UNIVERSITÀ PARTHENOPE
DiST
Dipartimento di Scienze e Tecnologie

Machine Learning (part II)

Decision Making
by
eXplainable Artificial Intelligence
in
Computer Vision

Angelo Ciaramella

Introduction

- Artificial intelligence (AI) and algorithmic decision making are having a profound impact on our daily lives

- high-stakes applications
  - healthcare, business, government, education, justice, …
  - become essential to make these systems safe, reliable, and trustworthy

Trustworthy AI

- Attention from the government and different scientific communities
  - European Union (EU)
    - ethical guidelines for trustworthy AI to govern and facilitate the development and working of AI systems
  - General Data Protection Regulation (GDPR)
    - "right to explanations" for AI decisions
  - National Institute of Standards and Technology (NIST)
    - framework to measure and increase user trust in AI systems

Trustworthy AI

- Attention from the government and different scientific communities
  - U.S. Government Accountability Office (GAO)
    - framework for the accountability and responsible use of AI
  - Defense Advanced Research Project Agency (DARPA)
    - launched a program known as Explainable Artificial Intelligence (XAI)

Trustworthy AI

- AI systems can fail and show dangerous consequences for humans
  - COMPAS algorithm
    - used across the nation to predict the risk of criminal recidivism
    - is biased against black people
  - Facial recognition software
    - tagged black people with inappropriate labels because of the low quality of sample data used to train the system

Trustworthy AI

- Resume screening
  - used by a major tech company
  - biased against women

- Self-driving car
  - killed a pedestrian on the road when its algorithm malfunctioned
  - did not respond when its sensors detected a pedestrian in the way

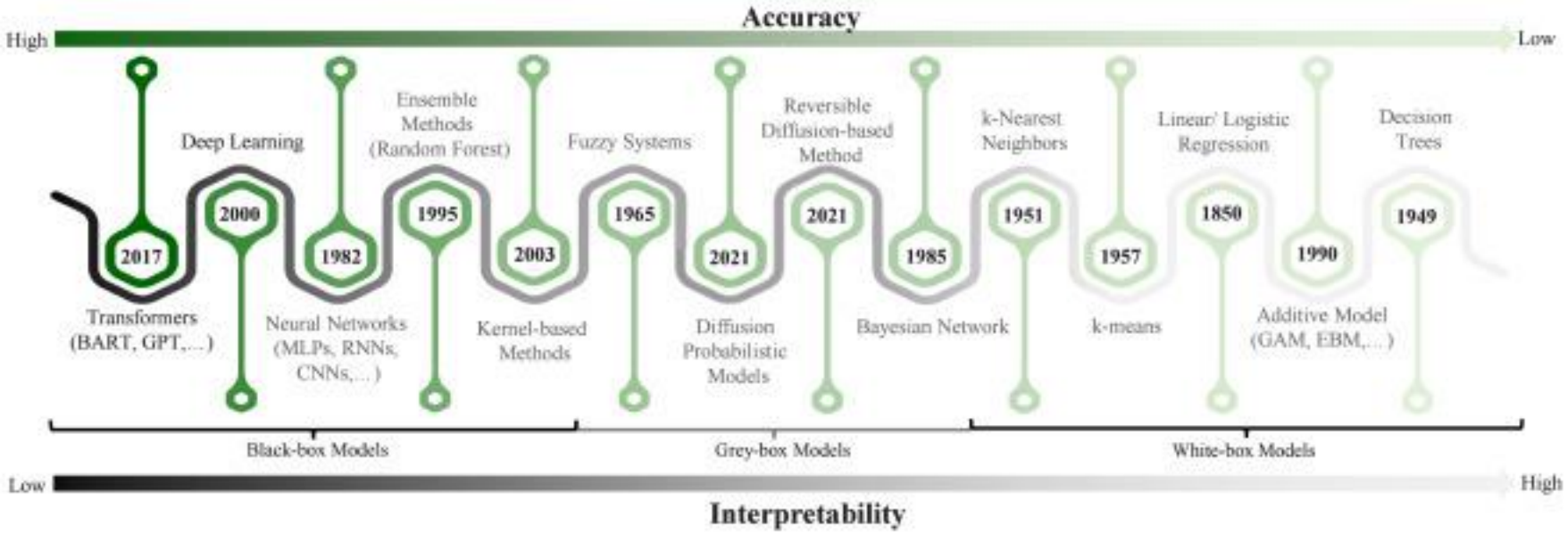
EU guidelines

- EU presented three guidelines for AI
  - lawful
    - AI system's development, deployment, and use should follow all the applicable laws and regulations
  - ethical
    - AI system should respect the ethical principles and guidelines of humans
  - robust
    - AI system should be technically robust while being ethical and lawful

EU guidelines

Relations among tustworthy based concepts

Accuracy and interpretability



Balance between accuracy and interpretability

Trustworthy AI requirements

- Main trustworthy AI requirements
  - Fairness
    - free from any bias and discrimination
  - Explainability
    - understand the reasons that led to a decision
  - Accountability
    - monitoring decision-making algorithms to ensure that they do not cause any harm
  - Privacy
    - protect the privacy of the data to both avoid harmful consequences and increase the users' trust in the system
  - Acceptance
    - increase the acceptance and trust for AI-based decision-making systems by carefully evaluating the system

Bias

- Bias
  - Data
    - the data on which the system is trained is biased
    - data does not represent a clear picture of reality
  - Model
    - the algorithm itself introduces bias
    - wrong objective function that does not capture the fundamental logic for the prediction
  - Evaluation
    - wrong evaluation metrics were used to evaluate the model

Bias mitigation solutions

- pre-processing
- in-processing
- post-processing

Bias mitigation solutions

- pre-processing
  - pre-processing the data to make it free from any bias and discrimination
  - Examples
    - bias mitigation method for word embedding
      - approximates the effect of removing a small sample of training data based on the bias of the resulting system
    - radioactive data labeling
      - labels the training dataset images with an identifiable mark to ensure biased data traceability

Bias mitigation solutions

- in-processing
  - mitigate bias by modifying the decision making algorithms
  - Example
    - adversarial learning
      - uses the concept of maximizing predictor accuracy while minimizing the ability to predict protected attributes

Bias mitigation solutions

- post-processing
  - mitigating bias by using the output of the predictors through post-processing
  - Example
    - removes bias by adjusting the learned predictor to balance among supervised learning methods

Explainability

- decision making systems
  - essential for different stakeholders involved with these systems to understand the reasons that led to a decision
  - communicate the reasoning for the AI system's decisions to different stakeholders
  - helps the system designers detect unknown vulnerabilities and correct errors and policymakers to design better laws to govern the system

eXplainable Artificial Intelligence

- eXplainable Artificial Intelligence (XAI) techniques are aimed at producing AI models with a good interpretability-accuracy
  - building white/gray-box ML models
    - they are interpretable by design (at least at some degree) while achieving high accuracy
  - improving the explainability of black-box models
    - endowing black-box models with a minimum level of interpretability when white/gray-box models are not able to achieve an admissible level of accuracy

Explainability

Grad-Cam application example (https://github.com/jacobgil/pytorch-grad-cam?tab=readme-ov-file)

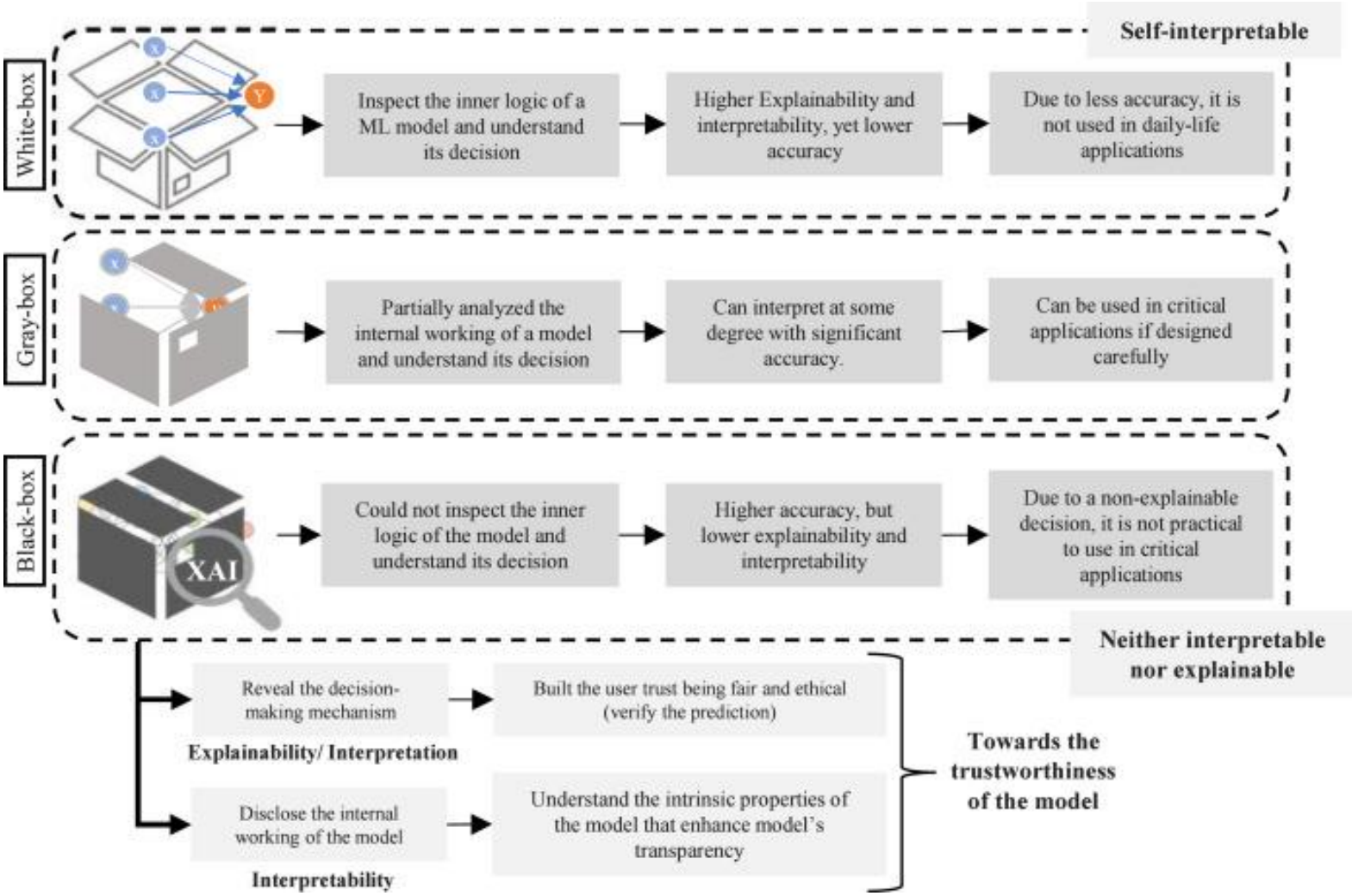
Deep Neural Networks

- There are two terminologies by which we can try to elucidate a Deep Neural Networks (DNN) model
  - interpretability
    - interface that gives additional information or explanations that are essential for interpreting an AI system's underlying functioning
  - explainability
    - insight into the DNN's decision to the end-user in order to build trust that the AI is making correct and non-biased decisions based on facts

Explainability

- Categories
  - ex-ante
    - explanations is to establish the initial trust in the system
  - ex-post
    - explaining the features and circumstances that lead to a particular decision

Explainability



A comparison of white-box, gray-box, and black-box models.

XAI approches

- pre-modeling approches
  - transparency and explainability by explaining the datasets on which the system is trained
  - Examples
    - visualization techniques to better understand the data before using it
    - dataset standardization methods
      - labeling and data sheet creation will facilitate communication and understanding between different entities using them

XAI approches

- In-modeling approches
  - making interpretable models
- Examples
  - decision trees, linear models, rule-based models
  - graph structure of trees
    - internal nodes represent tests on features and leaf nodes represent class labels
  - decision/rule sets
    - association rules like an if-then rule or m-of-n rule to generate classification rules
- Disadvantage
  - they are only usable when the size of the classification rules and the dimensionality of the features are within a human-understandable range

XAI approches

- Post-modeling approches
  - building proxy models on top of black-box/complex models
  - Categories
    - feature importance
    - example based
    - rule-based
    - visualization-based

Feature importance

- Aim
  - explainability by assigning feature importance values to the input variables
    - which features played a more critical role in the decision-making process

- Examples
  - Local Interpretable Model-agnostic Explanations (LIME)
    - highlighting essential features that led to the decision
  - Layerwise Relevance Propagation (LRP)
    - for image classification algorithms, which includes interpretability by computing every pixel's contribution to the prediction made by the classifier (heat maps)
  - Automated Concept-based Explanation (ACE)
  - SHapely Additive exPlanations (SHAP)
    - interpretability by assigning feature scores to each attribute for different predictions

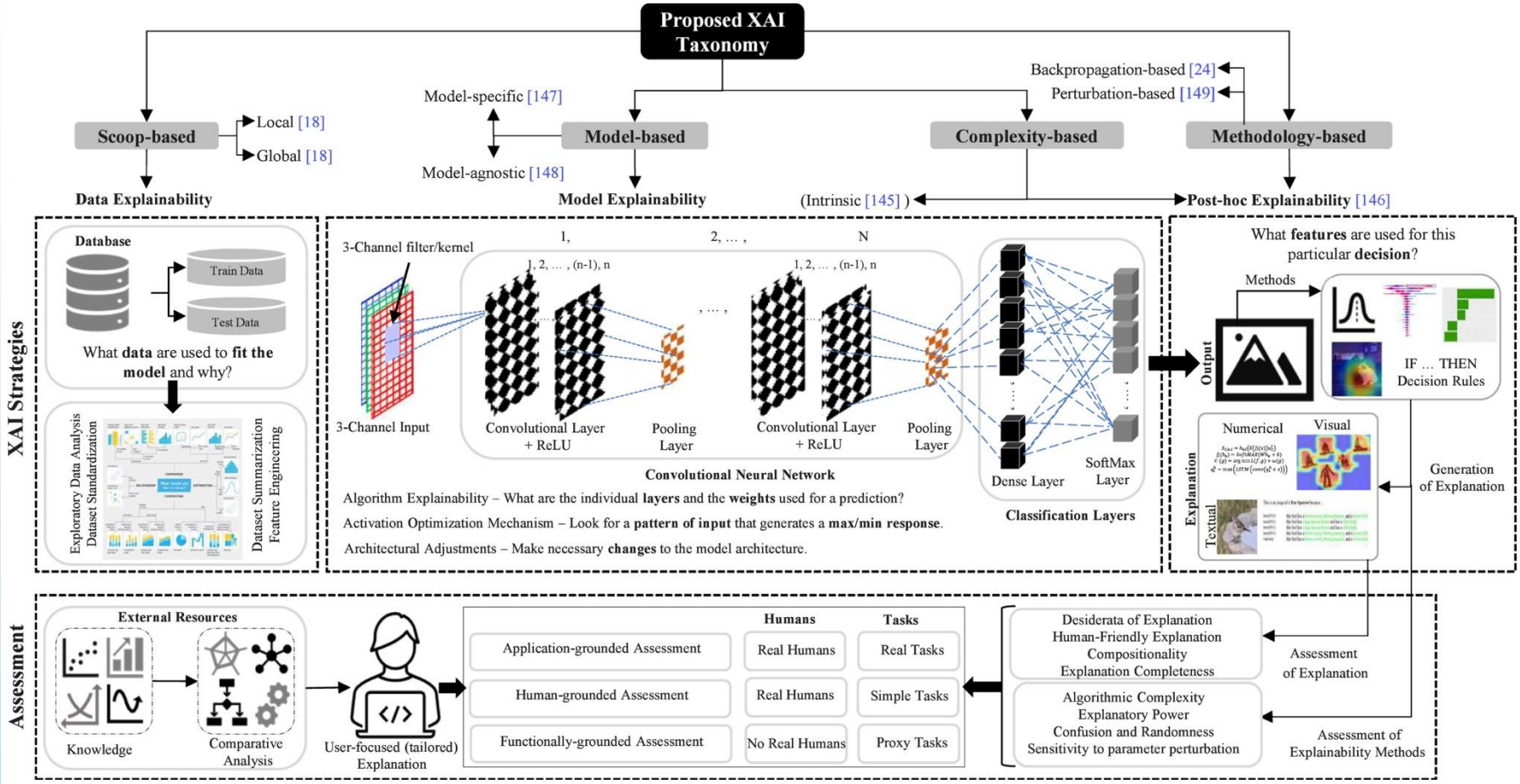
Rule based

- Aim
  - provide explainability by extracting useful information from the model
  - are usually applied to artificial neural networks
    - useful information is extracted using the hidden layers to provide interpretability
- Examples
  - Anchors
    - algorithms to extract IF-THEN rules that highlight the characteristics of an input instance that are sufficient for a classifier to make a prediction
  - model compression
  - algorithms to extract Fuzzy IF-THEN rules

Visualization based

- Aim
  - explainability by visualizing the internal working of opaque AI systems
- Examples
  - explaination by visualizing how the change in the feature importance affects the model performance
  - contribution of the evidence for the final decision to provide explainability to classifiers

XAI strategies



Four-axes XAI methodology (S. Ali et al. 2023)

XAI axes

A list of research questions that address several levels/axes of explanation.

