

Explainable(X) AI

AI for Images

Sarah A. Carneiro & Caroline M. Rodrigues

Introduction



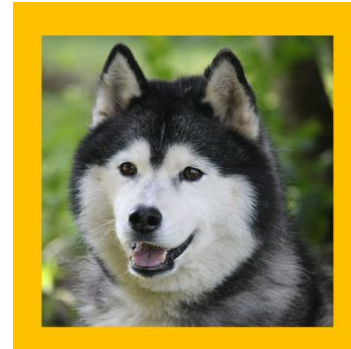
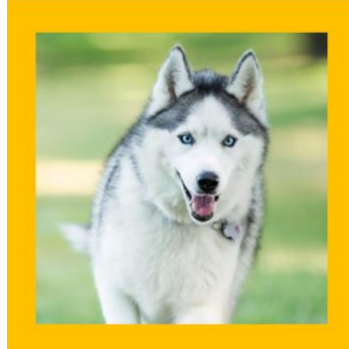
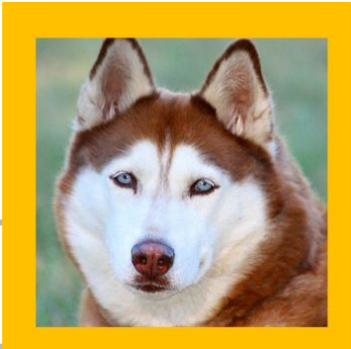
- XAI - Explainable Artificial Intelligence
- Domain that focuses on developing AI systems and models in a way that **allows humans to understand, interpret, and trust** their decisions
 - Provides insight into the **inner workings** of the algorithms and the **reasons** behind their outputs



Introduction

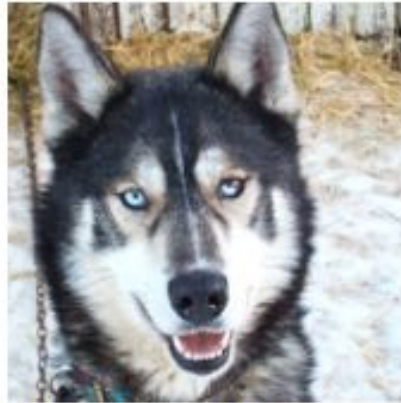


Wolves vs. Huskies classification



Introduction

- Accuracy ✓



(a) Husky classified as wolf

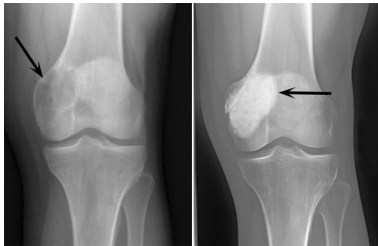


(b) Explanation

Introduction

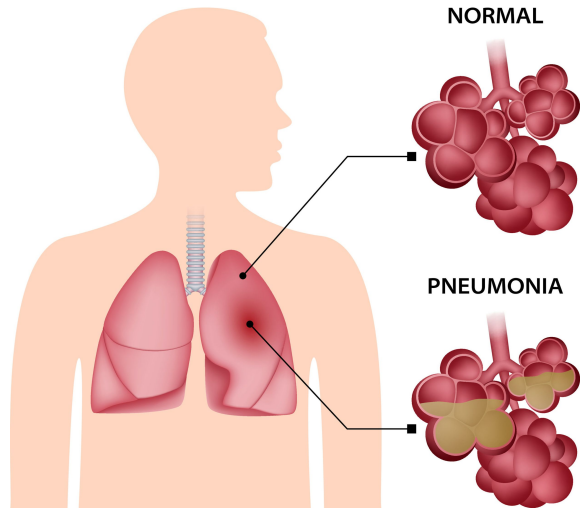


Introduction

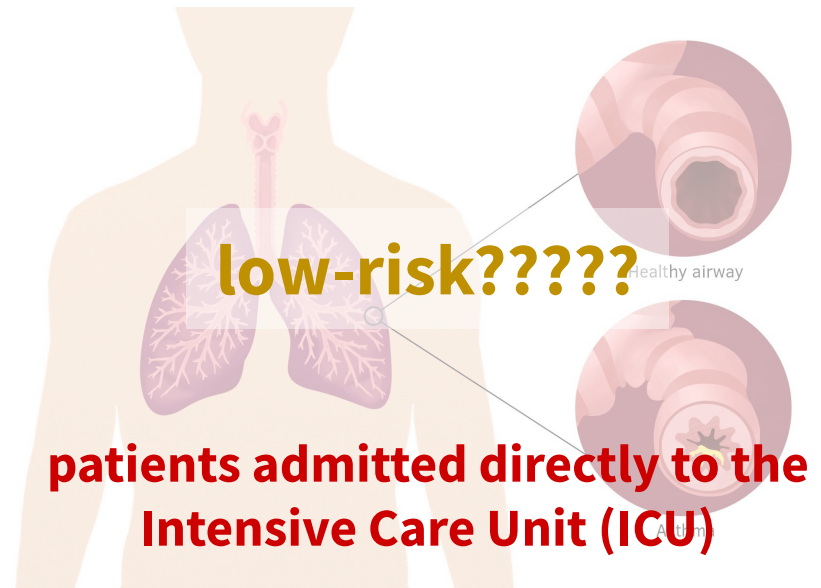


Introduction

Diagnose pneumonia: high-risk vs. low-risk



Asthma patients



European Regulations and Standards



High-Risk AI Systems

The **AI Act** identifies certain AI systems as high-risk, such as those used in healthcare, transportation, and law enforcement. **High-risk** AI systems are **subject to** stringent regulatory requirements, including **transparency and auditing**

General Data Protection Regulation (GDPR)

European Union regulation which includes the "**Right to Explanation**" granting individuals the right to an explanation of automated decision. In addition, **organizations handling personal data** must ensure that AI systems comply with **data protection principles**, including **data minimization** and **purpose limitation**

Benefits xAI



- Enhanced **accountability** and **responsible** AI development
- **Improved decision-making** in critical applications
- Increased user **trust** and **acceptance** of AI technologies
- Compliance with **legal** and **regulatory requirements**
- A more **ethical** and **transparent** approach to AI

Challenges in XAI



XAI faces several challenges in bridging the gap between complex AI models and human understanding. Explaining deep learning and complex models can be particularly challenging due to their opacity.

- Balancing model **accuracy** and **interpretability** is an ongoing challenge.
- Computation **complexity**, data size, and model dimensionality present hurdles in XAI.
- Ensuring **fairness**, **avoiding bias**, and maintaining **accountability** are **ethical** challenges in XAI
- Adherence to **data protection and privacy regulations**, like GDPR, requires transparent AI models
- Making XAI tools **accessible** and **user-friendly** is critical for practical adoption

Domain-specific Interpretations



XAI should provide domain-specific explanations for various industries and applications

Healthcare

- In healthcare, XAI should explain medical diagnoses in a way that's meaningful to clinicians and patients.

Finance

- In finance, interpretability is critical for risk assessments, and compliance with regulations like Basel III.

Autonomous Vehicles

- For autonomous vehicles, XAI should provide insights into driving decisions to ensure safety.

xAI Key Points



Explainability

Making AI more accessible and accountable



Transparency

Providing access to the inner workings of the model, so that humans can see what happens inside the "black box."



Interpretability

Making the outputs of AI systems comprehensible, providing explanations and insights into how and why a specific decision or prediction was made

White and Black box



Models can be categorized as either "white-box" or "black-box." This classification **relates to the transparency and interpretability of the models.**

White-box Models

High transparency and interpretability. They are designed to be easily understood by humans.

Examples: Linear regression, decision trees, and logistic regression are common examples of white-box models

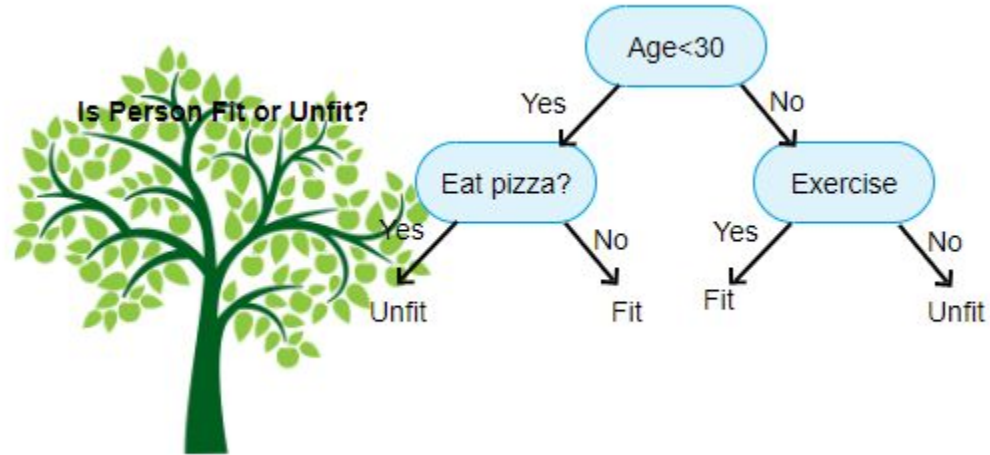
Black-box Models

Are **complex and opaque**, making it **challenging to understand** how they arrive at their predictions. They **prioritize predictive** performance over interpretability.

