Explainable & Interpretable AI (XAI)

Yongzhe YAN

yongzhe.yan@etu.uca.fr
Philosophic question: What is interpretability?

- There is no complete or ultimate interpretability.
- Explaining AI is actually a human related task:
  - To what extent?
    - Psychological assurance: The ability to explain or to present in understandable terms to a human. [1]
    - Until we know how to debug/improve it.[2]
  - We want to interpret the model because it is still not perfect.

[1]: Towards A Rigorous Science of Interpretable Machine Learning
[2]: The Mythos of Model Interpretability
A broad view of interpretability on CNN

1. Understanding why and how CNN works (how to open the black box)?
   a. Visualizing the CNN representation
   b. Disentangling the CNN features (textures, colors, etc.)
   c. Mining the high-dimensional activation (especially FC)
   d. Explaining the causality of the input & the output (explain the decision process)
   e. Building CNN models combined with explainable models (decision trees)

2. Understanding how CNN is trained (how the black box is built)?
   a. Why/How CNN can be optimized by stochastic gradient decent?
   b. Why CNN can be well-generalized even though it is over-paramatized?
   c. How to find the correct capacity of the CNN model?

3. Using the knowledge from other domain
   a. Opening the black box of Deep Neural Networks via Information
   b. Why does Deep Learning work? A perspective from Group Theory
   c. Why Deep Learning Works: A Manifold Disentanglement Perspective
The interpretability is still **not** clearly defined

- Most of the survey/tutorials are more or less biased and cover only a part of the subject (mine as well).
- Several Surveys & Tutorials
  - Good to see this at first: [https://youtu.be/gCJCgQW_LKc](https://youtu.be/gCJCgQW_LKc)
  - [Interpretable Deep Learning under Fire](https://www.sciencedirect.com/science/article/pii/S0925231219302935)
  - [A Survey Of Methods For Explaining Black Box Models](https://arxiv.org/abs/1810.04197)
Why do we interpret the CNN?

- Debugging, diagnosing and improving CNNs
- Responsibility in medical, autonomous driving etc.
- Against adversarial attack in security & financial areas
- Compliance to legislation (GDPR)
- Curiosity
Why do we interpret the CNN?

- Also related to lots of other tasks
  - Weakly/unsupervised learning
    - Understanding the features and helps the transfer/weakly-supervised learning
  - Network redundancy reduction
    - Reducing the useless weights
  - Domain adaption / Style transfer
    - Understanding the latent representation of the CNN
Planning of the presentation

● What is interpretability?
● Why we do interpretability?
● How to interpret the CNNs?
  ○ How to **visually** explain the CNNs?
    ■ Perturbation based methods
    ■ Backpropagation based methods
    ■ Activation based methods
  ○ Others
    ■ How to understand the high dimensional FC layer?
    ■ Context/Data bias
● How to **visually** explain the CNN?
  ● Others
Perturbation based visualization

- Occlude a part of the image
- Verify how the correct class is changed
- Iterate two steps above on the entire image
Occlusion based methods - Disadvantages

- Time consuming
- Dependant on the occlusion size

Figure 1: Attributions generated by occluding portions of the input image with squared grey patches of different sizes. Notice how the size of the patches influence the result, with focus on the main subject only when using bigger patches.
RISE: Randomized Input Sampling for Explanation of Black-box Models
LIME - Theory

Local Interpretable Model-Agnostic Explanation

Figure 3: Toy example to present intuition for LIME. The black-box model’s complex decision function $f$ (unknown to LIME) is represented by the blue/pink background, which cannot be approximated well by a linear model. The bold red cross is the instance being explained. LIME samples instances, gets predictions using $f$, and weighs them by the proximity to the instance being explained (represented here by size). The dashed line is the learned explanation that is locally (but not globally) faithful.
LIME - Practice on Image (super pixel)

Original Image

Interpretable Components

Sample 1

Sample 2

Sample 3
LIME - Local Linear Regression

Original Image P(tree frog) = 0.54

Perturbed Instances P(tree frog)

<table>
<thead>
<tr>
<th>Perturbed Instance</th>
<th>P(tree frog)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Image</td>
<td>0.54</td>
</tr>
<tr>
<td>Image 1</td>
<td>0.85</td>
</tr>
<tr>
<td>Image 2</td>
<td>0.00001</td>
</tr>
<tr>
<td>Image 3</td>
<td>0.52</td>
</tr>
</tbody>
</table>

Locally weighted regression

Explanation

https://blog.csdn.net/qq_303022
LIME - Explain for each class
Perturbation based visualization - Conclusion

● Advantages
  ○ Model agnostic
  ○ Easy to implement

● Disadvantages
  ○ Time consuming

More methods:

Real Time Image Saliency for Black Box Classifiers

Interpretable Explanations of Black Boxes by Meaningful Perturbation

Towards Explanation of DNN-based Prediction with Guided Feature Inversion

EXPLAINING IMAGE CLASSIFIERS BY COUNTERFACTUAL GENERATION
Backpropagation based visualization

- **Gradient** Based
- **Deconvolution** Based
- **Weight Relevance** Based
Gradient Based Method - Saliency Map

Let's backpass the gradient!

Can be used for segmentation? Yes

indicates which pixels need to be changed the least to affect the class score the most.

softmax

Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps
Stanford CS230 Slide week 7
Gradient Based - Class Model Visualization

Given this trained ConvNet, generate an image which is representative of the class “dog” according to the ConvNet.