By data explainability	By model explainability	By post-hoc explainability
D1: What sort of information do we have in the database? **D2:** What can be inferred from this data? **D3:** What are the most important portions of the data? **D4:** How is the information distributed? **D5:** Is it possible to increase the model's performance by lowering the number of dimensions? **D6:** Can a better explanation be offered by using data summarizing techniques?	**M1:** What makes a parameter, objective, or action important to the system? **M2:** When did the system examine a parameter, objective, or action, and when did the model reject it? **M3:** What are the consequences of making a different decision or adjusting a parameter? **M4:** How does the system carry out a certain action? **M5:** How do these model parameters, objectives, or actions relate to one another? **M6:** What factors does the system take into account (or disregard) when making a decision? **M7:** In order to achieve a goal/inference, which techniques does the system utilize or avoid?	**P1:** What is the reason behind the model's prediction? **P2:** What was the reason for occurrence X? What would happen if Y was the cause of occurrence X? **P3:** What variables have the most influence on the user's decision? **P4:** What if the information is altered? **P5:** To keep current results, what criteria must be met? **P6:** Is there anything that can be done to have a different outcome? **P7:** Why is it essential to make a certain conclusion or decision?

Research questions for explanation

Data explainability

- Aim
  - involves a group of techniques aimed at better comprehending the datasets used in the training and design of AI models

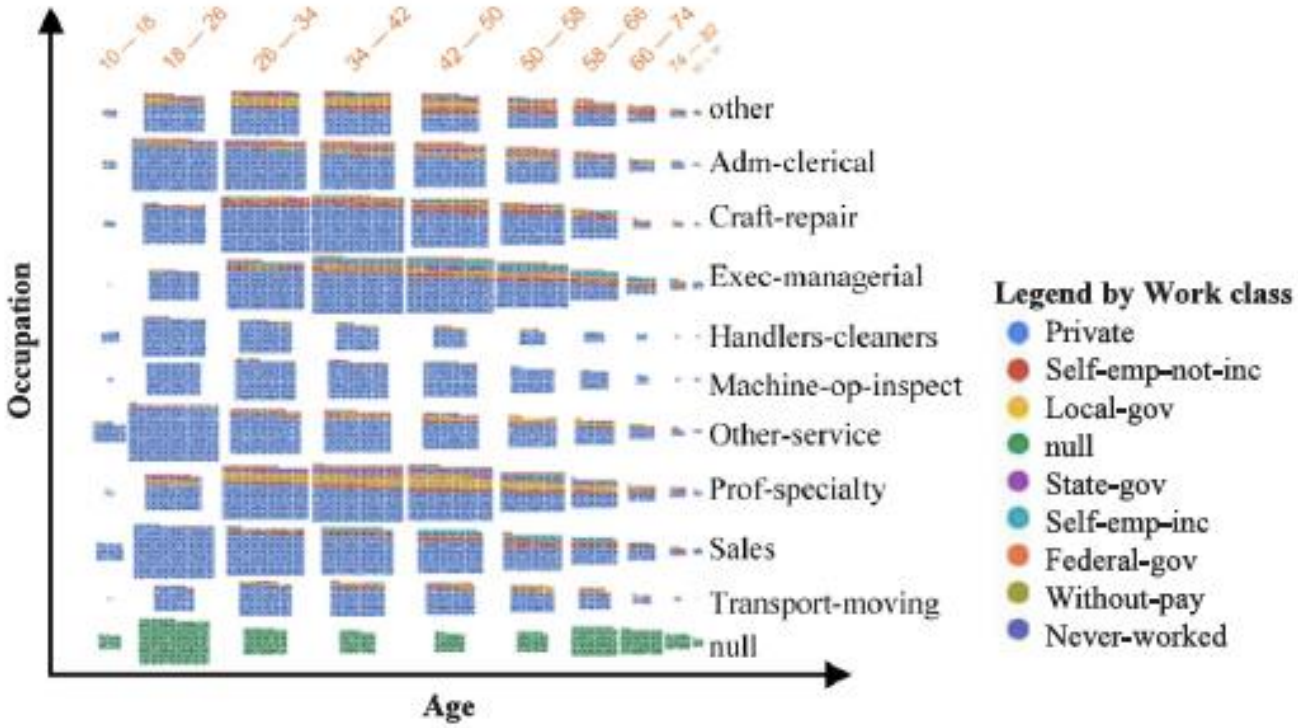
- Main aspects to consider
  - Exploratory Data Analysis (EDA)
  - explainable feature engineering
  - dataset description standardization
  - dataset summarizing methodologies
  - knowledge graphs

Exploratory Data Analysis

- Aim
  - compile a list of the most significant characteristics of a dataset
    - dimensionality, mean, standard deviation, range, and missing samples
    - e.g., Google Facets

- Example
  - UCI Census Income data
    - basic supervised binary classification task in which a model is used to distinguish whether an employee has an annual income over 50K or not

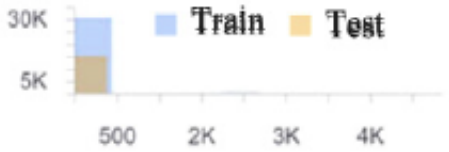
Google Facets – UCI Census



All 16282 training data points that show the relationship between one feature (Age) and another feature (Occupation), then faceting is performed in a different dimension according to a discrete feature (Work class)

Google Facets – UCI Census

	Count	Missing	Mean	Std Dev	Zeros	Min	Median	Max	Bar Chart
Capital Loss									
	32.6K	0%	87.3	402.95	95.33%	0	0	4356	
	16.3K	0%	87.9	403.09	95.31%	0	0	3770	

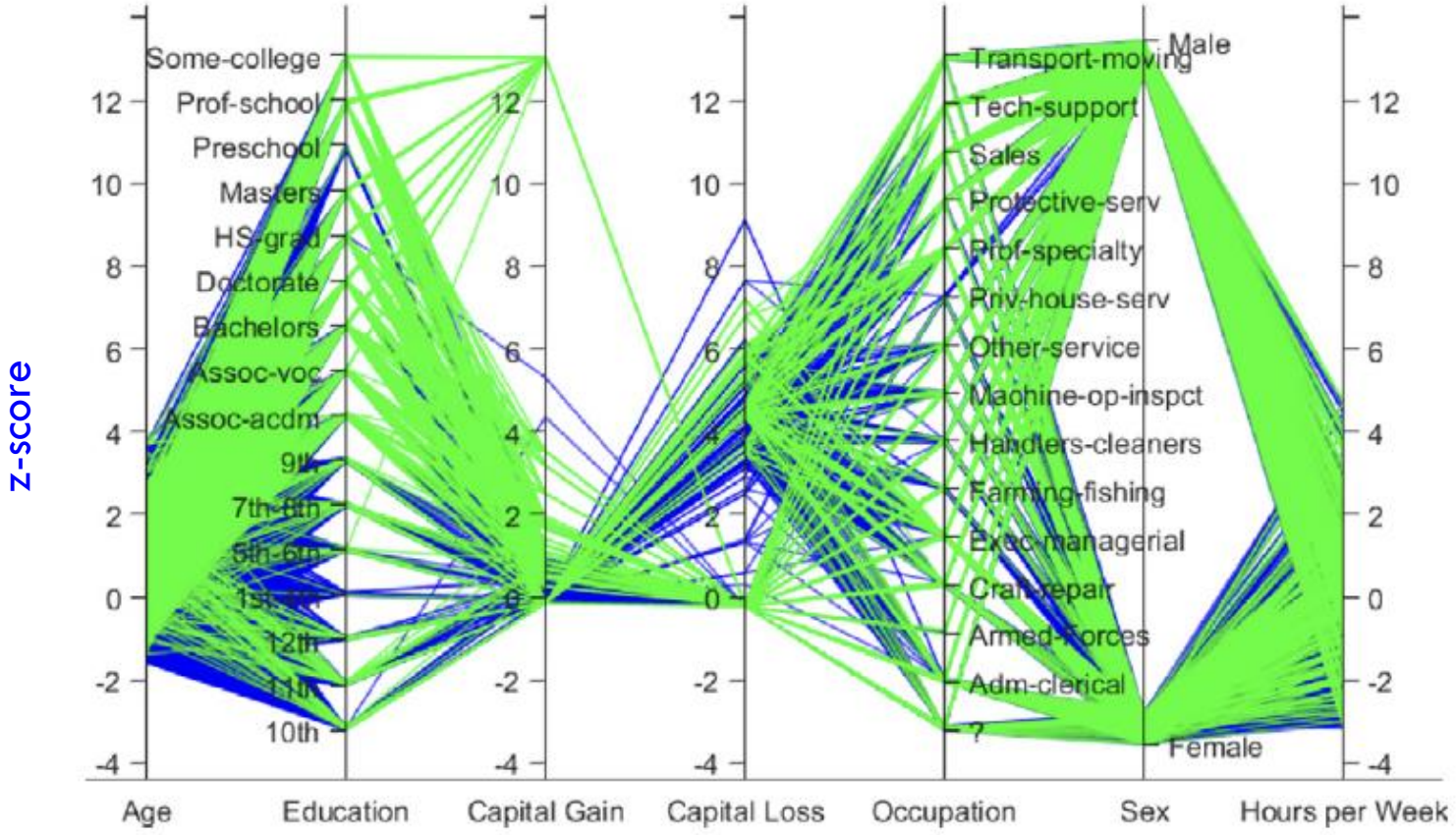


Six integer-type statistical values from the UCI Census datasets

	Count	Missing	Unique	Top	Freq Top	Avg Str Len	Bar Chart
Target							
	32.6K	0%	2	$\leq 50K$	24.7K	4.76	
	16.3K	0%	2	$\leq 50K$	12.4K	5.76	

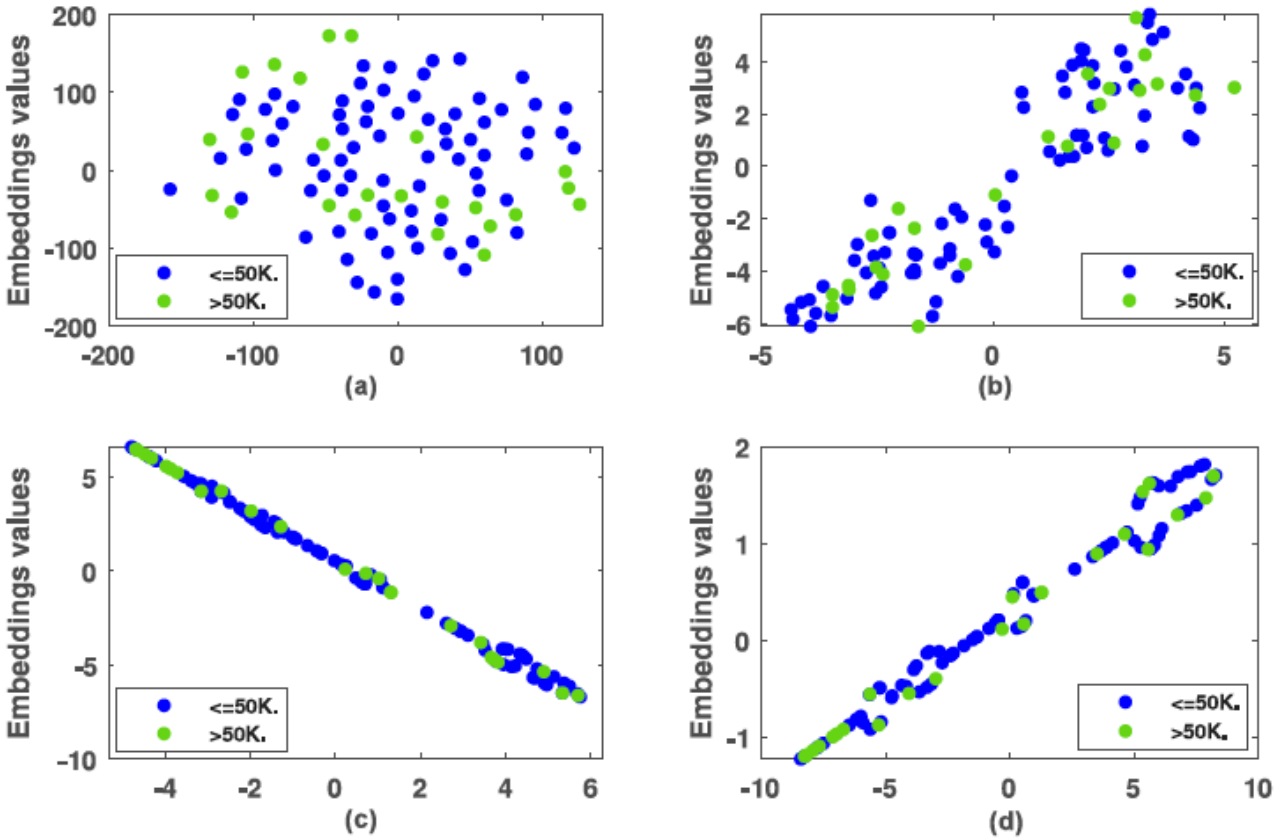
The table displays one categorical (string) type feature out of the nine features in the UCI Census dataset. A model trained and tested on such data would provide an incorrect assessment as a result of the label imbalance problem

Parallel Coordinate Plots



It can be seen in the distinct cluster, the features of age and education have a significant role in determining a given class. In the class prediction, the capital gain, on the other hand, does not create separation boundaries. Thus, this feature may be left out of the classification task. The green line represents the target value > 50K, and the blue line denotes income value < 50K.