Examples: Deep neural networks, random forests, and gradient-boosted trees are typical examples of black-box models

Examples of white-box

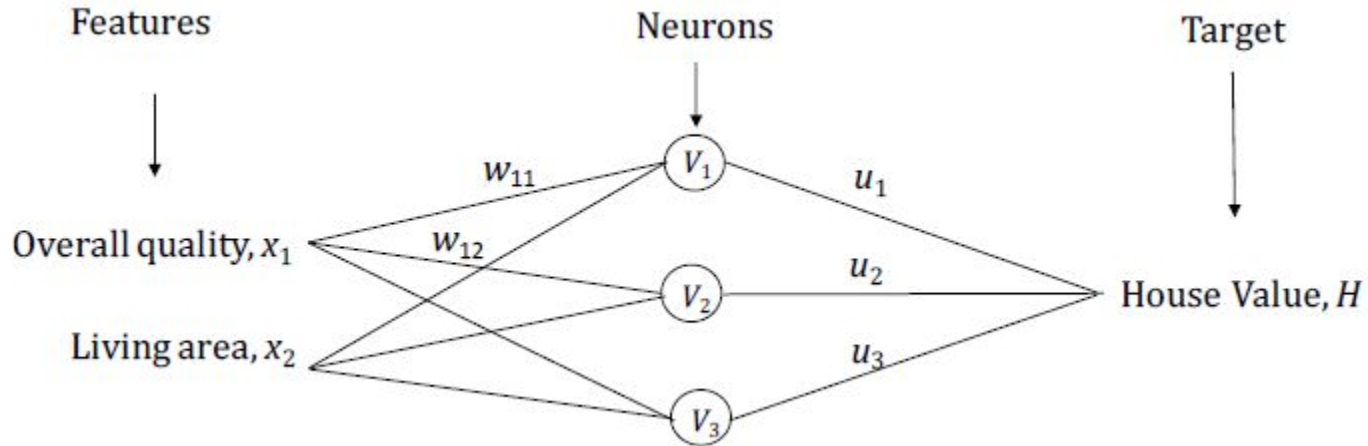
- Decision trees



Examples of white-box

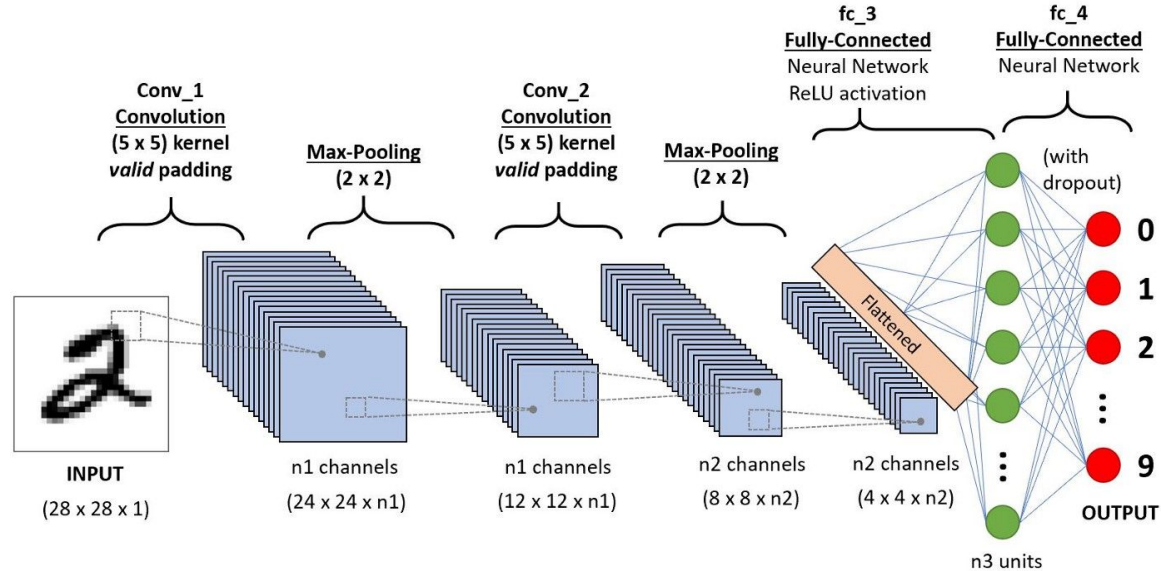


- Linear models: linear regression



Example of black-box

- Convolutional neural networks



White and Black box



Do we need to stop using black-box models?

White and Black box



The Trade-off

- A trade-off exists between model transparency and predictive performance.
- White-box models are more interpretable but may sacrifice predictive accuracy.
- Black-box models often offer superior predictive power but may lack transparency.
- The choice between white-box and black-box models depends on the specific task, domain, and the importance of interpretability.

What if we use black-boxes?



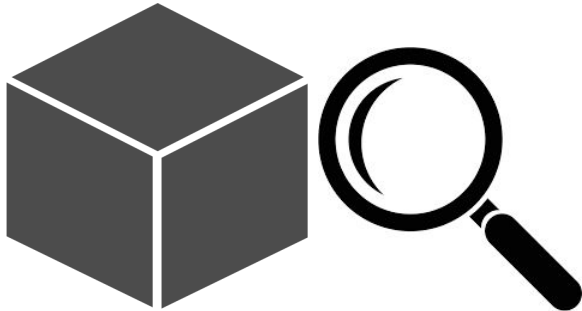
Agnostic and model-specific explanations

How can we explain models?

Agnostic explanations

More general techniques that can be applied to different already trained models.

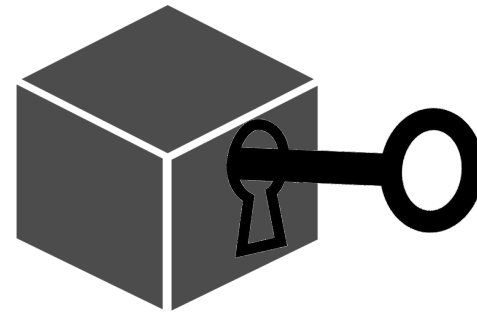
Examples: features importance, example-based explanations



Model-specific explanations

Techniques adapted to specific models that are generally planned during architectural design phase.

Examples: attention mechanisms, generative models



xAI techniques



- Feature importance
- Example-based
- Counterfactuals
- Surrogate models
- Concepts

Feature Importance

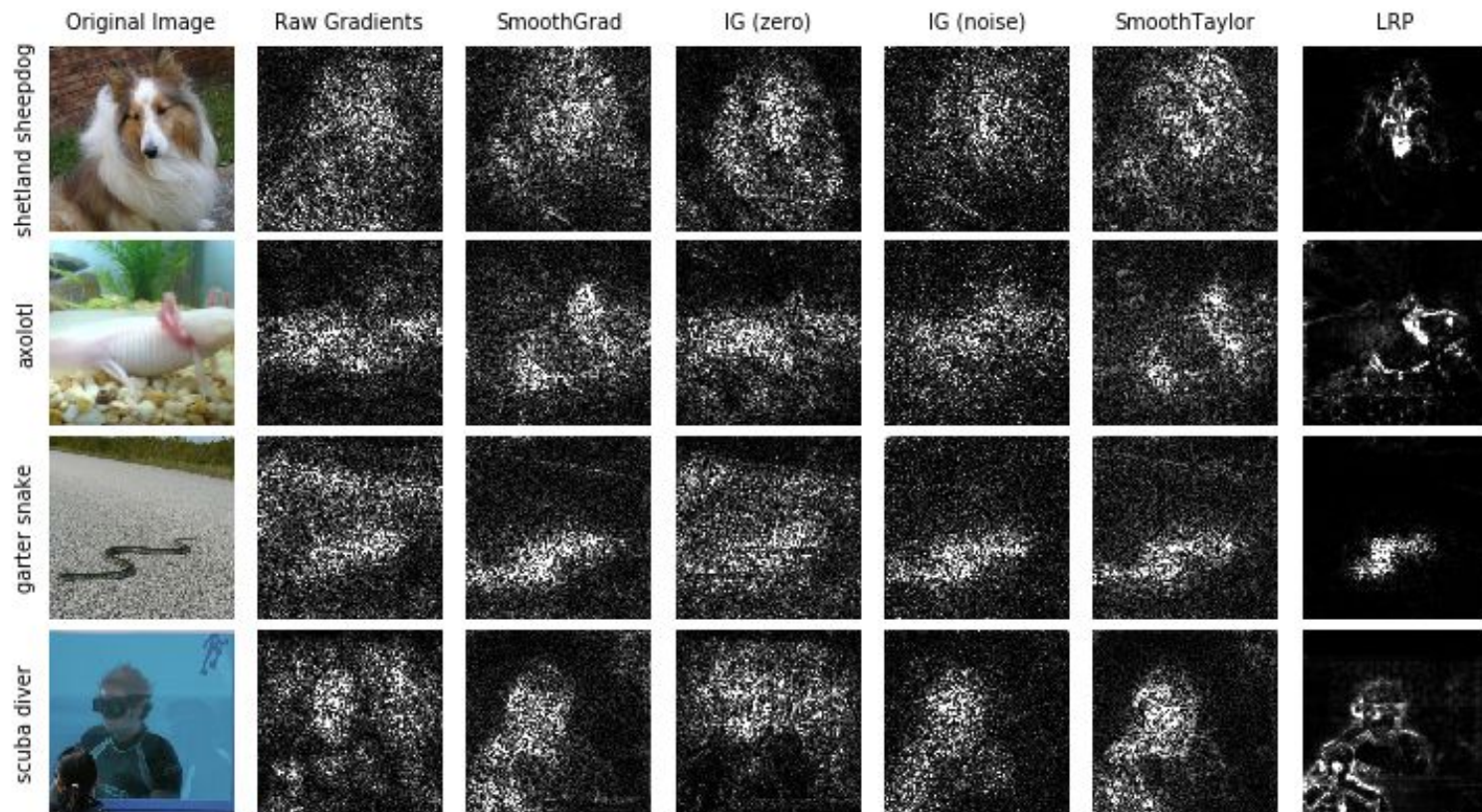


Refers to the assessment of the significance of input variables (features) in predicting the target variable.