Noise Image

Keep the weights fixed and use gradient ascent on the input image to maximize this loss:

\[ L = s_{dog}(x) - \lambda \|x\|_2^2 \]

Gradient ascent:

\[ x = x + \alpha \frac{\partial L}{\partial x} \]

“x should look natural”

Repeat this process:
1. Forward propagate image x
2. Compute the objective L
3. Backpropagate to get dL/dx
4. Update x’s pixels with gradient ascent
Class Model Visualization - Results

We can do this for all classes:

- Goose
- Ostrich
- Kit fox
- Husky
- Dalmatian

Looks better with additional regularization methods.

Very different from GAN!

L2 Regularization

References:
- Understanding Neural Networks Through Deep Visualization
- Stanford CS230 Slide week 7
Deconvolution based method

Let’s backpass the **activation**!

**Motivation of DeconvNets for visualization:** Here is a CNN, trained on ImageNet (1.3m images, 1000 classes), we’re trying to interpret by reconstructing the activation’s zone of influence in the input space.

Visualizing and Understanding Convolutional Networks
Deconvolution based visualization

Visualizations of Layer 1 and 2. Each layer illustrates 2 pictures, one which shows the filters themselves and one that shows what part of the image are most strongly activated by the given filter. For example, in the space labeled Layer 2, we have representations of the 16 different filters (on the left).
Unifying Gradient & Deconv - Guided backprop

ReLU Backward Pass is tricky!

STRIVING FOR SIMPLICITY: THE ALL CONVOLUTIONAL NET
Guided Backpropagation

Target class: Mastiff (243)
Layer-wise Relevance Propagation (LRP)

Let's backpass the weight relevance!

\[
\begin{align*}
    r_{z_1 \leftarrow v_1} &= \frac{W_{1,1}^{(2)} z_1}{W_{1,1}^{(2)} z_1 + W_{2,1}^{(2)} z_2} v_1 \\
    r_{z_2 \leftarrow v_1} &= \frac{W_{2,1}^{(2)} z_2}{W_{1,1}^{(2)} z_1 + W_{2,1}^{(2)} z_2} v_1 \\
    r_{u_1 \leftarrow v_1} &= \frac{W_{1,1}^{(1)} u_1}{W_{1,1}^{(1)} u_1 + W_{1,2}^{(1)} u_2} r_{z_1 \leftarrow v_1} + \frac{W_{1,2}^{(1)} u_1}{W_{1,2}^{(1)} u_1 + W_{2,2}^{(1)} u_2} r_{z_2 \leftarrow v_1}
\end{align*}
\]
LRP - Visualization

Classification

Pixel-wise Explanation

Image $x$

Features

Classifier output $f(x)$

cat =
no cat =

$f(x) = \sum \text{Feature Relevances} = \sum \text{Pixel Relevances}$
Backpropagation based visualization - Conclusion

● Advantages
  ○ Quick to compute
  ○ Fine-grained interpretation

● Disadvantages
  ○ Low quality
  ○ Usually difficult to understand
  ○ Only for CNN (connectionism)
Activation based visualization - CAM

Hypothesis: Each channel on the last conv layer presents spatial information for an abstract concept (a dog head, a dog tail etc.).
Activation based visualization - Network dissection

From visualization to interpretation:

1. Define a broad dictionary of candidate concepts.

<table>
<thead>
<tr>
<th>Broden Dataset</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADE20K</td>
<td>Zhou et al, CVPR ’17</td>
</tr>
<tr>
<td>Pascal Context</td>
<td>Mottaghi et al, CVPR ’14</td>
</tr>
<tr>
<td>Pascal Part</td>
<td>Chen et al, CVPR ’14</td>
</tr>
<tr>
<td>Open Surfaces</td>
<td>Bell et al, SIGGRAPH ’14</td>
</tr>
<tr>
<td>Desc Textures</td>
<td>Cimpoi et al, CVPR ’14</td>
</tr>
<tr>
<td>Colors</td>
<td>generated</td>
</tr>
</tbody>
</table>

Total = 63,305 images
1,197 concepts

[Network Dissection: Quantifying Interpretability of Deep Visual Representations](#)
Network dissection

2. Test each internal unit on segmentation of every concept.
Network dissection

3. Measure segmentation quality and match units to concepts.

IoU of the best-matched concepts quantify interpretability
Network Dissection: Quantifying Interpretability of Deep Visual Representations
Activation based visualization - Conclusion

- **Advantages**
  - Easy to interpret

- **Disadvantages**
  - Coarse mask
  - CNN based
● How to **visually** explain the CNN?

● Others
Mining the high dimensional FC layers

1. Use t-SNE to embed the feature (dimension reduction)
   a. [https://harveyslash.github.io/TSNE-Embedding-Visualisation/](https://harveyslash.github.io/TSNE-Embedding-Visualisation/)
   b. the reason why we do the dimension reduction is under assumption of **strong correlation** between the neurons

2. Using a linear classifier probes

   Use an interpretable tool (linear classifier) to measure the capacity of each FC layer (Maybe wrong)
Linear classifier probes - results

Some counter-intuitive conclusions!

(a) After initialization, no training.  
(b) After training for 10 epochs.
Context - dataset bias

Examining CNN Representations with respect to Dataset Bias/Distribution

Where does the CNN look at when it classify the “wearing lipstick” attribute?
Context - dataset bias

Examining CNN Representations with respect to Dataset Bias

\[ Y_{black\ hair} = Yes \]
\[ Y_{smiling} = Yes \]

Representation of black hair

Representation of smiling.

Conflict

Great overlap

Ground-truth attribute relationship
Context - dataset bias

How facial attributes are correlated
Integrate explainable models into CNNs

(decision trees)
Thank you