Projections



t-SNE: Produces a graph with well-defined clusters and a small number of integer data points. To get a better separation between the clusters of the UCI Census Income dataset, several distance measures are used: (a) Mahalanobis, (b) Cosine, (c) Chebyshev, and (d) Euclidean. Further tools are Embedding Projector toolbox and Uniform Manifold Approximation and Projection

Explainable feature engineering

- Common approaches
  - domain-specific
    - rely on domain expert knowledge as well as insights gained via EDA to extract and identify significant features
  - Example
    - binary classifier on satellite images to distinguish cloudy pixels from ice/snow pixels that looked quite similar
  - model-based methods
    - use of a number of mathematical models to determine the underlying structure of a dataset
  - Example
    - Clustering, dictionary learning, disentangled representation learning

Dataset description

- Standard dataset descriptions
  - Datasheets for Datasets, Dataset Nutrition Labels, Data Declarations for Natural Language Processing (NLP)

- Example
  - dataset document that includes information on many modules
    - metadata, statistics, pair plots, the probabilistic model, provenance, and ground truth correlations

Dataset summarizing

- Aim
  - generates predictions for a given input and compares them to training samples/cases using a distance metric
- Example
  - document summarization, scene summarizing, prototype selection, data squashing

Knowledge graphs

- Aim
  - modeling entities and their relationships by means of a directed, edge-labeled graph, often organizing them in an ontological schema
- Examples
  - Doctor XAI
    - creates an agnostic XAI approach for ontology-linked data classification
  - Data Mining Ontology for Grid Programming (DAMON)
    - model for data mining approaches and existing tools
  - KD-DONTO
    - emphasizes the development of data mining techniques

Physics-informed neural network

- Aim
  - incorporating physical equations and constraints into neural networks for modeling complex and non-linear processes
- Examples
  - Earth system science, two-step process to improve the spatio-temporal resolution of turbulent flows

Model explainability

- Aim
  - select the modeling technique from a set of techniques that are deemed interpretable (white-box models)
  - phases to ensure interpretability
    - Algorithmic transparency
    - Simulatability
    - Decomposability
- Examples
  - Decision Tree, Decision Sets, Rule set, Case-based reasoning, Interpretable Fuzzy Systems, Generalized Additive Models

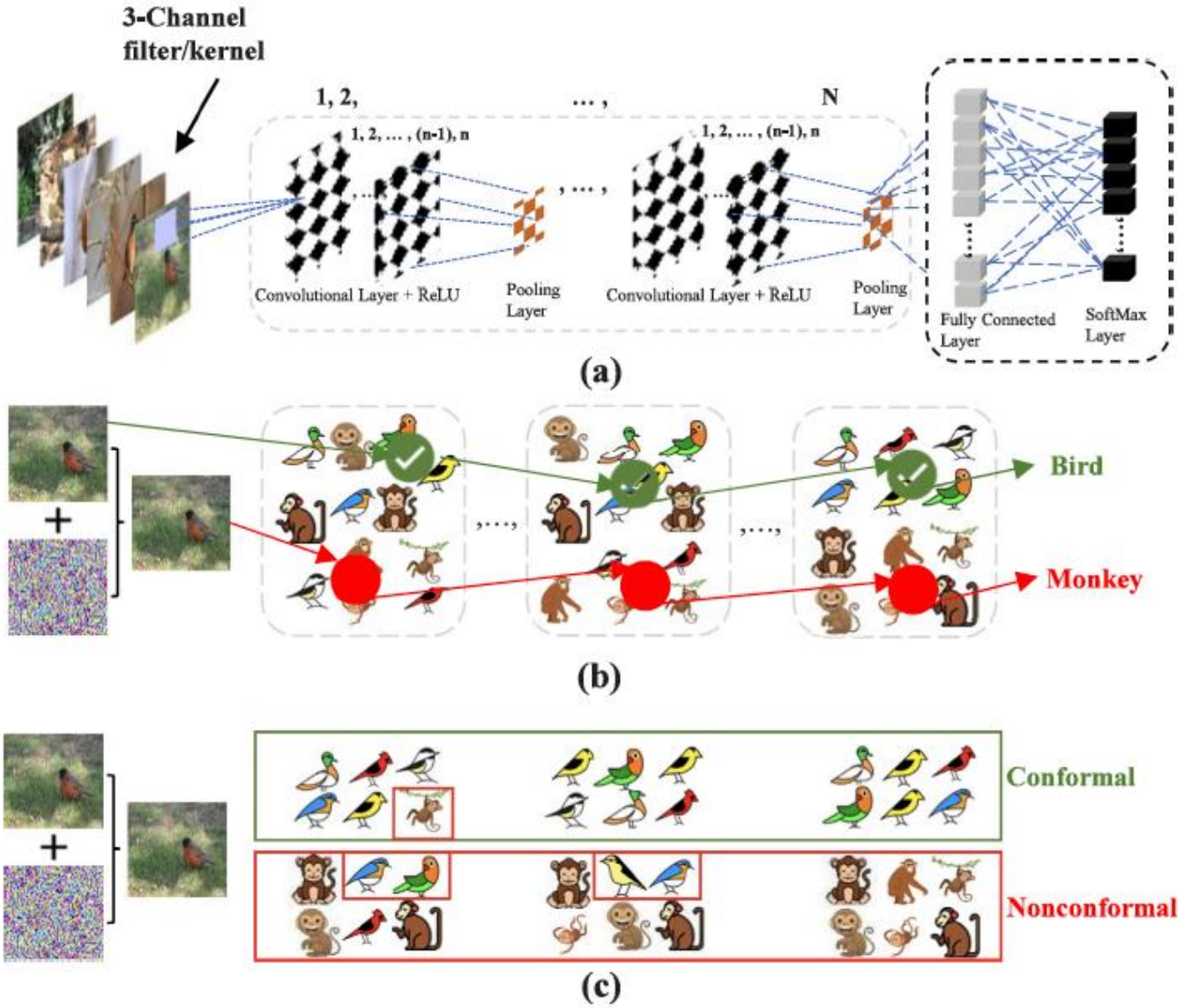
Hybrid explainable models

- Aim
  - combine an inherently interpretable modeling technique with a sophisticated black-box method
- Examples
  - Deep k-Nearest Neighbors (DkNN)
    - K-NN inference on the hidden representation of the training dataset that is learned via layers of a DNN
  - Self-Explaining Neural Networks (SENN)
    - generalize a linear classifier by utilizing NNs to learn its features, their associated coefficients, and how the networks are aggregated into a prediction

Hybrid explainable models

- Examples
  - BagNets bag-of-features model
    - the features are learned using a DNN
  - Neural-symbolic (NeSy) models
    - X-NeSyL (eXplainable Neural Symbolic Learning) via knowledge graphs
    - Finite State Automata
    - Neuro-Fuzzy models

DkNN

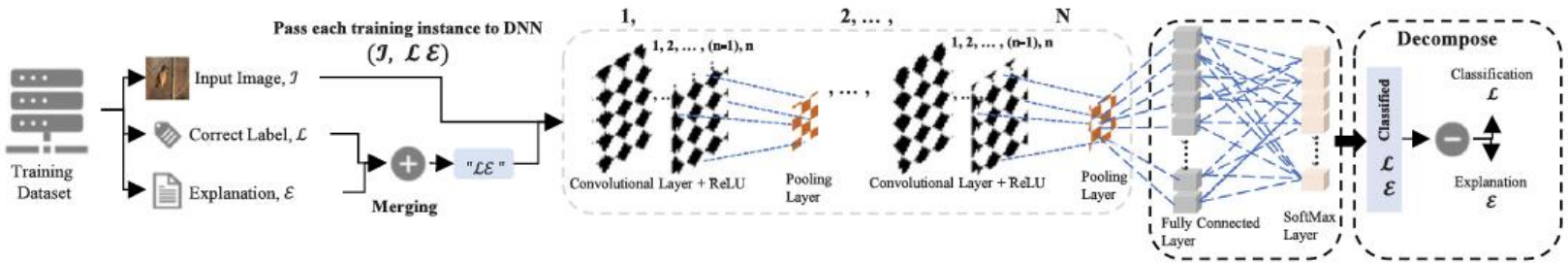


Example of DkNN

Joint prediction and explanation

- Aim
  - model may be trained to give both a prediction and an explanation
- Examples
  - Teaching Explanations for Decisions (TED) framework
    - supplement the training dataset by including a collection of features, and output, as well as the user's reasoning for that decision, which is called an explanation, in each sample
  - Rationalizing Neural Predictions (RNP) model
    - which consist of two parts (both trained simultaneously), a generator and an encoder
    - in order to make a prediction, the generator uses the distribution of input text segments as potential explanation

TED

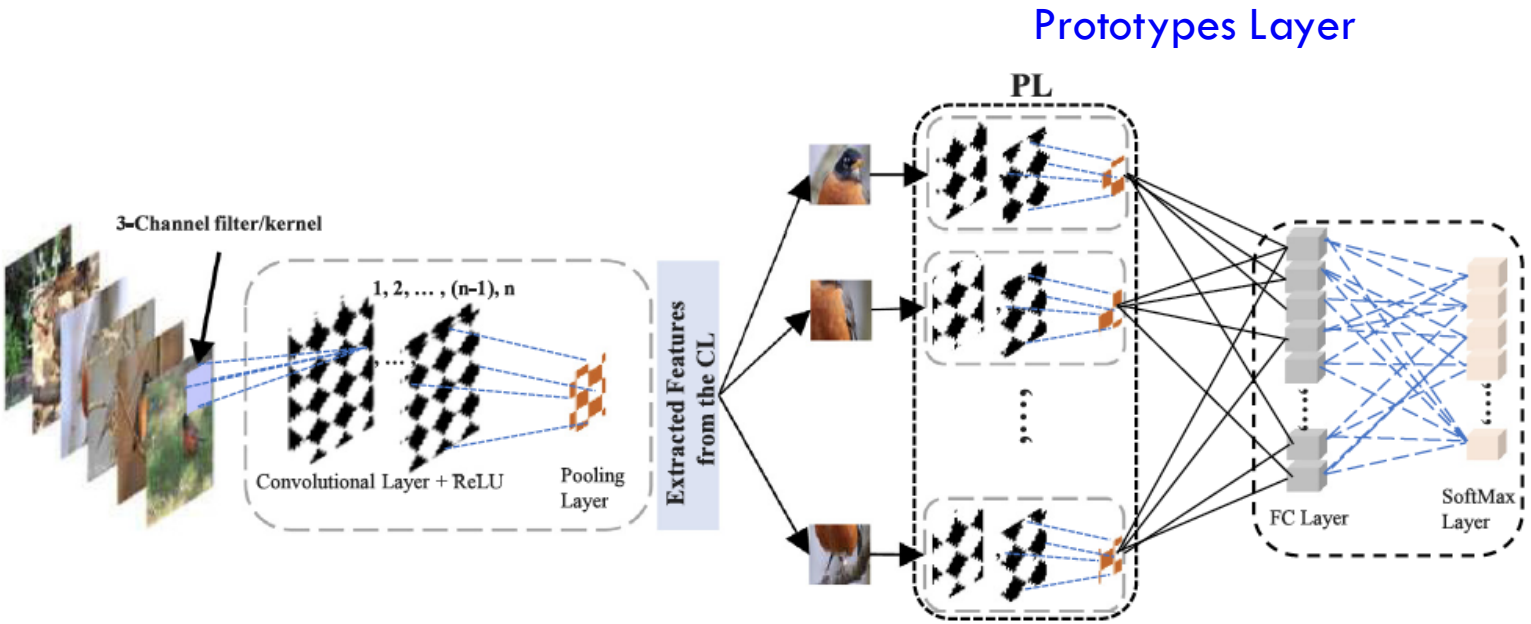


Example of TED

Architectural adjustments

- Aim
  - by adjusting model architectures, it is also possible to improve model explainability
- Examples
  - This Looks Like That
    - Explainable Deep Network (EDN) architecture for image recognition
    - how people explain classification reasoning in terms of different parts of an image being compared to a collection of learned image component prototypes
  - Attention mechanisms
    - provide some degree of explainability and they have altered the way how DL algorithms are used (Attention Map)

This Look Like That

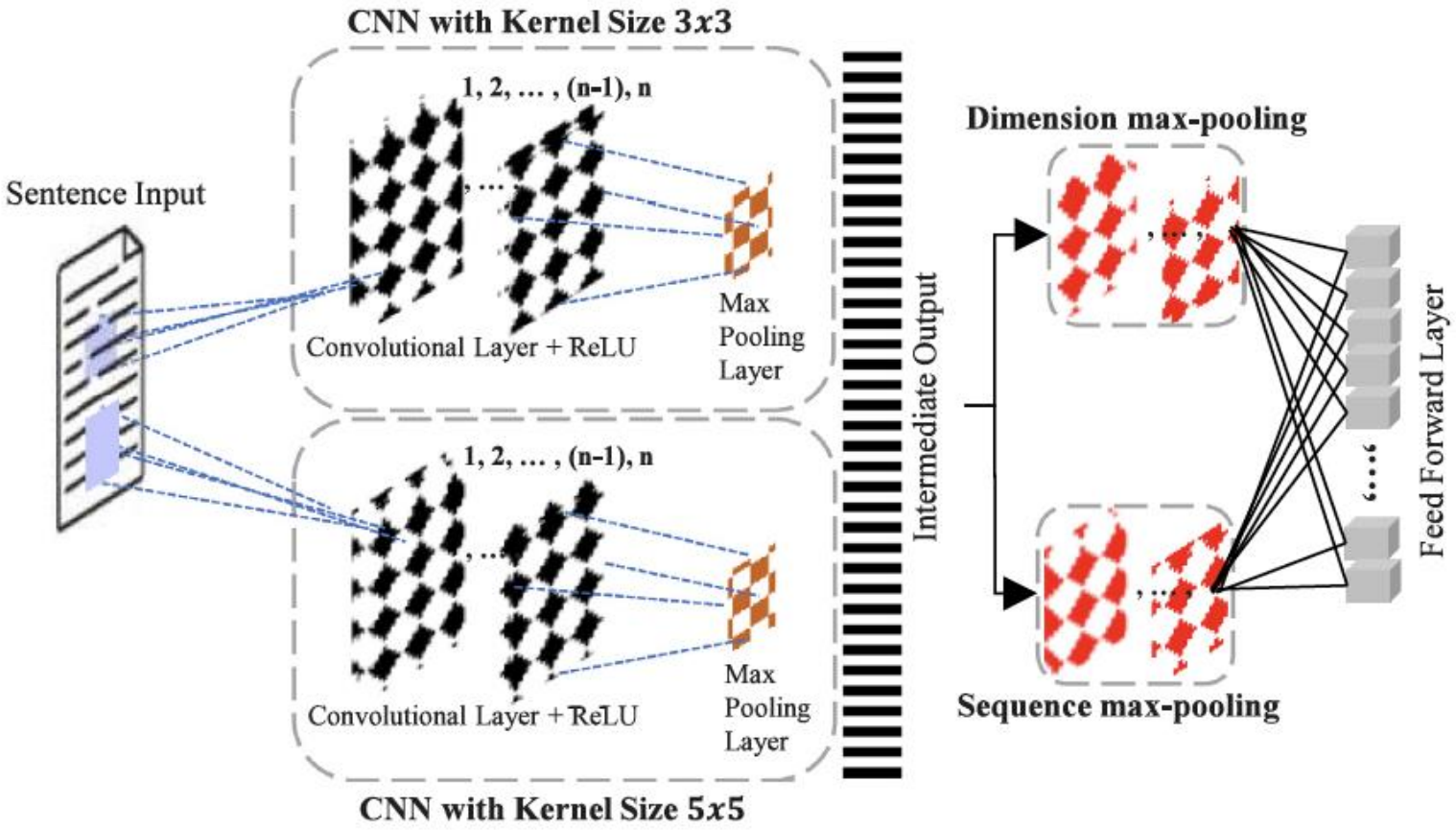


Example of This Look Like That

Regularization

- Aim
  - enhance the prediction performance of AI models, and may also be used to increase model explainability
- Examples
  - Tree Regularization
    - encourage people to learn a model with a decision boundary that can be well approximated using a tiny Decision Tree, allowing humans to simulate the predictions
  - Saliency learning
    - expert annotations concentrate on important portions of the input rather than irrelevant parts, as well as having annotations at the word embedding level rather than at the input dimension level