Why Feature Importance?

- Understanding which features are influential helps us:
 - Select the most relevant features for model building.
 - Gain insights into the underlying data and problem domain.
 - Identify potential factors driving predictions or decisions.

Feature Importance



Feature Importance



Techniques for Assessing Feature Importance

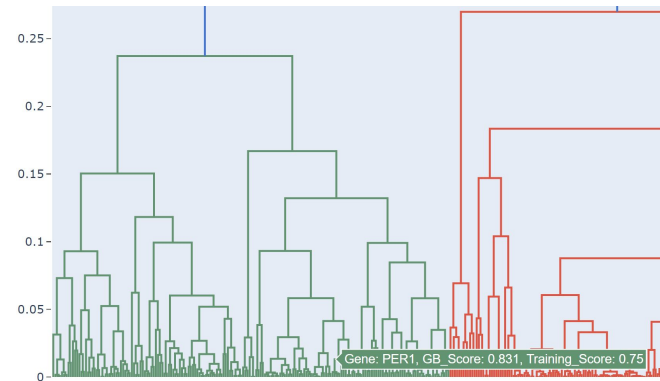
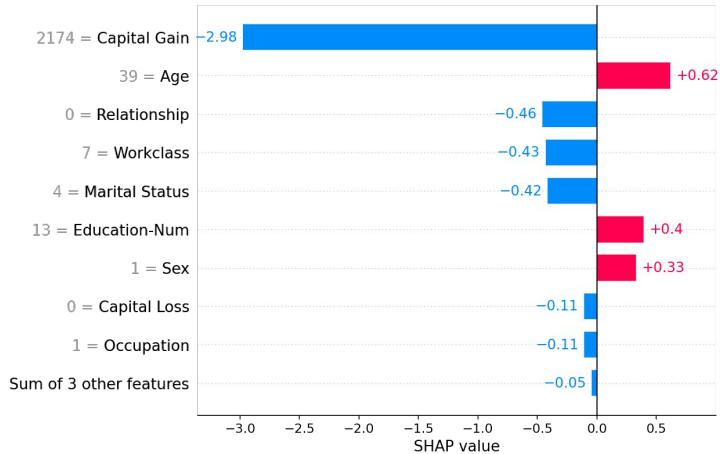
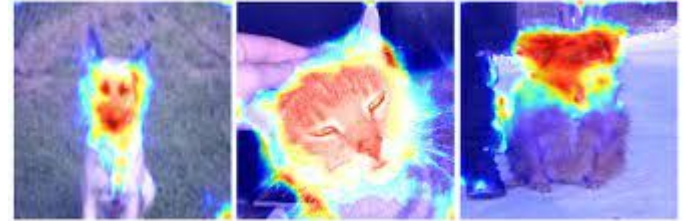
- Various techniques are used to measure feature importance, including:
 - Tree-Based Methods: Decision trees, random forests, and gradient boosting algorithms provide feature importance scores based on the number of splits they make on each feature.
 - Permutation Feature Importance: This method evaluates a model's performance when feature values are randomly shuffled. Features with the most significant drops in performance are deemed more important.
 - LASSO (L1 Regularization): LASSO regression assigns a coefficient to each feature. Features with non-zero coefficients are considered important.
 - Recursive Feature Elimination (RFE): RFE recursively removes the least important features and evaluates the model's performance. The remaining features are considered important.

Feature Importance



Visualizing Feature Importance

- Feature importance scores can be visualized through:
 - Bar plots, showing the importance of each feature.
 - Heatmaps, displaying correlations between features and their importance.
 - Decision tree or dendrogram plots, revealing the hierarchy of feature importance.

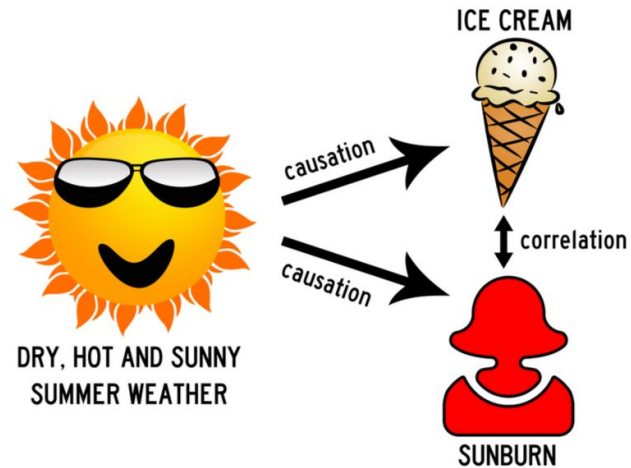


Feature Importance



Interpretation

- Feature importance doesn't always indicate causation but highlights associations.



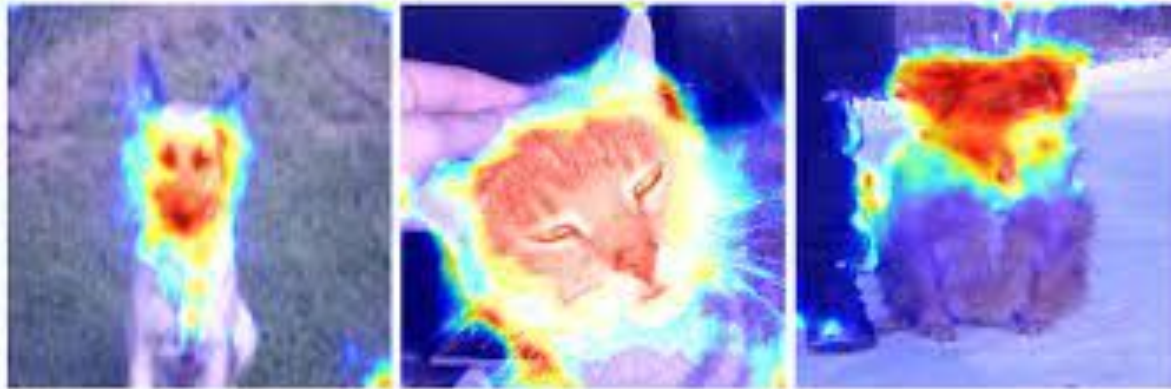
- It aids in focusing on relevant variables during analysis and decision-making.

Feature Importance



Interpretation

- Feature importance doesn't always indicate causation but highlights associations.



- It aids in focusing on relevant variables during analysis and decision-making.

Feature Importance for CNNs

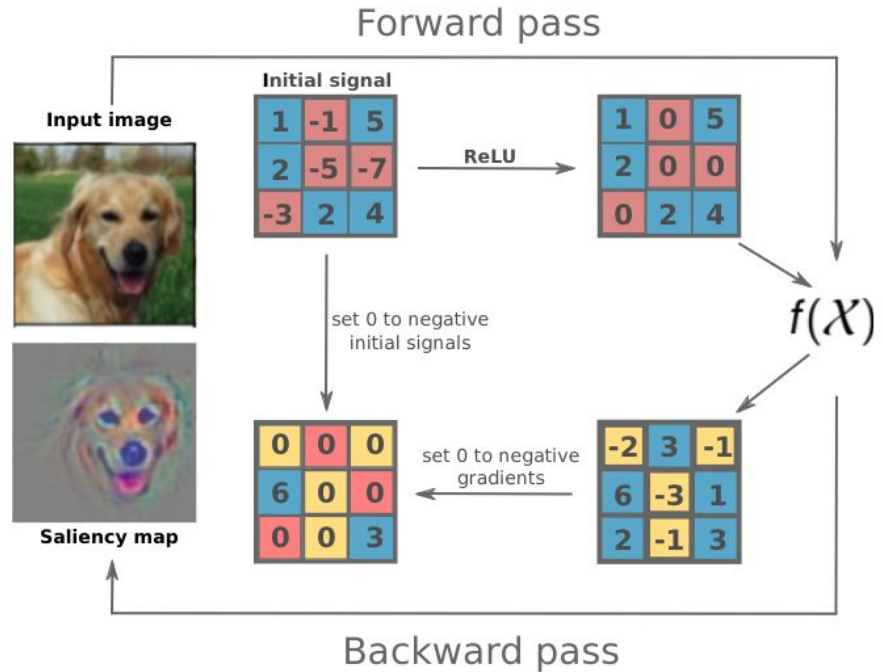


- Gradient-based methods
- CAM methods
- SHAP

Guided-Backpropagation

Given an input image X , Guided Backpropagation:

- backpropagates the gradients from the output ($f(X)$) to the input (X) resetting the negative gradients in order to find most influential features
- it also resets positions where the initial signal (before ReLU) was negative, avoiding noise due to unimportant features



Integrated Gradients



Considering a sample $X \in \mathbb{R}^n$ composed by features $x_i \ i = 0, 1, \dots, n$ and a neutral sample (baseline) $X' \in \mathbb{R}^n$ composed by features $x'_i \ i = 0, 1, \dots, n$, we want to compute the impact in the output (gradient) of changing from baseline X' to X



$$IG_i(X) ::= (x_i - x'_i) \times \int_{\alpha=0}^1 \frac{\delta F(X' + \alpha(X - X'))}{\delta x_i} d\alpha$$

