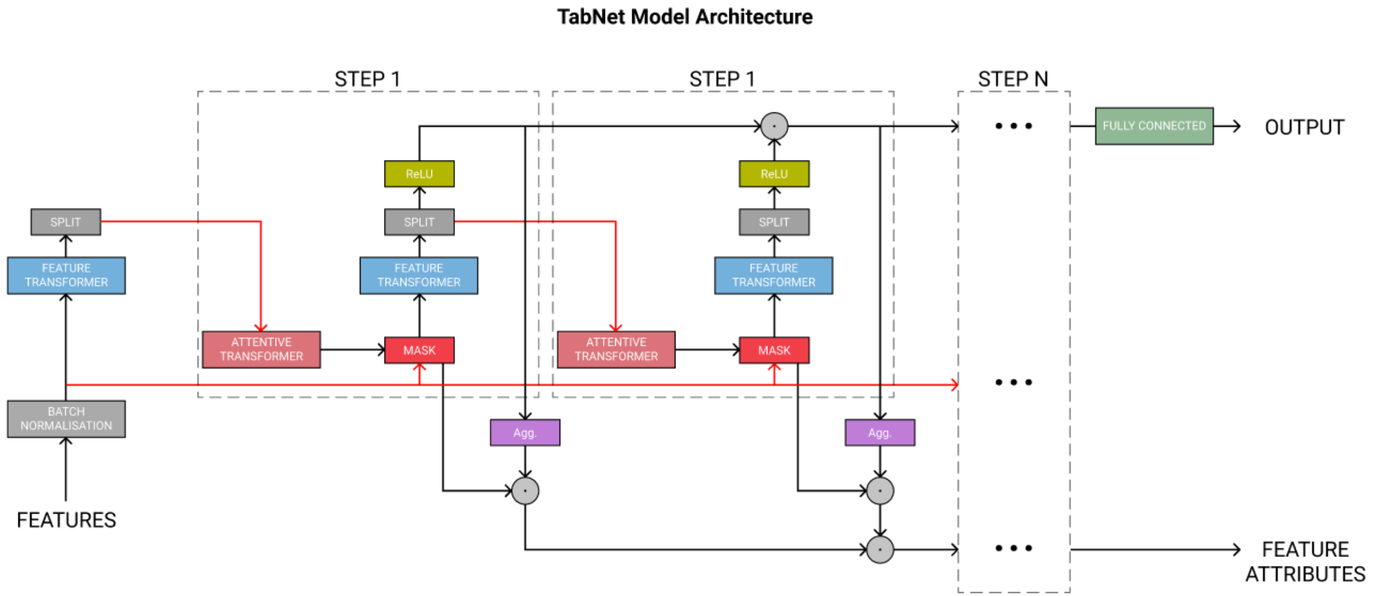
2021

TabNet: Attentive Interpretable Tabular Learning

It is a deep tabular data learning architecture that uses sequential attention to choose which features to reason from at each decision step.

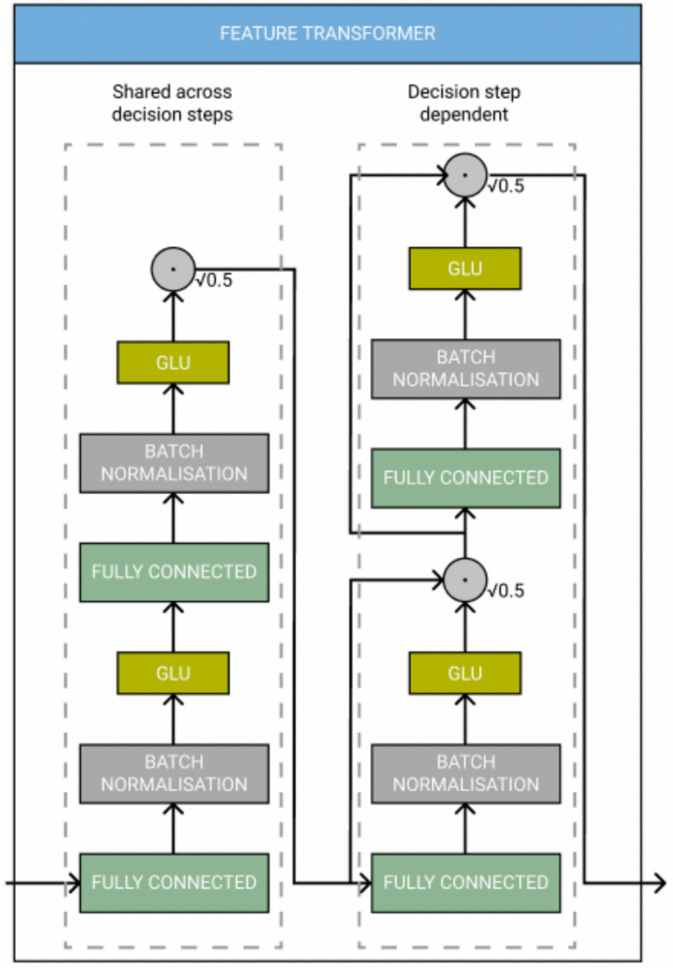
The TabNet encoder is composed of a feature transformer, an attentive transformer and feature masking.

TabNet uses instance-wise feature selection, which means features are selected for each input and each prediction can use different features.



TabNet: Attentive Interpretable Tabular Learning

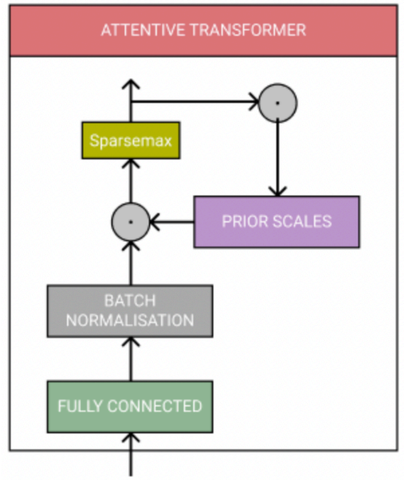
Feature Transformer - It is a network which has an architecture of its own. It has multiple layers, some of which are shared across every Step while others are unique to each Step.



Attention Transformer - Once features have been transformed, they are passed to the Attentive Transformer and the Mask for feature selection.

Prior scales is used to keep a track of how much each feature has been used by the previous steps.

This is used to derive the Mask using the processed features from the previous Feature Transformer.



Work Plan

1. Pixel-wise image importance + Semantic segmentation of image
2. In-painting to iteratively remove objects from an image to learn it’s affects on the model outcome
3. Saliency map on the input image for the conversion model by using an architecture with custom embedding architecture
4. Concept Activation Vector approach to understand the affect of different textures, objects, styles

Thank you