Saliency learning

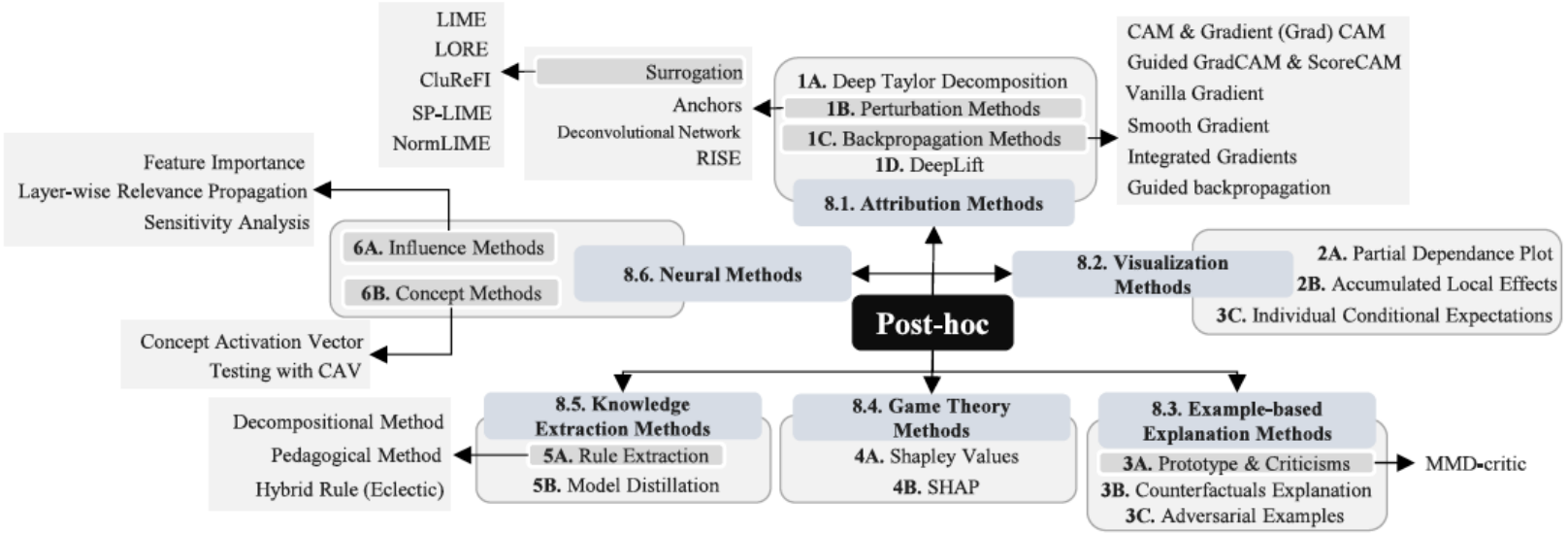


Example of Saliency learning

Post-hoc explainability

- Aim
  - explain black-box models
  - Categories
    - attribution methods
    - visualization methods
    - example-based explanation methods
    - game theory methods
    - knowledge extraction methods
    - neural methods

Post-hoc explainability



Taxonomy of post-hoc explainability

Attribution models

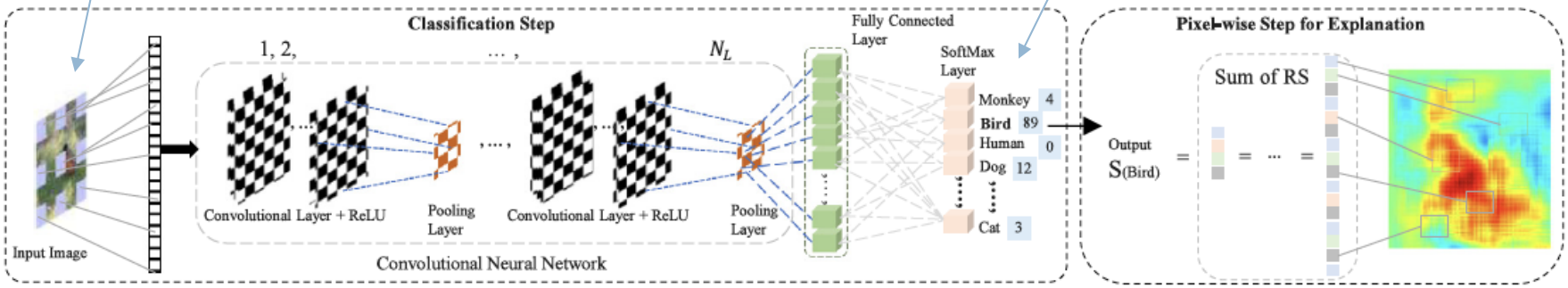
- Aim
  - each pixel of the input image is given an attribution value known as its relevance or contribution
  - Estimate the Relevance Score (RS)
    - Attribution Map
  - Families
    - Deep Taylor Decomposition (DTD)
    - Perturbation Methods
    - Backpropagation Methods
    - DeepLift

Deep Taylor Decomposition

Scores

$$\mathcal{I} = [i_1, i_2, \ldots, i_m] \in \mathbb{R}^N$$

$$S(\mathcal{I}) = [S_1(\mathcal{I}), S_2(\mathcal{I}), \ldots, S_c(\mathcal{I})]$$



Example of DTD

RS of i_m to the output S_c

$$\mathcal{T}(x) = \sum_{i=1}^{d} \mathcal{RS}_i(x) + \mathcal{O}(xx^{\mathcal{T}})$$

Taylor series expansion

$$\mathcal{RS}_i(x) = \frac{\partial \mathcal{T}}{\partial x_i}\Big|_{x=\overline{x}} \cdot (x_i - \overline{x}_i)$$

$$\mathcal{T}(\overline{x}) = 0$$

Pertubation methods

- Aim
  - calculate the attribution of a training instance feature directly by deleting, masking, or changing the input instance
  - a forward pass on the modified input is executed before comparing the obtained results to the original output

Pertubation methods

- Surrogation
  - a distinct model is created to explain the black-box decision either locally or globally
  - the model created is intrinsically interpretable

$$\mathcal{F}^* = \arg\min_{w \in \mathcal{I}} \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \mathcal{FS}\Big(\mathcal{F}(x), \mathcal{B}(x)\Big)$$

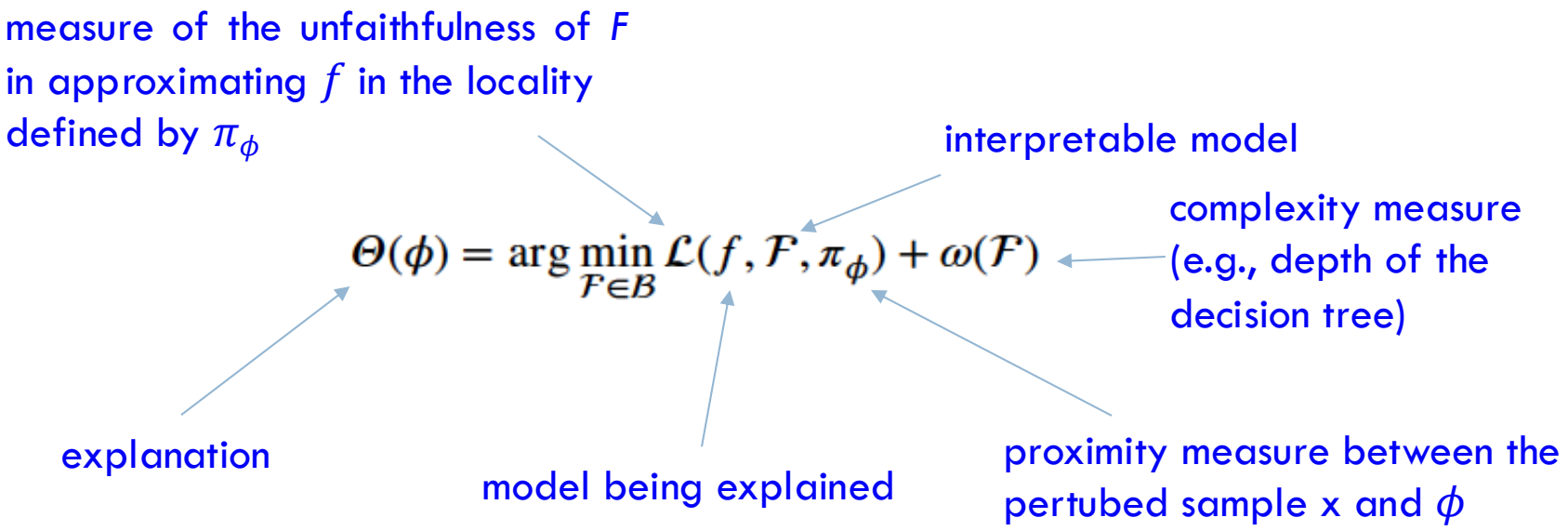
Surrogate

Black-box

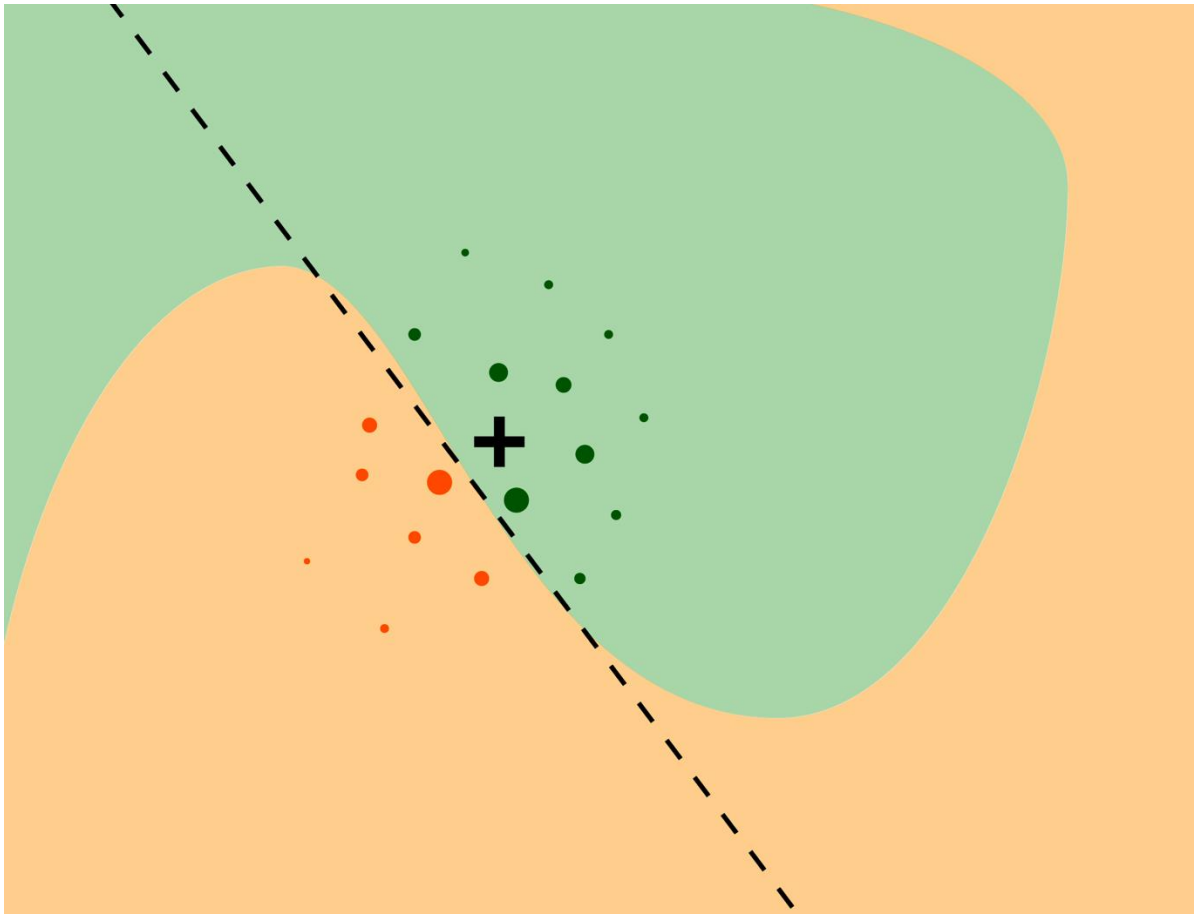
Fidelity score

LIME

- Locally Interpretable Model-Agnostic Explainer
  - local surrogate models to explain individual predictions of black-box ML techniques
  - creates a new dataset using altered samples and the black-box model's predictions



LIME



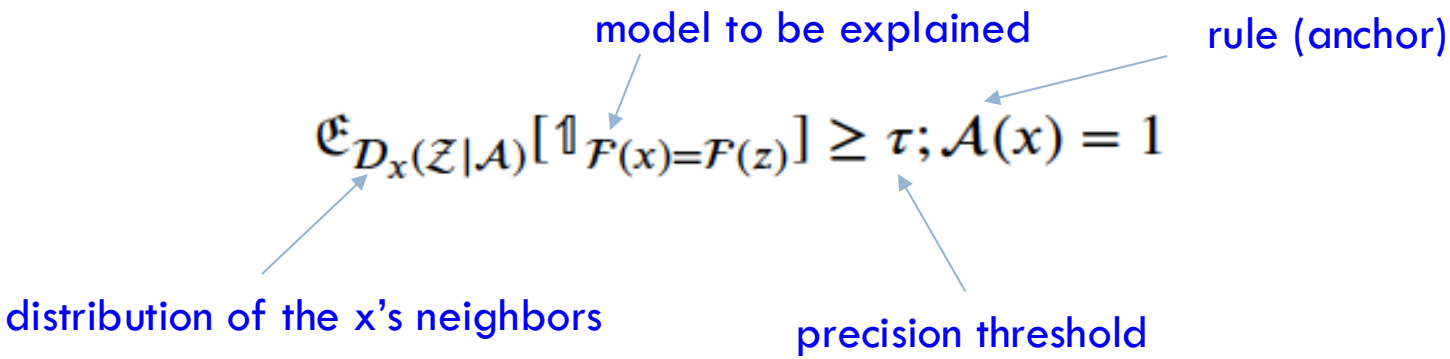
The coloured areas correspond to decision regions for a complex binary classification model. The black cross indicates the instance (observation) of interest. Dots correspond to artificial data around the instance of interest. The dashed line represents a simple linear model fitted to the artificial data. The simple model "explains" local behavior of the black-box model around the instance of interest.

LORE

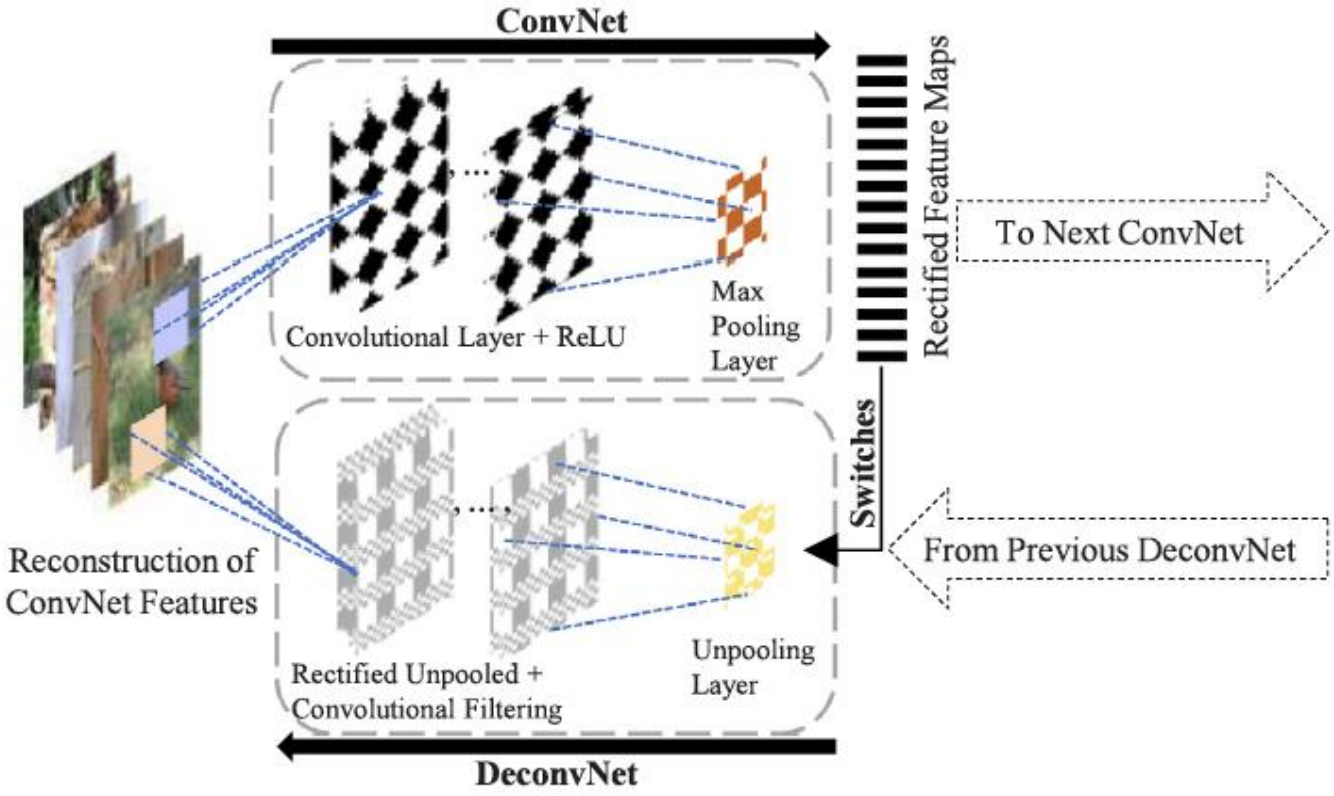
- Local Rule-based Explanation
  - constructs a simple interpretable predictor
  - first using an ad-hoc "genetic algorithm" to generate a balanced set of neighbor instances of the given instance x
  - a decision tree classifier is extracted

Anchors

- Aim
  - variant from LIME that looks for a decision rule that will explain individual predictions of any black-box classification model



Deconvolutional network



DeconvNet - Every layer in the ConvNet has a DeconvNet linked to it, allowing a continuous route back to the original input. DeconvNet can rebuild an approximate replica of the feature identified by ConvNet.