At the end, each attribution is weighted according to the amount of variation from feature x'_i to x_i

Integrated Gradients



Input

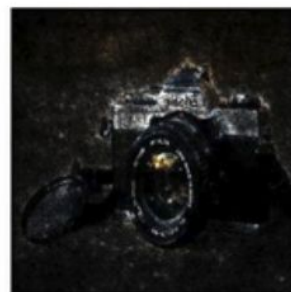


Class: Reflex Camera

Score prediction: 0.99

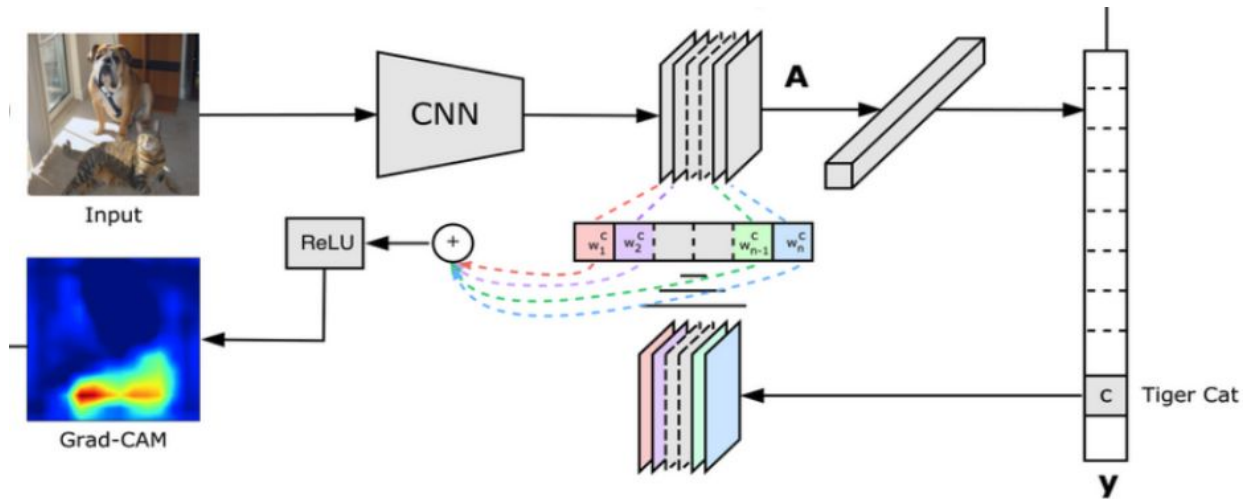
└───┬───
 └───>
Highlighting lens

Gradients



Integrated
Gradients

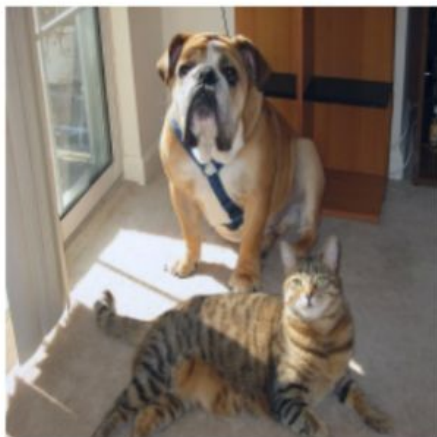
Gradient Class Activation Mapping - GradCAM



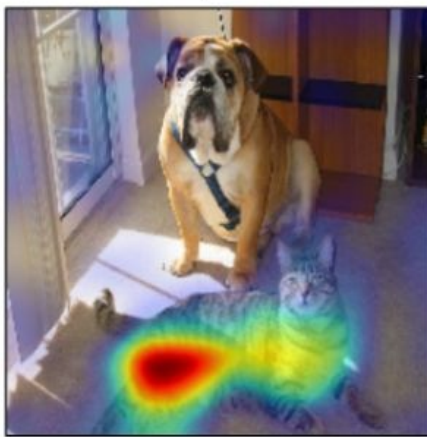
Gradient Class Activation Mapping - GradCAM

Proposed in order to combine the class awareness from Grad-CAM and the high-resolution (pixel attributions) from Guided-Backpropagation

Heatmaps obtained by Grad-CAM
(overlaid on the original image)



