

Explainable AI

Pushpak Bhattacharyya

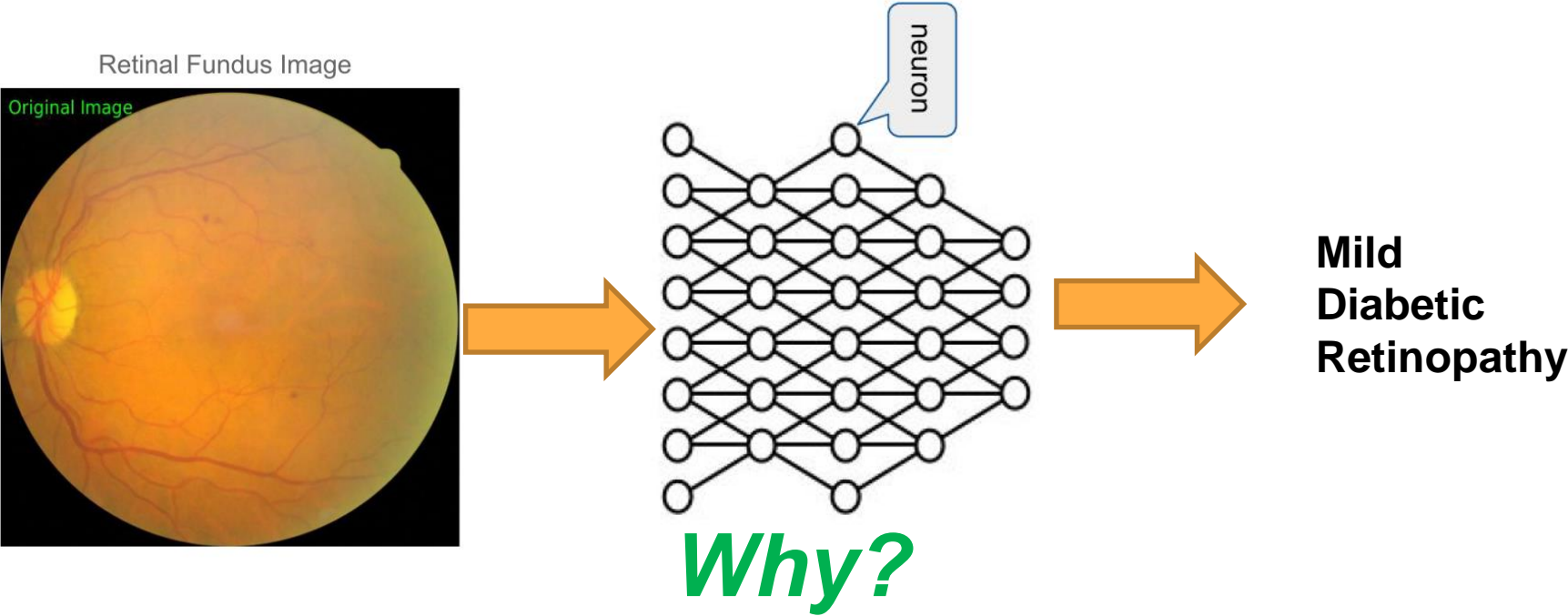
21st January, 2020

CEP on NLP

IIT Patna

(Acknowledgement: help from Kevin, Prashaant)

Motivation



Acknowledgment:
http://theory.stanford.edu/~ataly/Talks/sri_attribution_talk_jun_2017.pdf

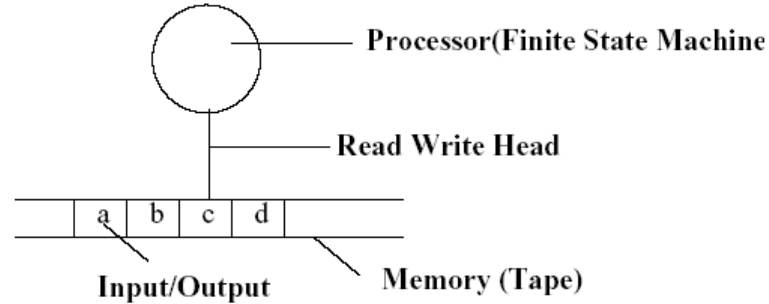
AI is actually a celebration of Natural Intelligence!!!