Backpropagation methods

- Aim
  - in one forward and one backward pass to the DNN, backpropagation methods calculate the attribution values for all the input features
  - Main methods
    - Class Activation Map
    - Vanilla-based gradient

Class Activation Map

- Aim
  - Global Average Pooling (GAP) as a structural regularizer in a CNN to reduce the number of parameters used while retaining exceptional performance

$$\mathbb{P}_C = \frac{exp\left(\sum_e w_e^C \cdot \mathcal{GAP}\right)}{\sum_C \left(exp\left(\sum_e w_e^C \cdot \mathcal{GAP}\right)\right)}$$

class probability

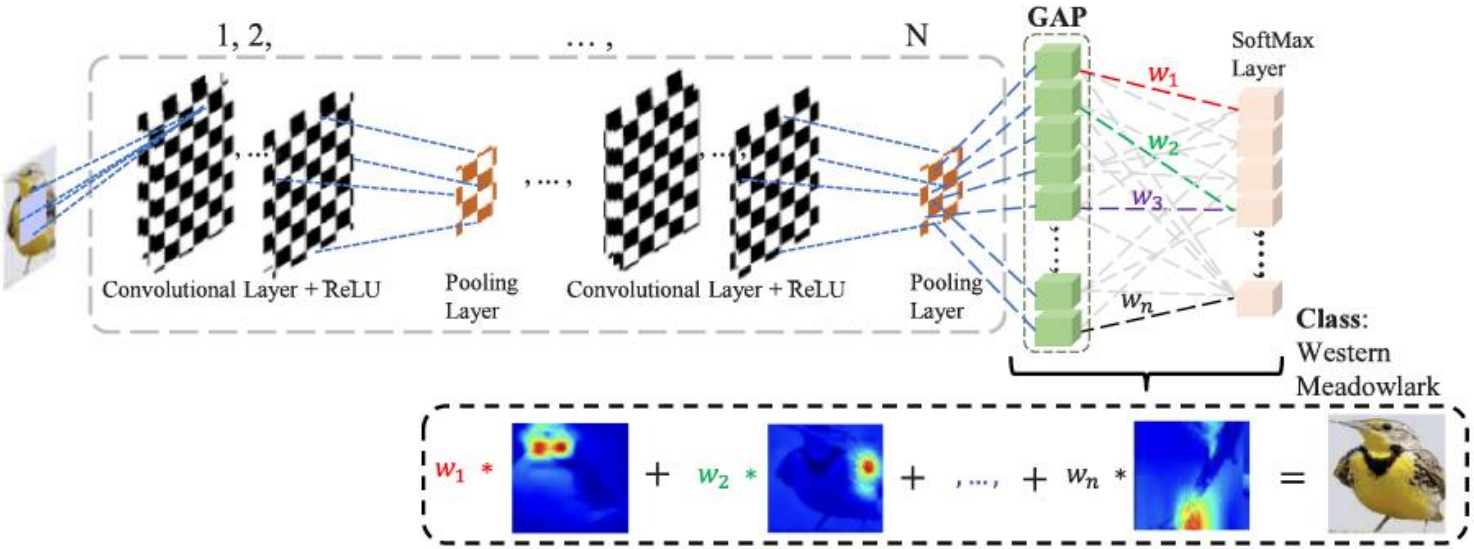
$$\mathcal{GAP} = \sum_{x,y} \mathcal{M}_e(x,y)$$

activation map

$$\mathcal{CAM}_C(x,y) = \sum_e w_e^C \cdot \mathcal{M}_e(x,y)$$

Map

CAM



The spatial average of each unit's feature map, from the last possible CL, is generated by the GAP. The final result is generated using a weighted sum of the spatial data.

The discriminative areas, distinct to each class, are highlighted in the CAM

Grad-CAM

- Gradient-weighted CAM
  - visual explanations for any model in the CNN family without needing to go through architectural modifications or retraining
  - assigns significance ratings to each neuron for the given target class using the gradient information backpropagated to the final convolutional layer

$$\mathcal{M}^{C}_{Grad-CAM} = ReLU\left(\sum_{k} w^{C}_{k} m_{k}\right)$$

Grad-CAM localization map

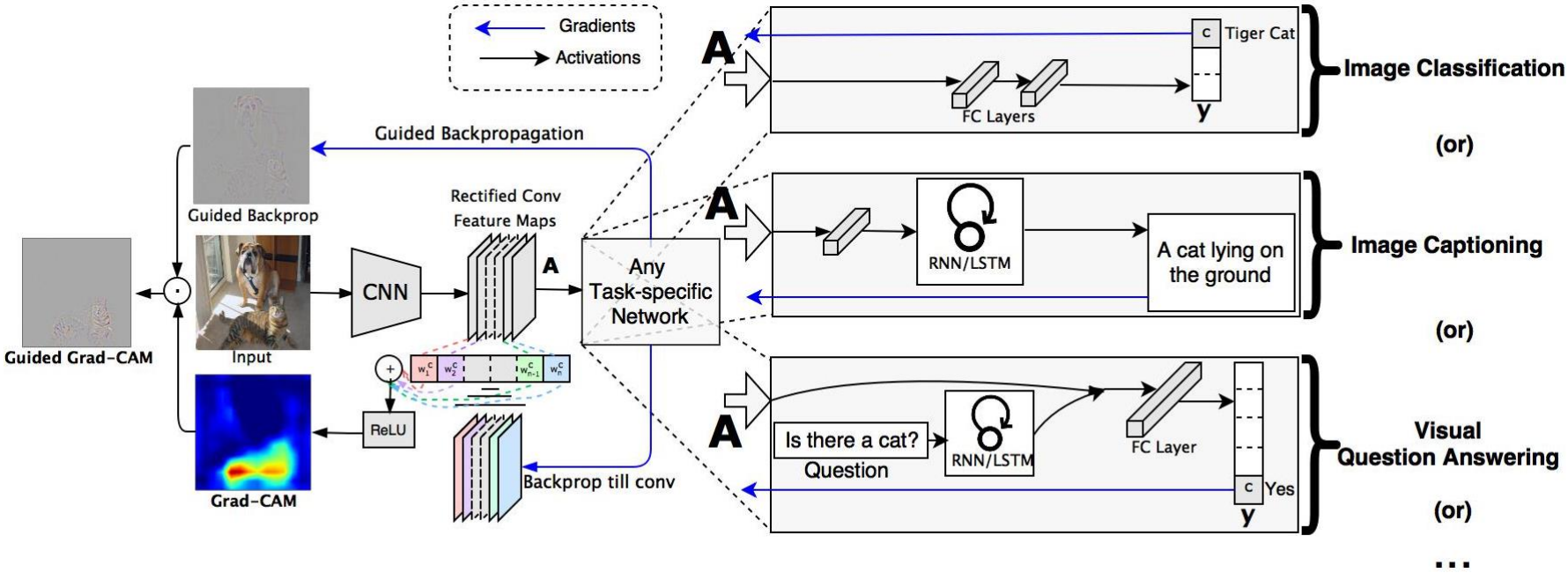
$$w^{C}_{k} = \frac{1}{y}\sum_{p}\sum_{q}\gamma_{k}(p,q) = \frac{1}{y}\sum_{p}\sum_{q}\frac{\partial S_{C}}{\partial m_{k}(p,q)}$$

class score

significant weights matrix for the neurons

Rectified Convolutional Feature Maps

Grad-CAM



Given an image and a class of interest (e.g., 'tiger cat' or any other type of differentiable output) as input, we forward propagate the image through the CNN part of the model and then through task-specific computations to obtain a raw score for the category. The gradients are set to zero for all classes except the desired class (tiger cat), which is set to 1. This signal is then backpropagated to the rectified convolutional feature maps of interest, which we combine to compute the coarse Grad-CAM localization (blue heatmap) which represents where the model has to look to make the particular decision. Finally, we pointwise multiply the heatmap with guided backpropagation to get Guided Grad-CAM visualizations which are both high-resolution and concept-specific.

Vanilla Gradient

- Image Specific Class Saliency (Saliency Maps)
  - the loss function's gradient is calculated with regard to the input pixels

class relevance score

pixels of image

$$S_C(I) = b_C + Iw$$

network's weight

bias

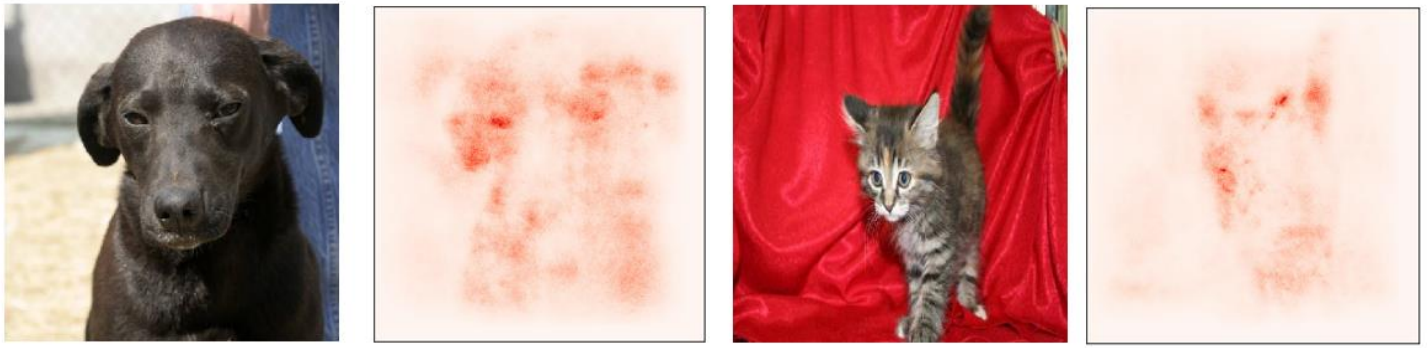
  - For DNN the scoring function is nonlinear

$$w = \frac{\partial S_C}{\partial I}\Big|_{I_0}$$

$$\mathcal{M}_{ij} = \max_{ch} \left| w_{idx(i,j,ch)} \right|$$

Saliency map for an RGB image

Saliency Map



Examples of saliency maps

DeepLIFT

- Aim
  - frames the topic of significance in terms of deviations from a reference condition which is selected by the user

Layer

interested neuron i

$$\mathcal{RS}_i^{\mathcal{N}_L}(I) = S_i(I) - S_i(\hat{I})$$

reference condition

$$\mathcal{RS}_i^{\mathcal{N}_L}(I) = \sum_i \frac{I_{ij} - \hat{I}_{ij}}{\sum_j I_{ij} - \sum_j \hat{I}_{ij}} \mathcal{RS}_i^{\mathcal{N}_L+1}$$

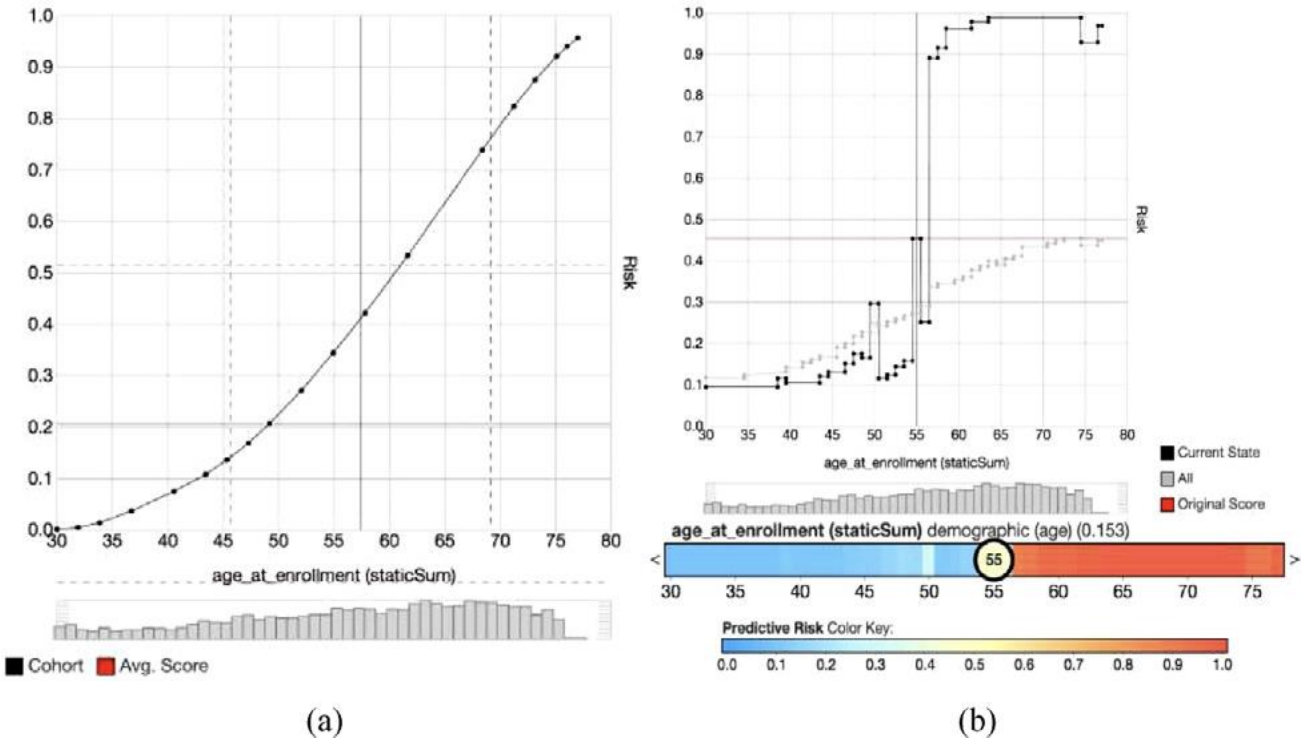
For all the layers

Visualization methods

- Aim
  - Understanding an AI model, by visualizing its representations to investigate the underlying patterns
  - Examples
    - Partial Dependence Plot
    - Individual Conditional Expectations
    - Accumulated Local Effects