Original image



Class **cat**



Class **dog**

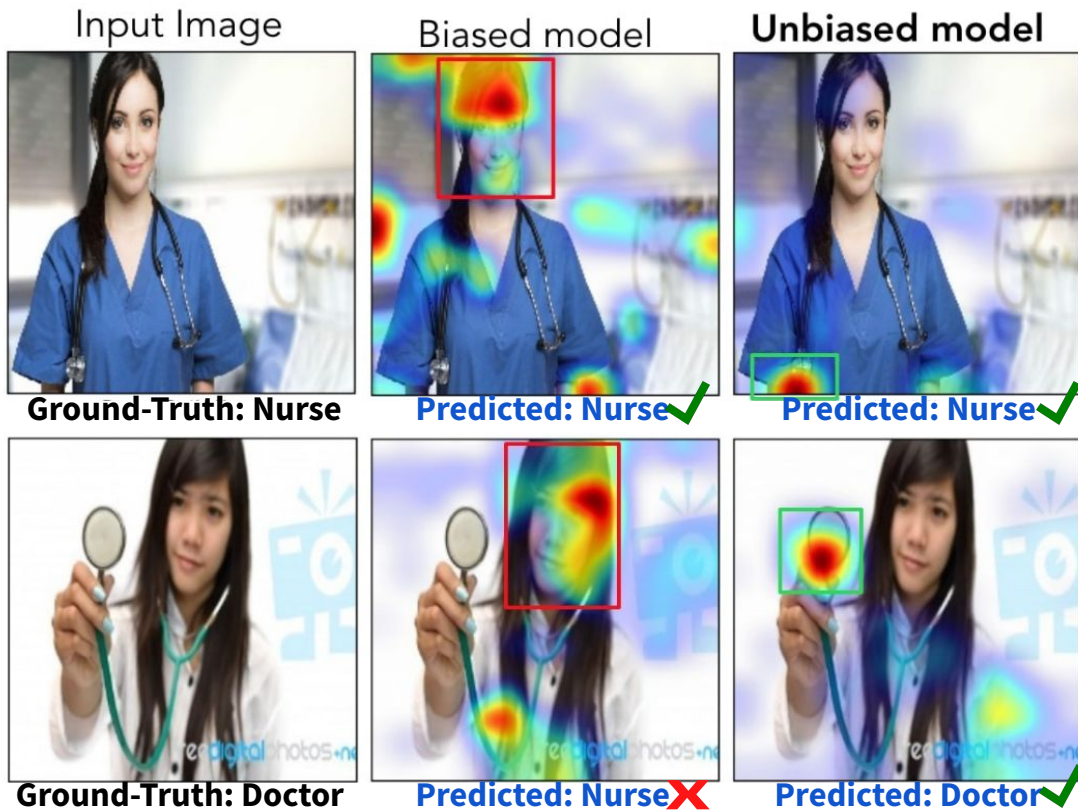
GradCAM - example bias detection



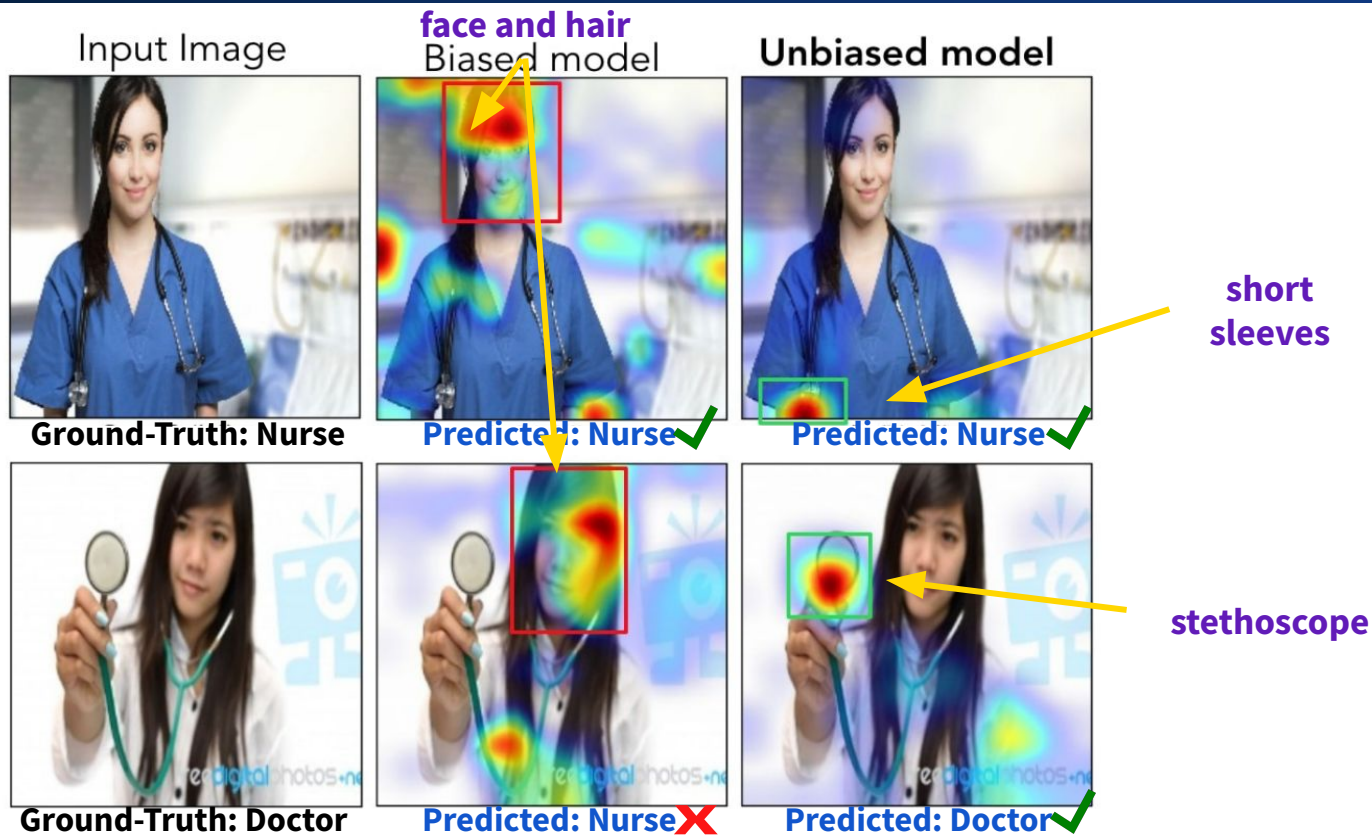
GradCAM - example bias detection



GradCAM - example bias detection

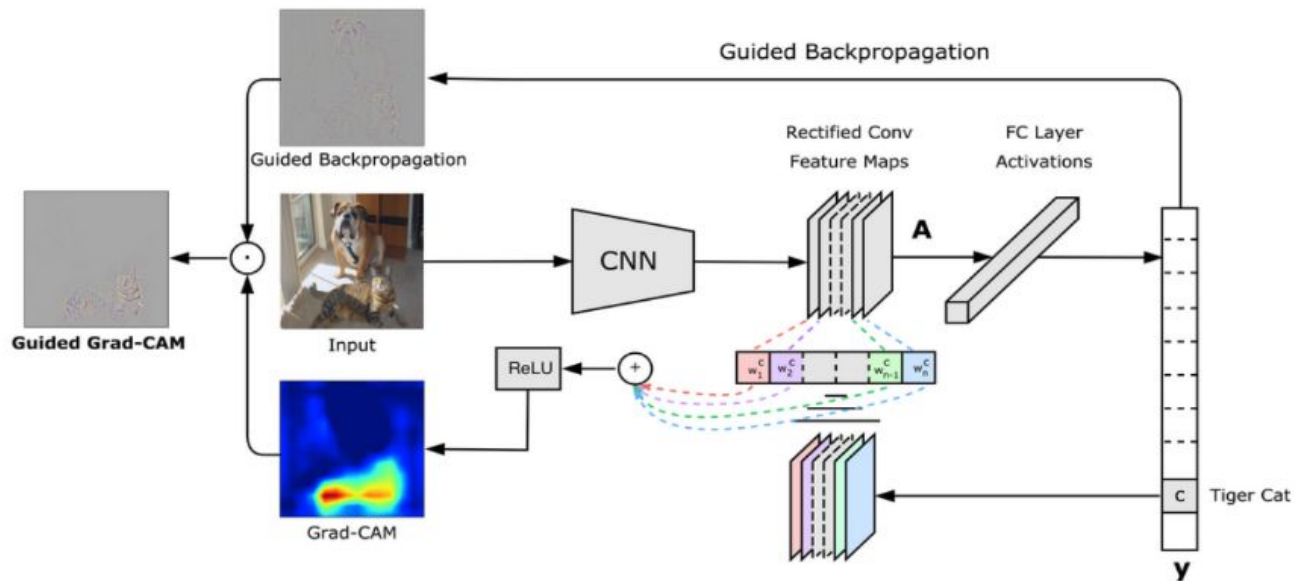


GradCAM - example bias detection



Guided - GradCAM

Proposed in order to combine the class awareness from Grad-CAM and the high-resolution (pixel attributions) from Guided-Backpropagation



SHAP



What is SHAP?

- SHAP stands for "SHapley Additive exPlanations," an advanced technique in Explainable Artificial Intelligence (XAI).
- It provides comprehensive and model-agnostic explanations for AI model predictions, offering insights into the contributions of individual features.

Motivation for SHAP

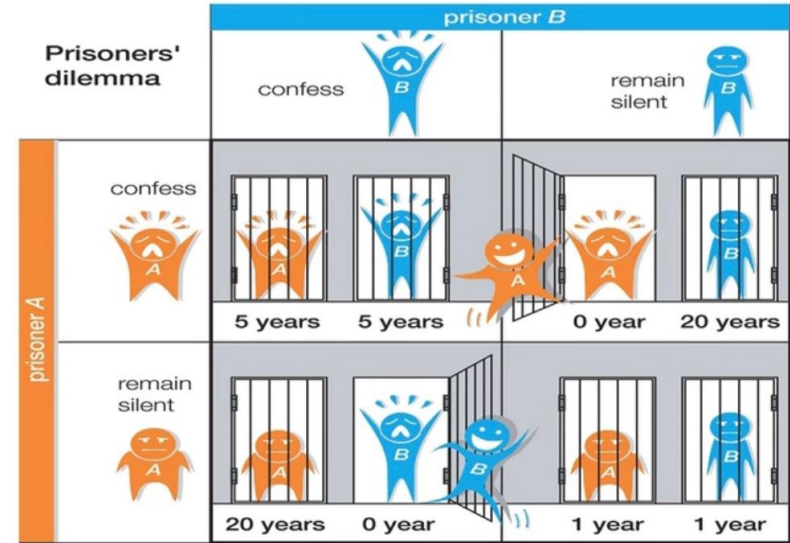
- SHAP addresses the need for a unified, theoretically sound framework for explaining the output of any machine learning model.
- It is rooted in cooperative game theory, specifically Shapley values, to ensure fairness and consistency in attributing feature importance.

SHAP



How SHAP Works

- SHAP quantifies the contribution of each feature to a model's prediction by considering **all possible feature subsets**.
- It calculates Shapley values for each feature, representing their **individual importance** in the prediction.
- These Shapley values can be positive or negative, indicating whether a feature **positively or negatively influences** the output.



Market Realist^Q

Source: Encyclopedia Britannica

SHAP



Coalitions $\xrightarrow{h_x(z')}$ Feature values

Instance x

Age	Weight	Color
1	1	1

Age	Weight	Color
0.5	20	Blue

Instance with
"absent"
features

Age	Weight	Color
1	0	0

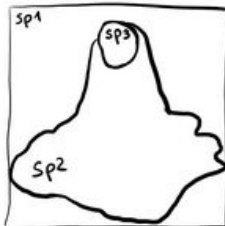
Age	Weight	Color
0.5	20	Blue
	↓	↓
	17	Pink

SHAP



Coalitions of super pixels $\xrightarrow{h_x(z)}$ Image

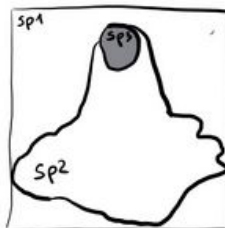
Instance x



sp1	sp2	sp3
1	1	1



Instance x
with absent
features



sp1	sp2	sp3
1	1	0





Benefits of SHAP

- SHAP offers a unified, consistent, and theoretically grounded framework for feature attribution.
- It helps ensure fairness in model predictions by revealing the impact of different features transparently.

Limitations of SHAP

- The computational complexity of SHAP can be a challenge for large datasets and complex models.
- Interpretability of SHAP values may require domain expertise to fully understand their implications.

Example-based



These methods use examples to explain the general knowledge of the model

- Prototypes and criticism
- Deep dream

Example-based



Prototypes



Criticisms



Prototypes



Criticisms



Example-based



Counterfactual



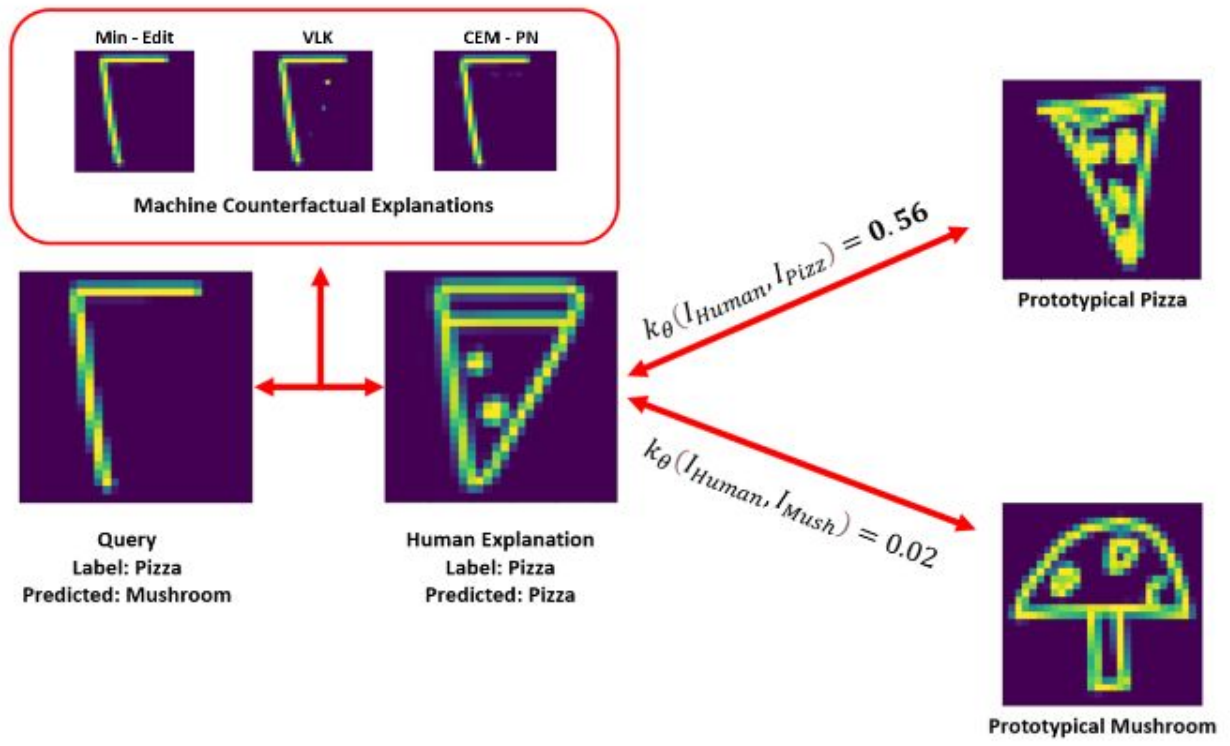
What Are Counterfactual Explanations?