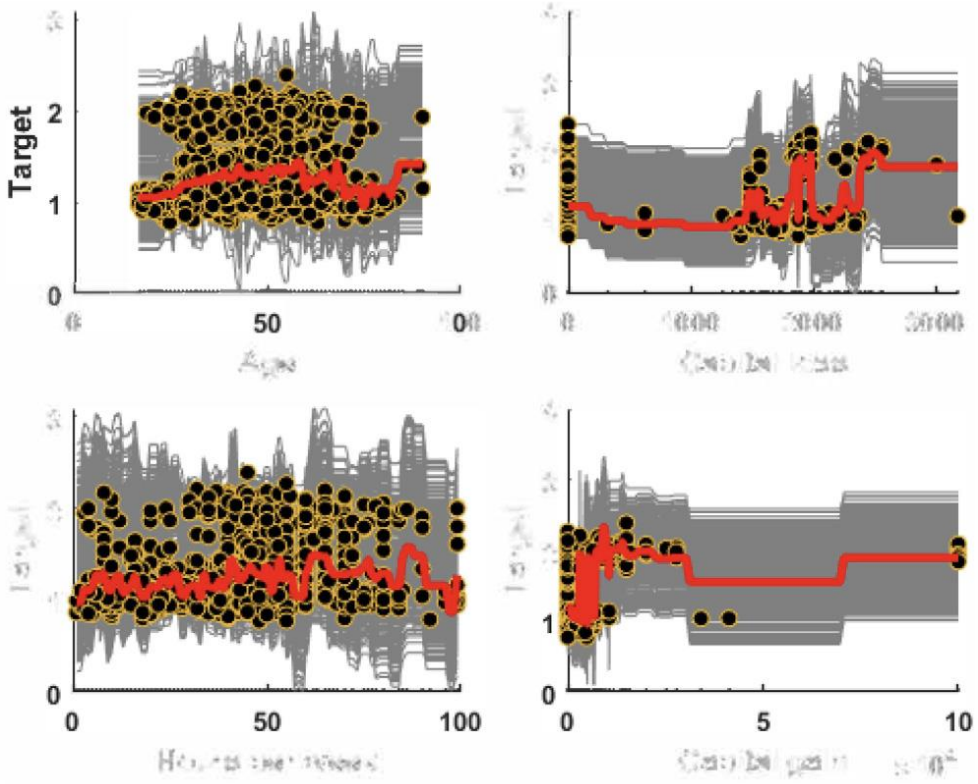
Partial Dependence Plot

- Aim
  - connection between an individual feature and the target



Individual Conditional Expectations

- Aim
  - connection between the target and a single feature, rather than the whole model



Example-based explanation methods

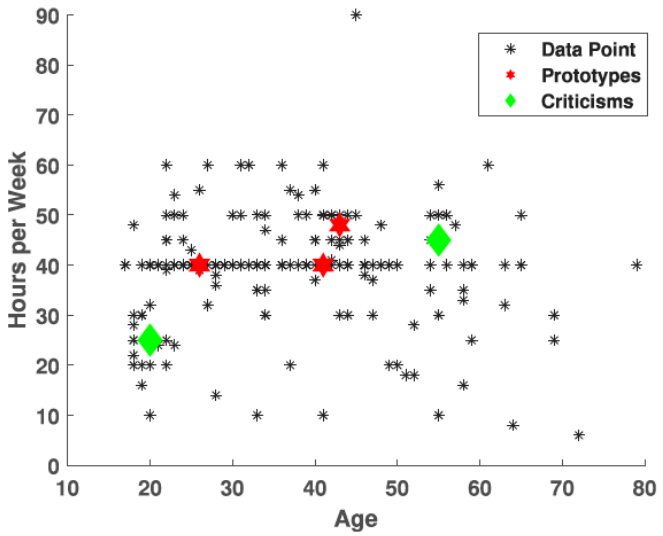
- Aim
  - case-based explanations
  - Examples
    - prototypes and criticisms
    - counterfactuals
    - adversarial examples

Prototypes and criticisms

- Prototype
  - are single instances with the capability to represent the entire dataset
- Criticism
  - is a data instance that is not included in the collection of prototypes because it is distinct enough for representing complimentary insights

- Maximum Mean Discrepancy
  - compares the data distribution with the distribution of selected prototypes
    - firstly, the user defines the number of prototypes and criticisms to be identified
    - prototypes and criticisms are discovered using a greedy search technique

Prototypes and criticisms



Prototypes

Criticisms

Counterfactuals

- Aim
  - "contrary-to-fact" examples
- Loss function

instance of interest x, a counterfactual x', and the desired outcome y'

$$\mathcal{L}(x, x', y', \lambda) = \lambda.\left(\hat{f}(x') - y'\right)^2 + d(x, x')$$

Manhattan distance

ls

(b) Counterfactual explanation

(c) Counterfactual class: *wing*

small set of segments that, in case of removal, alters the classification.

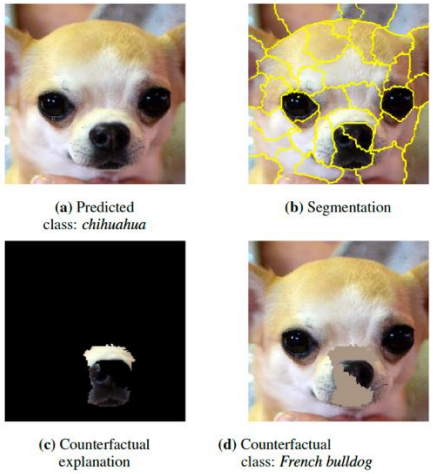
(a) *French bulldog*

(b) *Segmentation*

(c) Counterfactual explanation

(d) Counterfactual class: *chihuahua*

Counterfactuals



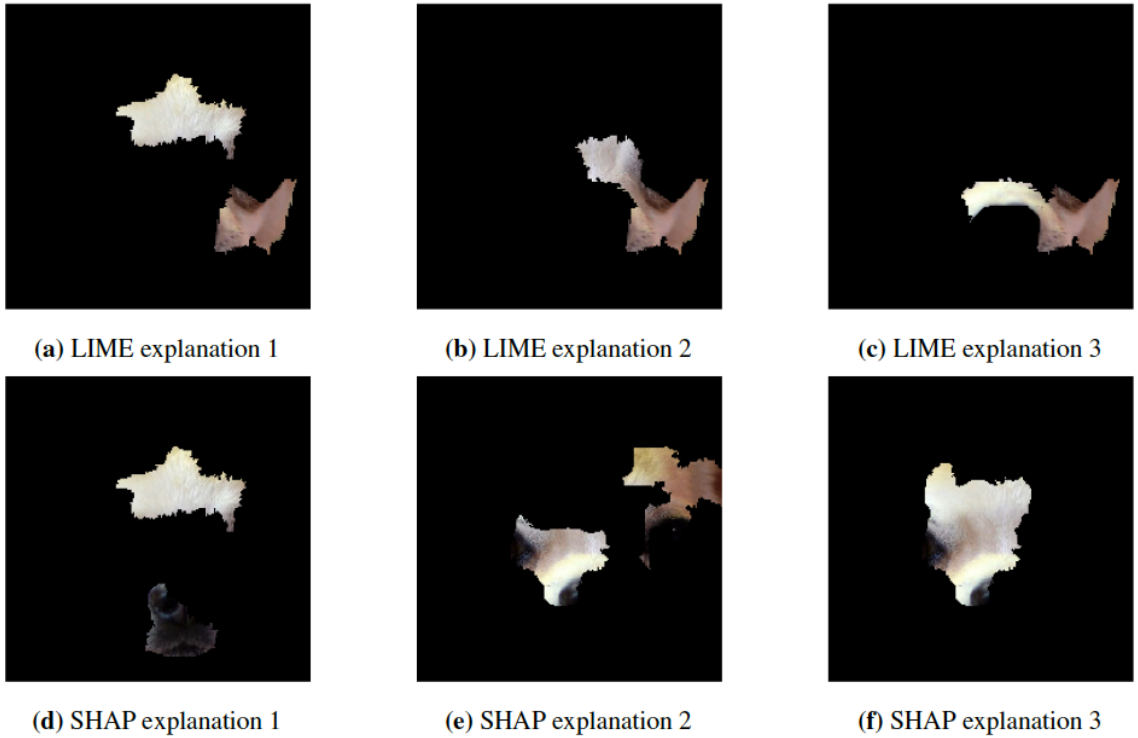
(a) Predicted class: *chihuahua*

(b) Segmentation

(c) Counterfactual explanation

(d) Counterfactual class: *French bulldog*

Post-hoc methdologies comparison



(a) LIME explanation 1

(b) LIME explanation 2

(c) LIME explanation 3

(d) SHAP explanation 1

(e) SHAP explanation 2

(f) SHAP explanation 3

Game theory methods

- Aim
  - how much each player in a coalition game contributes (Lloyd Shapley, 1953)
  - game
    - single instance of a dataset's prediction in a task
  - gain
    - difference between the actual prediction for the given prediction and the average of predictions for all instances in the dataset
  - players
    - are the instance's feature values who work together to obtain the gain
    - the Shapely value of a feature tells us how much it contributes to a particular prediction outcome
- Examples
  - Shapley Additive Explanation (SHAP)

SHAP

- Aim
  - is a unified way to understand the output of any ML model
  - explaining individual predictions using the coalition game's best Shapley values
  - a player can be represented by a single feature value, such as in tabular data
  - a player can also be made up of a collection of feature values
    - pixels can be grouped into superpixels, and the information to make the prediction that describes the image is spread among them
  - The Shapley value explanation is an Additive Feature Attribution approach

SHAP

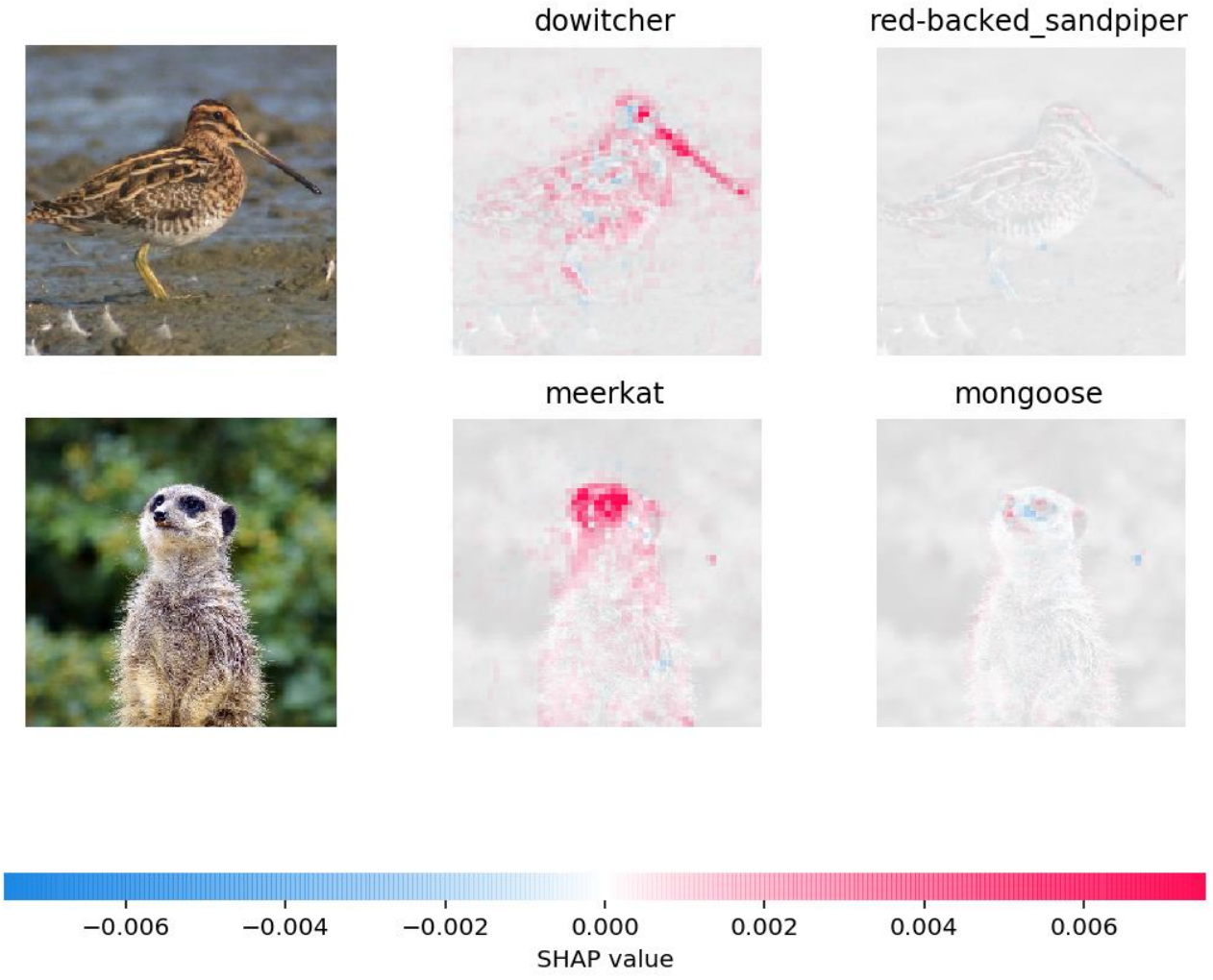
- Explanation

$$g(\hat{z}) = \Phi_0 + \sum_{i=1}^{M} \Phi_i \hat{z}_i$$

the feature attribution for ith feature is Φ_i, the maximum size of the coalition is M, the coalition vector (the simplified features) is denoted by $Z \in \{0, 1\}^M$

Where 1 in the coalition vector indicates that the relevant feature value is "present", whereas 0 indicates that the feature is "missing".

SHAP



Red pixels represent positive SHAP values that increase the probability of the class, while blue pixels represent negative SHAP values the reduce the probability of the class.

Knowledge extraction methods

- Aim
  - describe how black-box ML models behave internally
- Examples
  - Rule extraction
  - Model Distillation

Rule extraction

- Aim
  - produces an understandable but rough approximation of a network's predicted behavior from the training data and the trained ANN
- Two main categories
  - Propositional/Boolean logic
  - Non-conventional logic
- Rule extraction techniques
  - Fuzzy modeling
  - Genetic programming
  - Boolean rule extracion

Rule extraction

- Relation between the extracted rule and the trained NN architecture
- Three distinct types of methods
  - Decompositional methods
    - operate on the neuron level rather than over the whole NN design
  - Pedagogical methods
    - operate disregarding the NN architectural design
  - Eclectic methods
    - combination of decompositional and pedagogical methods

Model distillation

- Aim
  - transferring information (dark knowledge) from a teacher network (e.g., a DNN) to a student network (e.g., a shallow NN) via model compression
- Interpretable Mimic Learning
  - method for learning phenotype features that are interpretable for generating robust predictions while imitating the performance of black-box DL models
- DarkSight
  - a visualization technique for understanding the predictions of black-box

Neural Methods

- Aim
  - explain specific predictions, simplify neural networks, or visualize the features and concepts that a neural network has learned
- Examples
  - Influence Methods
  - Concept Methods

Influence methods

- Aim
  - By altering the input or internal elements and analyzing which ones (and how much) change model performance, these methods assess the significance of a feature

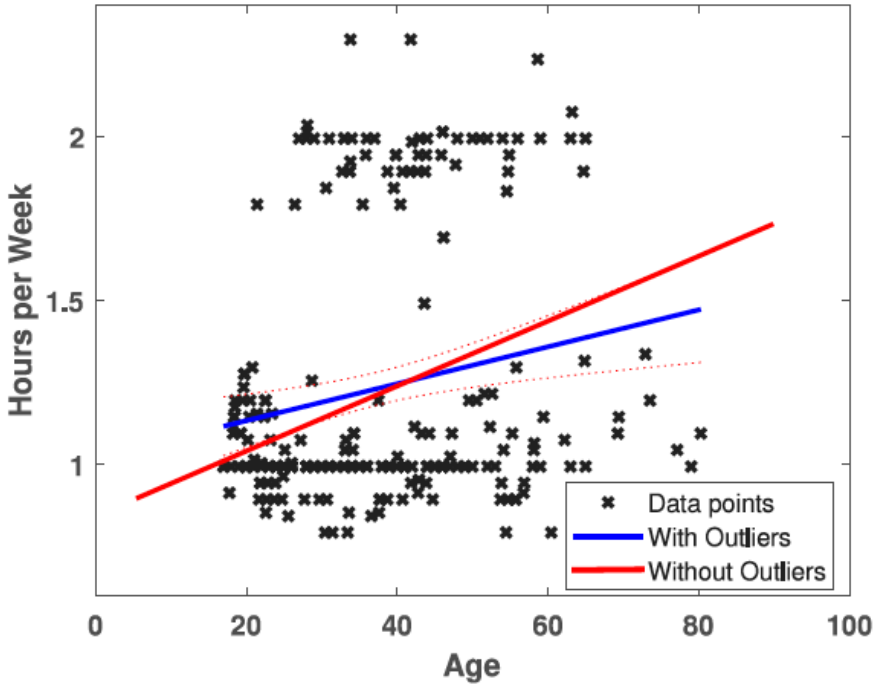
- Three different techniques in the literature for determining the significance of an input variable
  - feature importance
  - Layer-wise Relevance Propagation (LRP)
  - Sensitivity Analysis (SA)

Feature importance

- Aim
  - is possible to assign a degree of significant value to each feature
  - especially useful when the selected instance has a significant impact on model performance

- Feature importance is calculated using the change in the model's error seen in the feature permutation process
  - Leave-One-Covariate-Out
  - Model Class Reliance

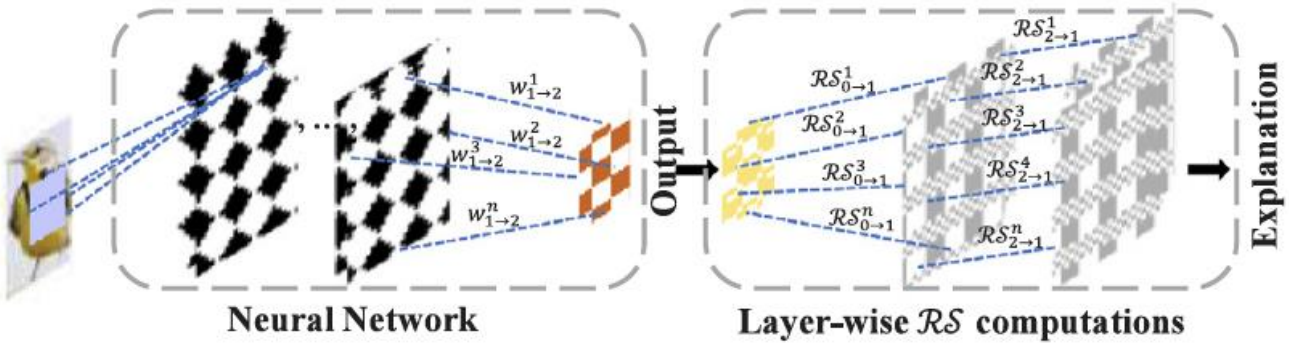
Feature importance



A linear model was trained on two cases, one with unimportant features and one without unimportant features. In the instance, without unimportant features, the slope produced by the model changes significantly in contrast to the instance with unimportant features.

Layer-wise Relevance Propagation

- Aim
  - Starting from a network's output layer and backpropagating up to the input layer, LRP redistributes the prediction functions in their opposite order



$$\sum_j \mathcal{RS}_{j\leftarrow k} = \mathcal{RS}_k \qquad \mathcal{RS}_j = \sum_k \mathcal{RS}_{j\leftarrow k}$$

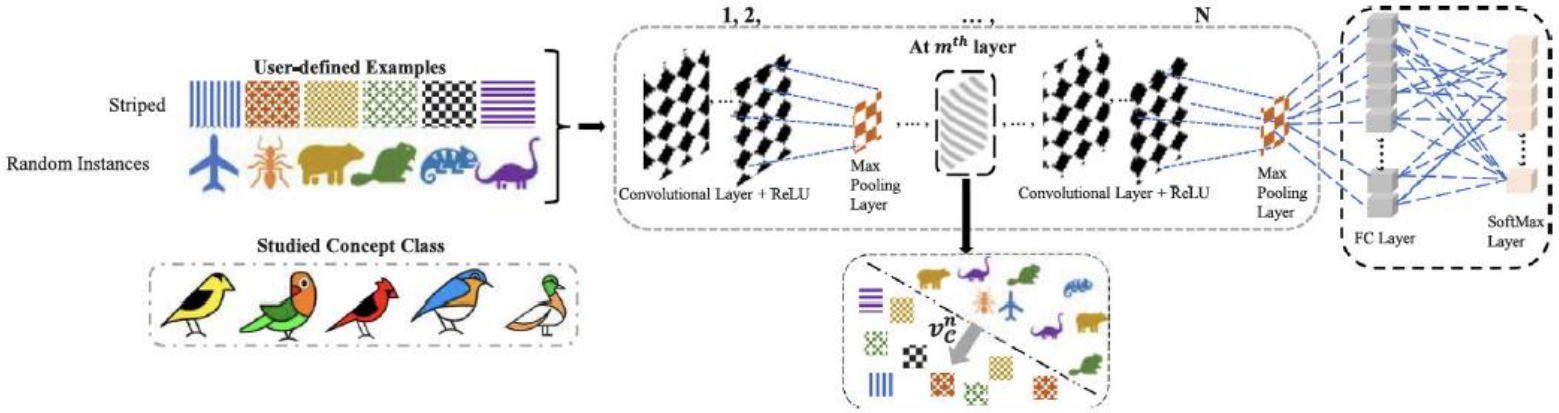
$$\sum_{i=1}^{d} \mathcal{RS}_i = \cdots = \sum_j \mathcal{RS}_j = \sum_k \mathcal{RS}_k = \cdots = \mathcal{F}(x)$$

Sensitivity analysis

- Aim
  - most important input features are those with the greatest impact on the output
  - often used to check for model trustworthiness and stability, as a tool for identifying or removing irrelevant input features

Concept Activation Vectors

- Aim
  - human-friendly explanations of the internal states of NNs globally



CAVs are created by teaching a linear classifier to discriminate between the activation generated by a concept's instances and the activations caused by examples at the m-th layer.

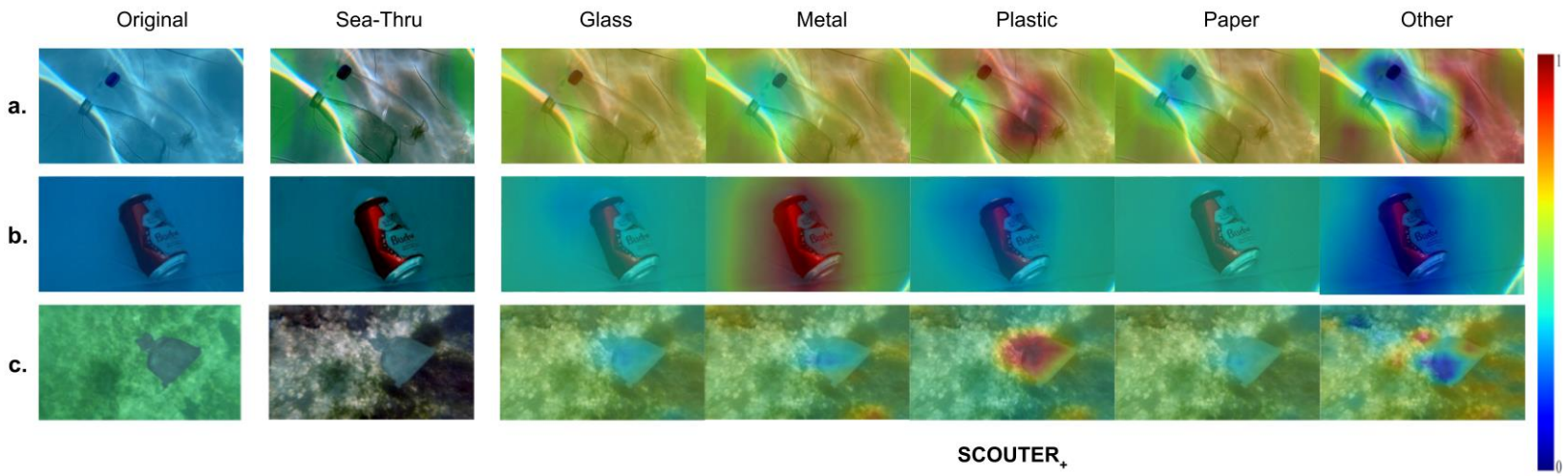
SCOUTER

- Slot Attention-based Classifier for Explainable Image Recognition
  - SCOUTER's explanation is involved in the final confidence for each category, offering more intuitive interpretation
  - all the categories have their corresponding positive or negative explanation
    - "why the image is of a certain category" or "why the image is not of a certain category"

SCOUTER

Positive and negative explanations. The images from top to down are from the test sets of MNIST, Con-text, and CUB-200 datasets. The models trained with positive (+) and negative (−) SCOUTER losses can respectively highlight the positive and negative supports, based on which one can reason why or why not the images are classified into the corresponding categories.

SCOUTER



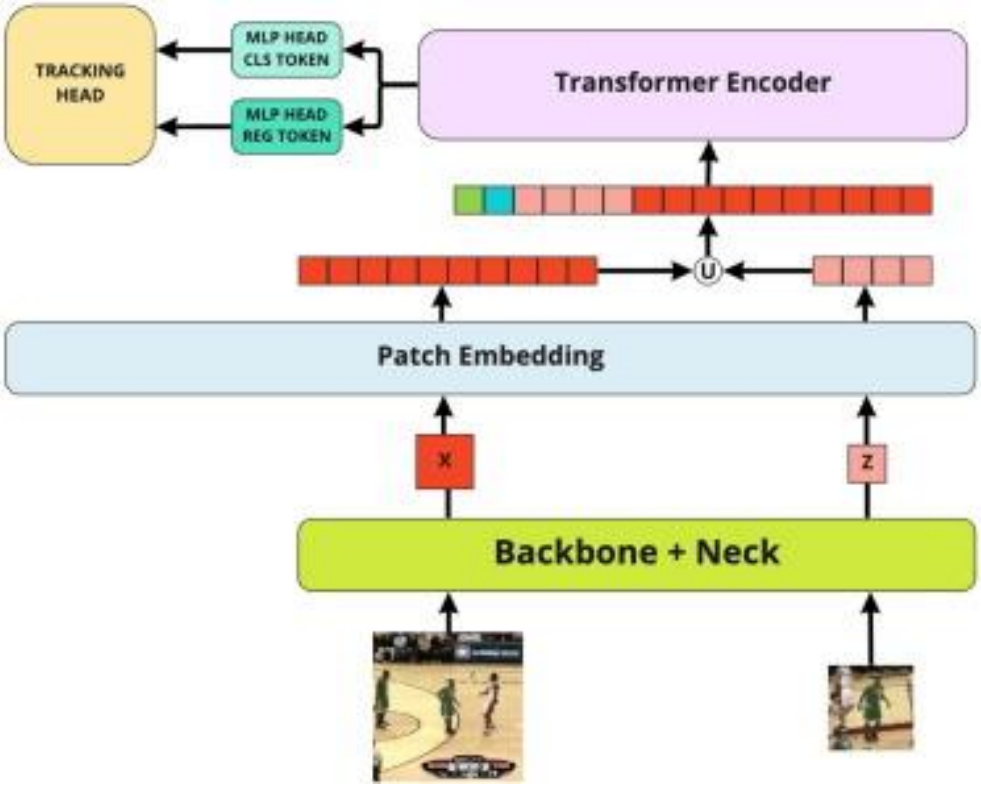
Classification using SeaThru with SCOUTER - a. Classification of a real plastic bottle image; b. Classification of a real metal can image; c. Classification of a synthetic plastic bag image.

G. Mellone et al., Exploring the Effectiveness of Slot Attention-based Classifier in Detecting Underwater Marine Litter: A Study, Smart Innovation, Systems and Technologies

ViTCRT

- Tracking vision transformer with Class and Regression Tokens
  - visual object tracking model based on siamese network and vision transformer
  - learn a robust characterization of the problem with an explainable architecture, understanding the motivation of the choice of the neural network