- Counterfactual explanations provide an alternative input or set of inputs that, when applied to an AI model, would have resulted in a different prediction.
- These explanations help users understand the factors that influenced the model's decision.

How Counterfactual Explanations Work

- Counterfactuals are generated by altering one or more input features while keeping other variables constant.
- The modified input is fed into the AI model, and the change in prediction is assessed.

Counterfactual



Counterfactual



Challenges in Generating Counterfactuals

- Creating counterfactual explanations can be challenging due to the need to find feasible input changes that result in a different prediction.
- The generation process may require optimization algorithms and domain knowledge.

Surrogate models



These are simpler models created to carry the same behavior of the original one.

They are approximations.

- LIME



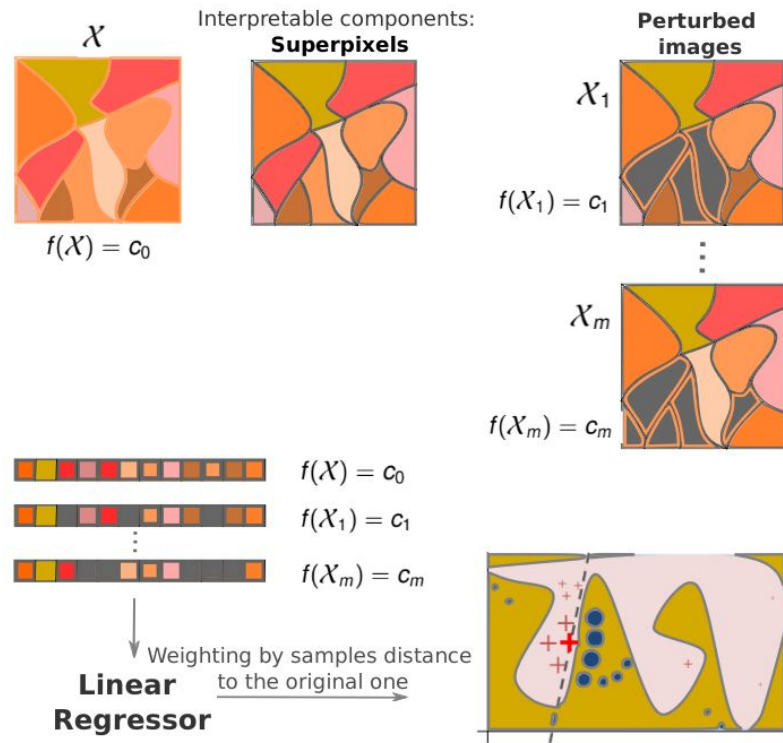
What is LIME?

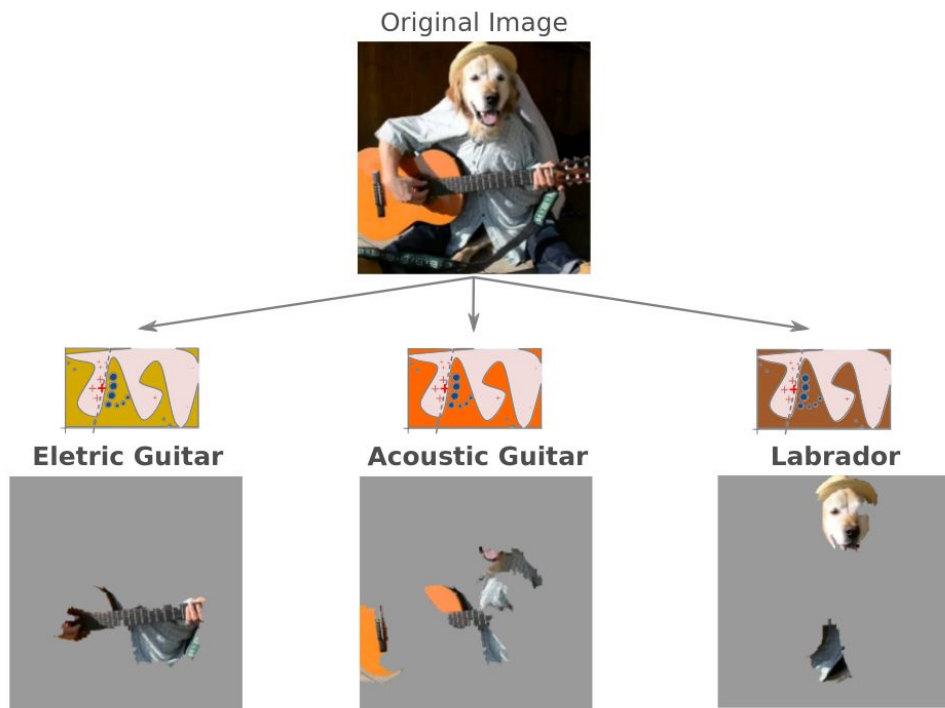
- LIME stands for "Local Interpretable Model-Agnostic Explanations."
- It's a popular and powerful technique in XAI that focuses on providing local explanations for individual AI model predictions.

Local Interpretability

- LIME provides local interpretability by explaining why a specific prediction was made for a particular instance.
- It highlights which features played a crucial role in the model's decision for that specific case.

- We translate X (a sample we want to explain) into interpretable components (superpixels for images)
- We generate perturbed samples X_1, \dots, X_m by including or excluding these components
- We train a linear regressor based on the original model response for the analyzed class ($f(X)$), to locally approximate the curve





Most important superpixels for the analyzed classes



Benefits of LIME

- LIME is a versatile tool that can be applied to a wide range of machine learning models, including deep neural networks.
- It enhances transparency and accountability, making AI systems more interpretable and trustworthy.

Limitations of LIME

- LIME's local explanations might not always generalize well to the overall model behavior.
- The choice of the surrogate model and the method of generating perturbed examples can impact the quality of explanations.

Local Explanations



Local Explanations

- Local explanations in XAI focus on explaining **individual predictions** or decisions made by an AI model.
- These explanations are **specific** to a **single instance** or data point and help understand why the model arrived at a particular output for that case.

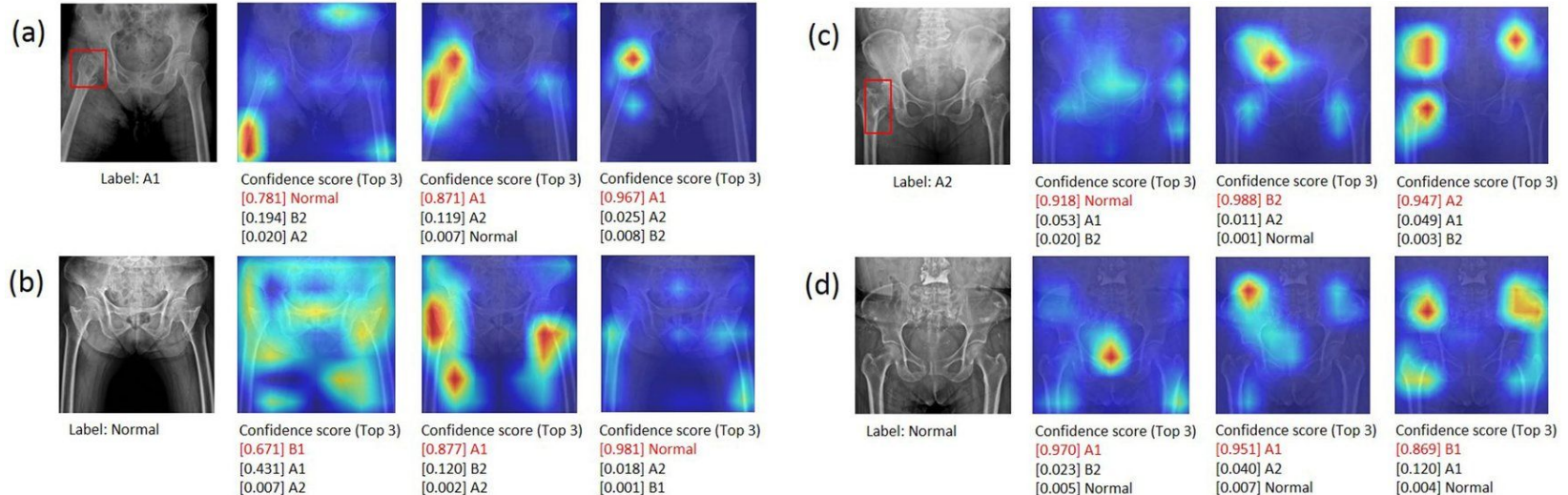
Techniques for Local Explanations

- Local explanation techniques aim to provide **insights into the features or attributes** that had the most influence on a particular prediction.
- Common techniques include LIME (Local Interpretable Model-Agnostic Explanations) and SHAP (SHapley Additive exPlanations), which build simplified models around specific instances to explain the model's behavior locally.

Use Cases for Local Explanations



- Local explanations are valuable when we need to understand the reasoning behind a **specific AI model's prediction**, especially in applications where outcomes can have significant consequences.
- Examples include understanding why a medical AI system diagnosed a patient with a particular condition, or why a credit scoring model rejected a loan application.



Global Explanations



Global Explanations

- Global explanations, on the other hand, aim to provide an **overall understanding of an AI model's** behavior across a dataset or a larger context.
- These explanations help identify **general trends, biases, and feature importance** across all model predictions.
- They can reveal **systemic issues or patterns** that might not be evident when focusing on individual instances.

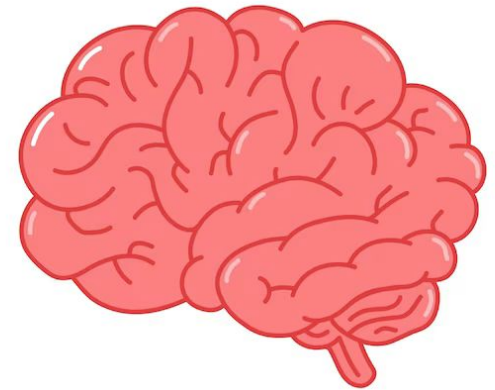
Techniques for Global Explanations

- Techniques for global explanations often involve methods like feature importance analysis, aggregating local explanations, and creating global surrogate models.
- These methods provide insights into which features or attributes are most influential in the model's decision-making across the entire dataset.

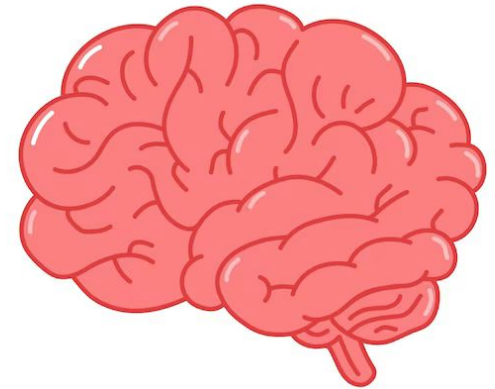
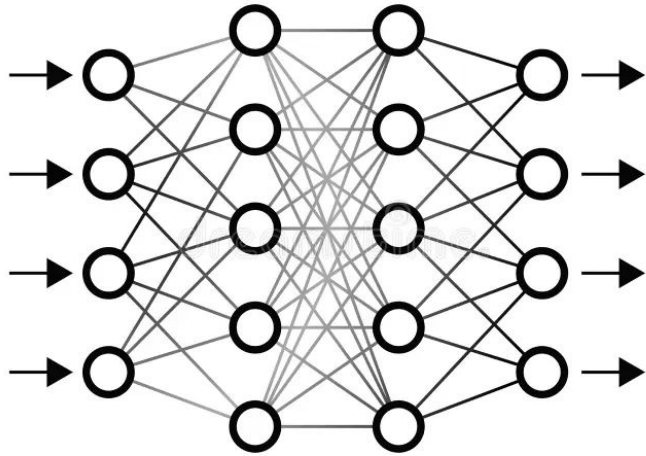
Concepts



Concepts



Concepts



Concepts



- **Semantics:** the branch of linguistics and logic concerned with meaning.



Concepts



- **Semantics:** the branch of linguistics and logic concerned with meaning.



= Money.

Concepts

- **Semiotics:** the study of signs and symbols and their use or interpretation.



Concepts

- **Semiotics:** the study of signs and symbols and their use or interpretation.

All money?



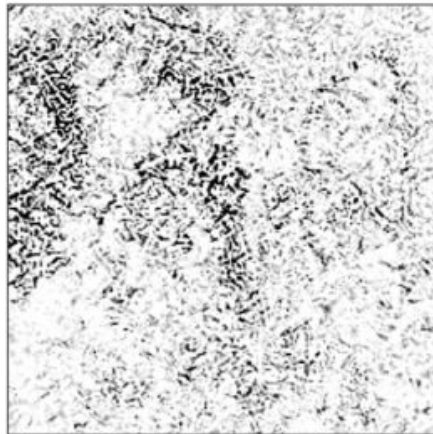
Concepts



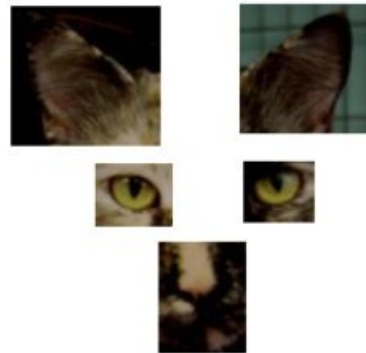
a.



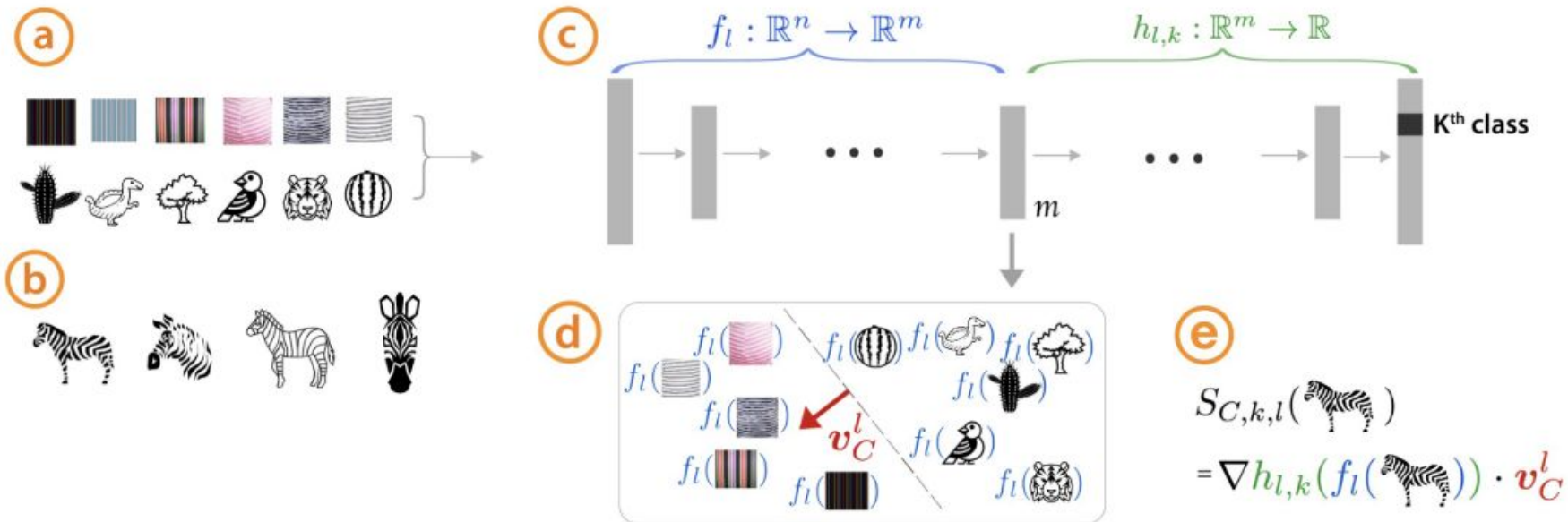
b.



c.