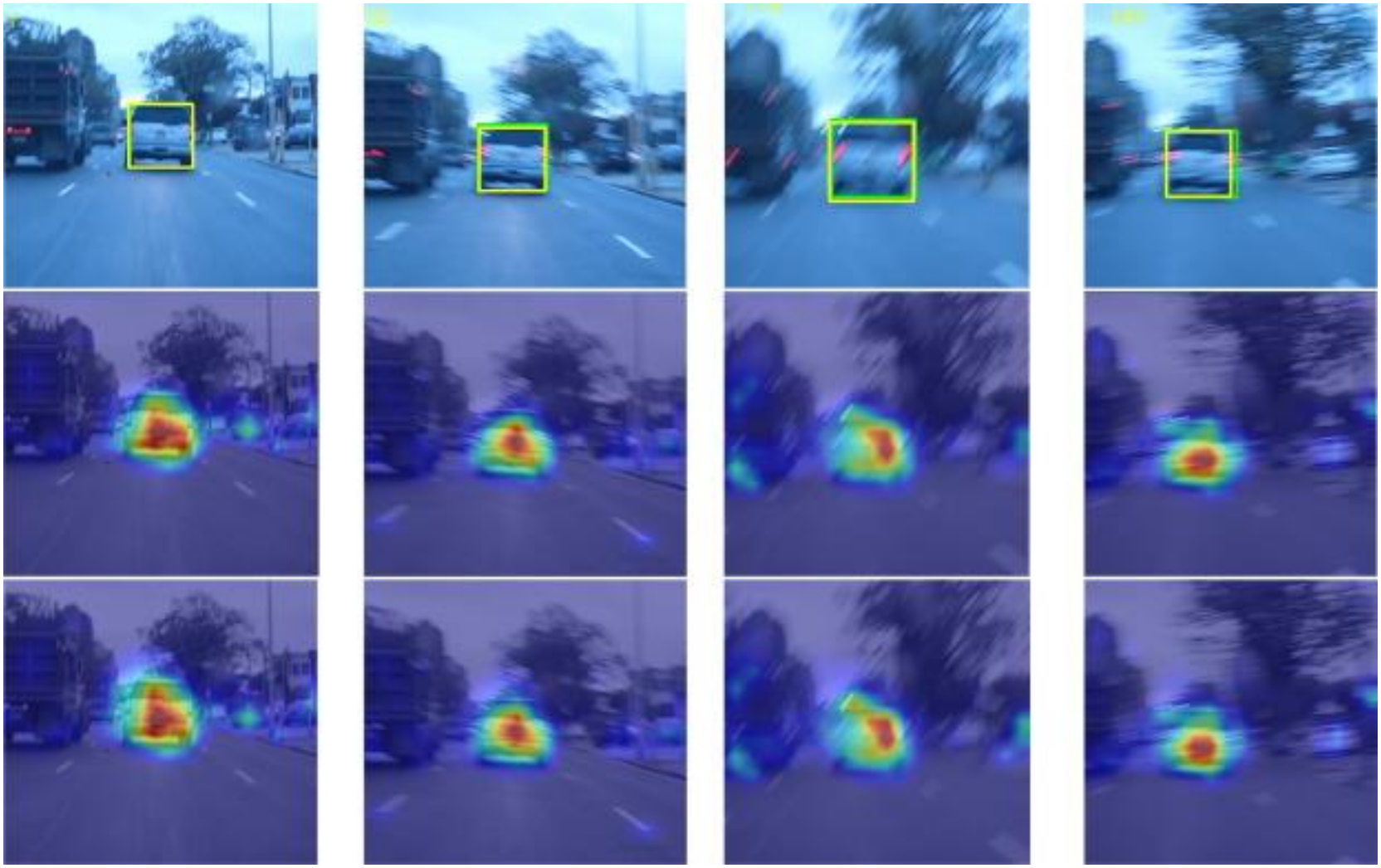
ViTCRT



Tracking vision transformer with class and regression tokens

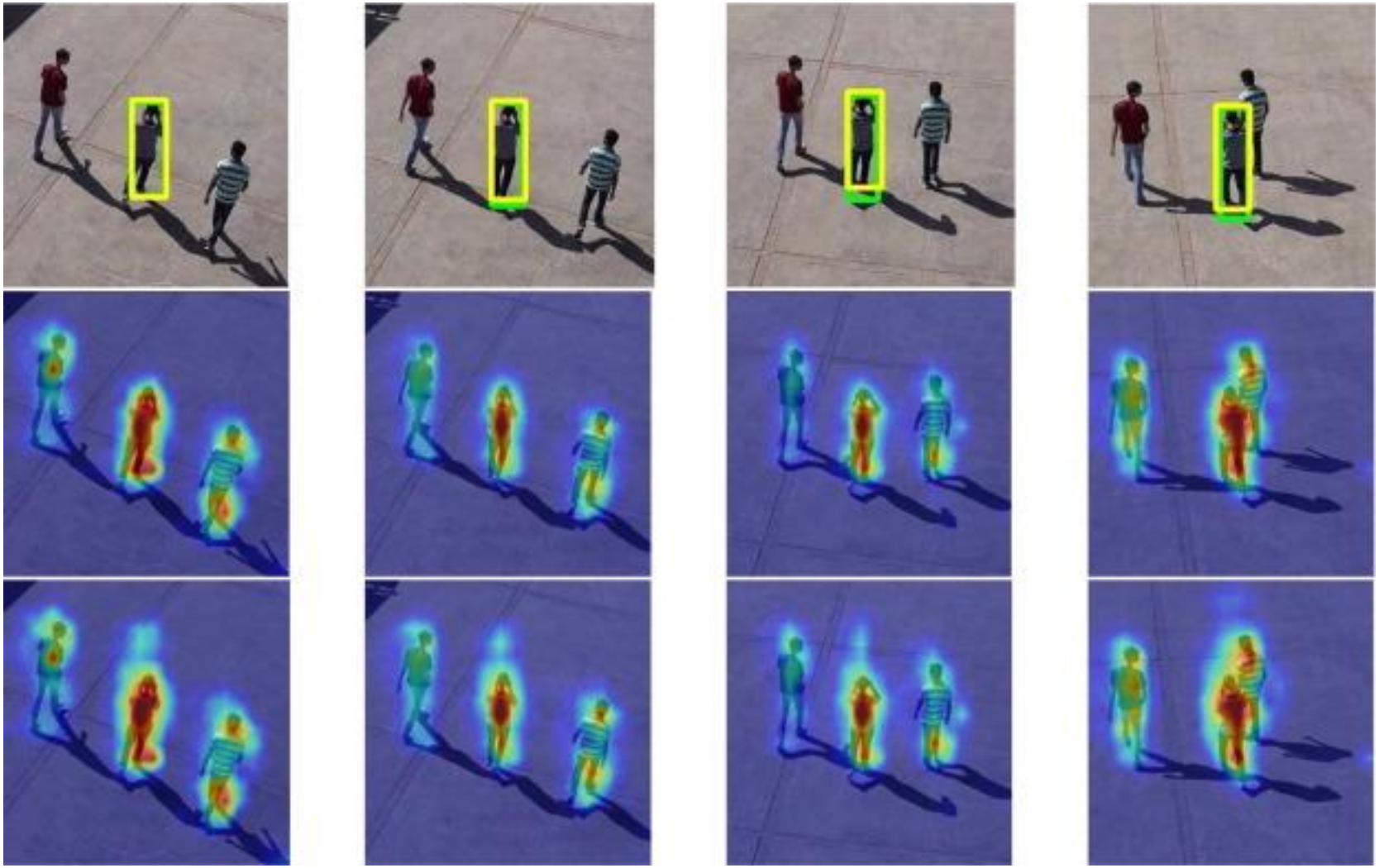
E. Di Nardo et al., Tracking vision transformer with class and regression tokens, Information Sciences, Vol. 619, p.p. 276-287, 2023
Kristan, M. et al., The Tenth Visual Object Tracking VOT2022 Challenge Results, ECCV 2022. Lecture Notes in Computer Science, vol 13808

Visual tracking



Outputs for sequence *BlurCar1* in OTB100 with the attention of two tokens overlaying the image. *First row*: Bounding Box outputs. *Second row*: Regression token attention. *Third row*: Classification token attention.

Visual tracking



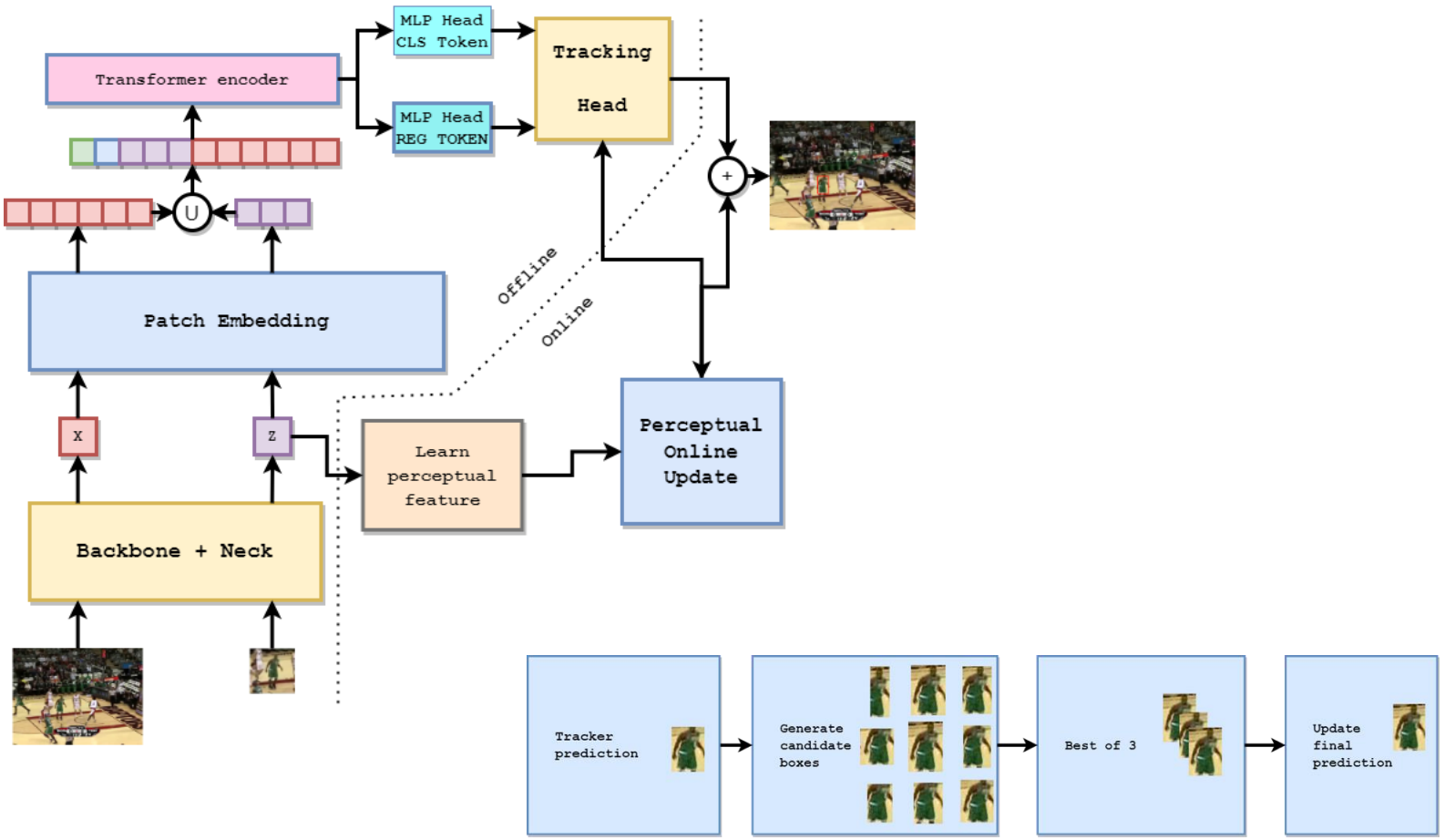
Outputs for sequence *Group1_2* in UAV123 with the attention of two tokens overlaying the image. *First row*: Bounding Box outputs. *Second row*: Regression token attention. *Third row*: Classification token attention.

On-line VOT

- Perceptual Learning for On-Line Visual Object Tracking
  - Perceptual Learning for on-line learning in combination with ViTCRT for learning multiple sub-tasks
  - Perceptual Learning is the process by which an individual's ability to recognize or discriminate between stimuli improves through practice or experience

D. De Cicco et al., Perceptual Learning for On-Line Visual Object Tracking, Smart Innovation, Systems and Technologies

On-line VOT



D. De Cicco et al., Perceptual Learning for On-Line Visual Object Tracking, Smart Innovation, Systems and Technologies

Captioning

- Discrete Diffusion Model for Image Captioning
  - diffusion-based captioning model fine-tuned by a self-critical reinforcement learning technique

DD. A girl blowing out candles on a birthday cake

DD-RL. A little girl holding a large piece of a cake

ATT. A woman is sitting at a table with food

ATT-RL. A woman sitting at a table with a plate of food

TR. A man sitting on a table with a pizza

TR-RL. A woman is sitting at a table with a pizza

FC A man sits with a dish of food

FC-RL. A man is sitting at a table with a plate of food

V. Silvio et al., Discrete Diffusion Model for Image Captioning by Self-Critical Learning, Smart Innovation, Systems and Technologies

Neuro-symbolic AI

- Neuro-symbolic
  - field of artificial intelligence that integrates neural and symbolic AI architectures to address the weaknesses of each, providing a robust AI capable of reasoning, learning, and cognitive modeling
  - Leslie Valiant
    - *the effective construction of rich computational cognitive models demands the combination of symbolic reasoning and efficient machine learning*
  - Gary Marcus
    - *We cannot construct rich cognitive models in an adequate, automated way without the triumvirate of hybrid architecture, rich prior knowledge, and sophisticated techniques for reasoning*

Neuro-symbolic AI

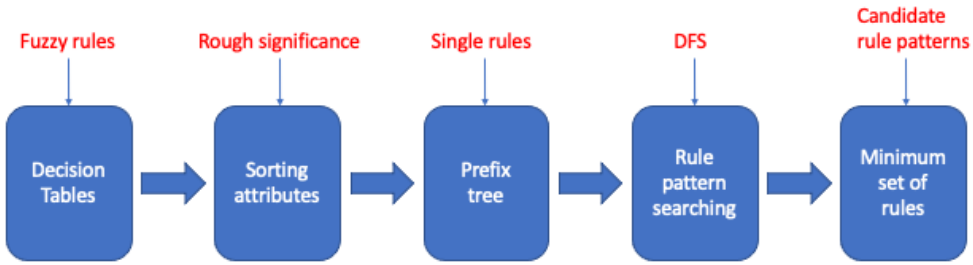
- Categories
  - Symbolic Neural symbolic
    - current approach of many neural models in natural language processing, where words or subword tokens are the ultimate input and output of large language models
    - Examples include BERT, RoBERTa, and GPT-3
  - Symbolic[Neural]
    - exemplified by AlphaGo, where symbolic techniques are used to invoke neural techniques
    - In this case, the symbolic approach is Monte Carlo tree search and the neural techniques learn how to evaluate game positions
  - Neural | Symbolic
    - uses a neural architecture to interpret perceptual data as symbols and relationships that are reasoned about symbolically
    - Neural-Concept Learner is an example

Neuro-symbolic AI

- Categories
  - Neural: Symbolic → Neural
    - relies on symbolic reasoning to generate or label training data that is subsequently learned by a deep learning model
      - e.g., to train a neural model for symbolic computation by using a Macsyma-like symbolic mathematics system to create or label examples
  - Neural_{Symbolic}
    - uses a neural net that is generated from symbolic rules
    - e.g., Neural Theorem Prover which constructs a neural network from an AND-OR proof tree generated from knowledge base rules and terms
    - Logic Tensor Networks also fall into this category
  - Neural[Symbolic]
    - allows a neural model to directly call a symbolic reasoning engine
    - e.g., to perform an action or evaluate a state
    - an example would be ChatGPT using a plugin to query Wolfram Alpha

Fuzzy Logic

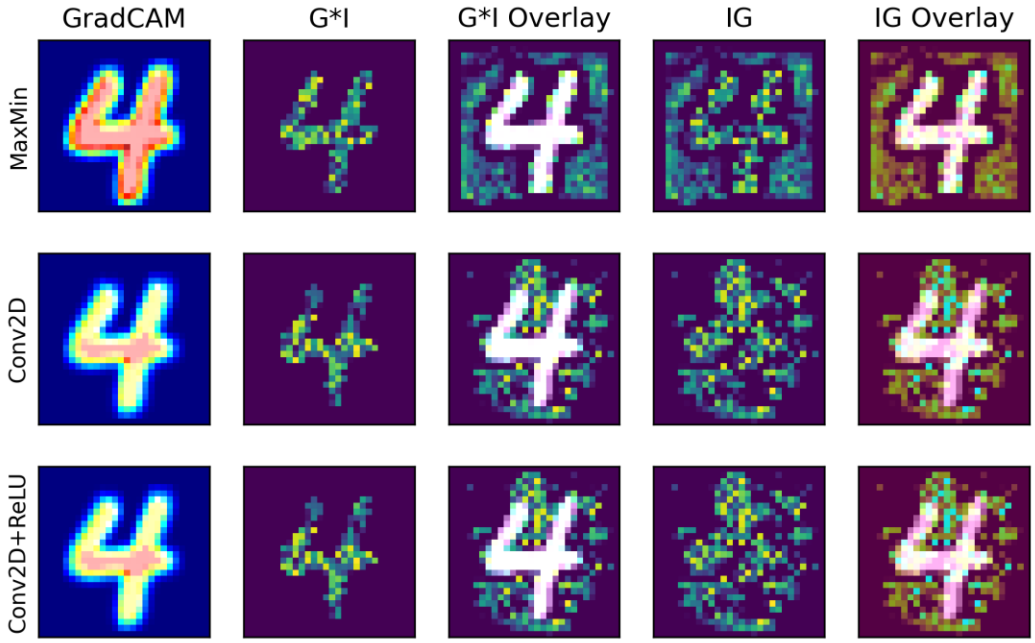
- A fuzzy rule base minimization perspective in XAI
  - The reduction of the fuzzy rules makes the rule base simpler, and thus easier to produce explainable inference systems (e.g., decision support systems and recommenders)



Camastra et al., A Fuzzy Rule Base Minimization Perspective in XAI, Proceedings of WILF 2021, CEUR Workshop Proceedings

Fuzzy Logic

- Advanced Fuzzy Relational Neural Network
  - model for extrapolating relevant information from images data permitting to obtain a clearer indication on the classification processes



Di Nardo et al., Advanced Fuzzy Relational Neural Network, Proceedings of WILF 2021, CEUR Workshop Proceedings

Visual Question Answering

- Aim
  - deep representation learning for visual recognition and language understanding, and symbolic program execution for reasoning

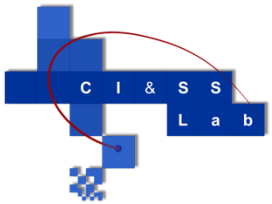
How many blocks are on the right of the three-level tower?

Will the block tower fall if the top block is removed?

What is the shape of the object closest to the large cylinder?

Are there more trees than animals?

Labs @ UniParthenope



Computational Intelligence and Smart System Lab
http://cisslab.uniparthenope.it

Computer Vision & Pattern Recognition "Alfredo Petrosino" Lab
http://cvprlab.uniparthenope.it

High Performance Scientific Computing Smart Lab
http://hpsclab.uniparthenope.it

Multidisciplinary Research Laboratory for the Artificial Intelligence at the Sea
https://neptunia.uniparthenope.it