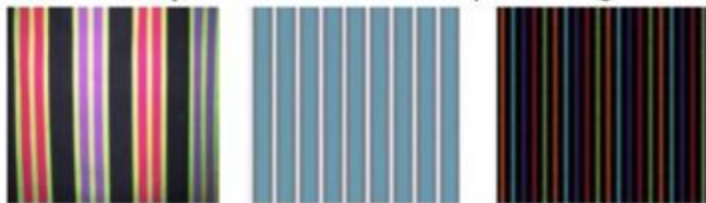


Testing with Concept Activation Vectors - TCAV



Testing with Concept Activation Vectors - TCAV

CEO concept: most similar striped images



CEO concept: least similar striped images



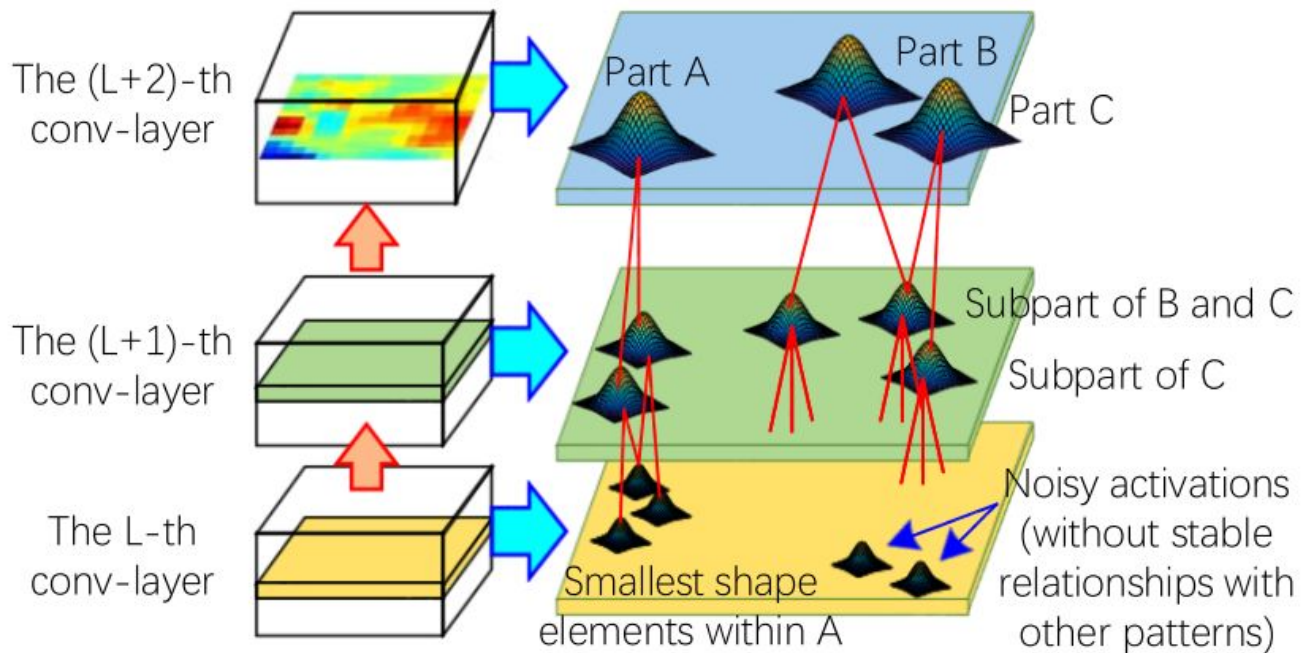
Model Women concept: most similar necktie images



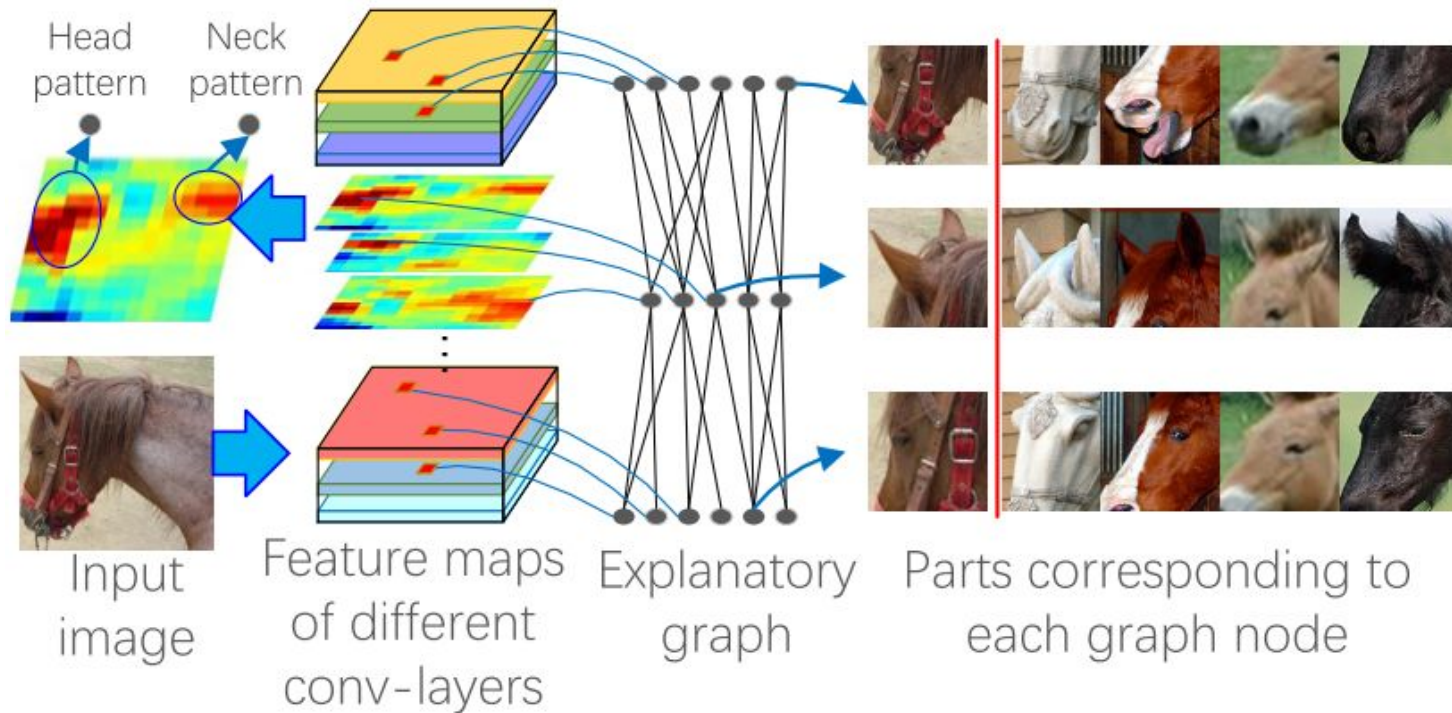
Model Women concept: least similar necktie images



Explanatory graphs



Explanatory graphs



Local vs Global



Key Takeaways:

- Local explanations focus on **explaining individual model predictions** and are valuable for specific use cases.
- Global explanations provide an overarching view of **model behavior** and are crucial for assessing fairness and bias.
- Striking a **balance** between these two types of explanations is essential for a comprehensive XAI strategy.

Trade offs between Explanations



Explanations	Local	Global	Agnostic	Model-specific
Strengths	<p>Precise insights for a specific prediction</p> <p>Understanding individual model behavior</p>	<p>Insights into feature importance and overall model behavior</p> <p>Identify systemic issues, biases, or patterns in the model</p>	<p>Techniques not tied to a particular model structure</p> <p>Can be applied to different types of models</p>	<p>Techniques adapted to specific contexts</p> <p>Can provide deeper explanations</p>
Trade-offs	<p>May not capture global patterns and trends</p> <p>Lack the broader context of model behavior</p>	<p>Might not provide insights into individual predictions</p> <p>Less precise for explaining specific instances</p>	<p>Can provide too general explanations, not adapted to the problems</p>	<p>Are tied to a particular model structure</p> <p>In some cases, need to be applied in the beginning of architectural design</p>

Tools for xAI

Pytorch Captum

Attribution

- [Integrated Gradients](#)
- [Saliency](#)
- [DeepLift](#)
- [DeepLiftShap](#)
- [GradientShap](#)
- [Input X Gradient](#)
- [Guided Backprop](#)
- [Guided GradCAM](#)
- [Deconvolution](#)
- [Feature Ablation](#)
- [Occlusion](#)
- [Feature Permutation](#)
- [Shapley Value Sampling](#)
- [Lime](#)
- [KernelShap](#)

NoiseTunnel

Layer Attribution

Neuron Attribution

Metrics

Utilities

Base Classes

Insights API Reference

Insights

Features

CAPTUM INSIGHTS

Instance Attribution

[Direct Target](#)[Export](#)

Filter by Classes

Animal and 2 other classes are selected. [Edit](#)

Filter by Instances

Instance Type:

Integrated Gradients

Approximation steps: [Fetch](#)

Predicted

[Deer \(0.553\)](#)[Plane \(0.166\)](#)[Bird \(0.129\)](#)[Cat \(0.044\)](#)

Label

Contribution

Photo (Image)



Original

Gradient Overlay

Predicted

[Truck \(1.000\)](#)[Car \(0.000\)](#)

Label

Contribution

Photo (Image)




Original

Gradient Overlay

Tools for xAI

Tensorflow - Lucid













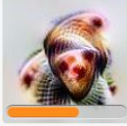


INPUT IMAGE



OUTPUT CLASSES

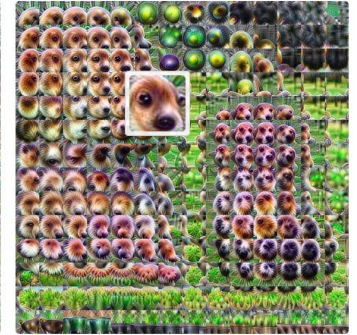
- Labrador Retriever
- Golden Retriever
- Tennis Ball
- Rhodesian Ridge...
- Appenzeller

TOP CHANNELS SUPPORTING LABRADOR RETRIEVER

MIXED3B	MIXED4A	MIXED4B	MIXED4C	MIXED4D
				
				
				
...
Showing 3 of 480	Showing 3 of 508	Showing 3 of 512	Showing 3 of 512	Showing 3 of 528



MIXED3A



MIXED4A

Tools for xAI



Microscope

 Microscope

MODELS ABOUT

Models

The OpenAI Microscope is a collection of visualizations of every significant layer and neuron of eight important vision models.

[LEARN MORE >](#)

AlexNet

A landmark in computer vision, this 2012 winner of ImageNet has over 50,000 citations.



28 nodes

AlexNet (Places)

The same architecture as the classic AlexNet model, but trained on the Places365 dataset.



28 nodes

Inception v1

Also known as GoogLeNet, this network set the state of the art in ImageNet classification in 2014.



83 nodes

Inception v1 (Places)

The same architecture as the classic Inception v1 model, but trained on the Places365 dataset.



83 nodes

VGG 19

Introduced in 2014, this network is simpler than Inception variants, using only 3x3 convolutions and no branches.



20 nodes

Inception v3

Released in 2015, this iteration of the Inception architecture improved performance and efficiency.



127 nodes

Inception v4

Released in 2016, this is the fourth iteration of the Inception architecture, focusing on uniformity.



190 nodes

ResNet v2 50

ResNets use skip connections to enable stronger gradients in much deeper networks. This variant has 50 layers.



94 nodes