Every AI triumph throws a bigger challenge and makes us bow to Him who created *our* intelligence

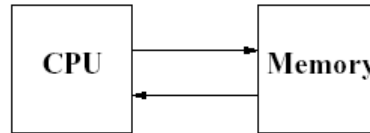
Just like every hillock scaled reveals a hill, and behind the hill is a mountain that makes us humble



Turing Machine & Von Neumann Machine



Turing machine



VonNeumann Machine

3 Models of computation: equivalence proved in 60s

- Turing Machines (Programming by humans)
- Neural Nets (programming by data aka ML; had to wait for 50 years for adequate data to become available)
- Recursively Enumerable Sets (A set of integers is said to be recursively enumerable if there exists a **recursive function** that can eventually generate any element in it. Limited functionalities; did not go forward)

Notion of “natural” problems: what is good for TM, what is good for NN?

- TM needs programming by humans
- Human understanding of the problem sets the limit
- Error or omission (Precision) and Error of commission (Recall)
- Problems that admit 0 error of omission and commission by human judgement are good for TM (e.g., mathematical and geometrical problems: sorting, convex hull finding, numerical analysis etc.)
- If we cannot put a “boundary” for the **scope** of a problem, we have to **resort to data** (e.g., machine translation, sentiment analysis etc.)

Let us not forget the historical perspective on Machine Learning

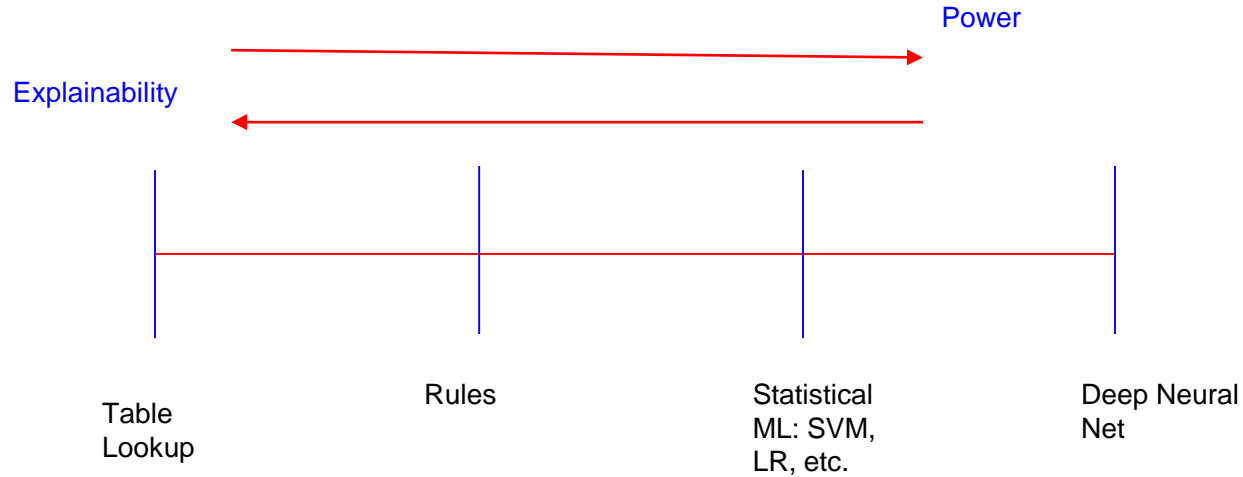
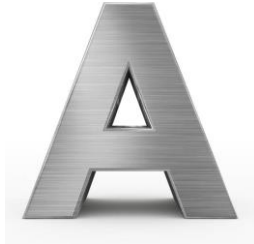


Table Look up



shutterstock.com • 1017846265

How many to store?

What is the essential
“Aness”?

Rules

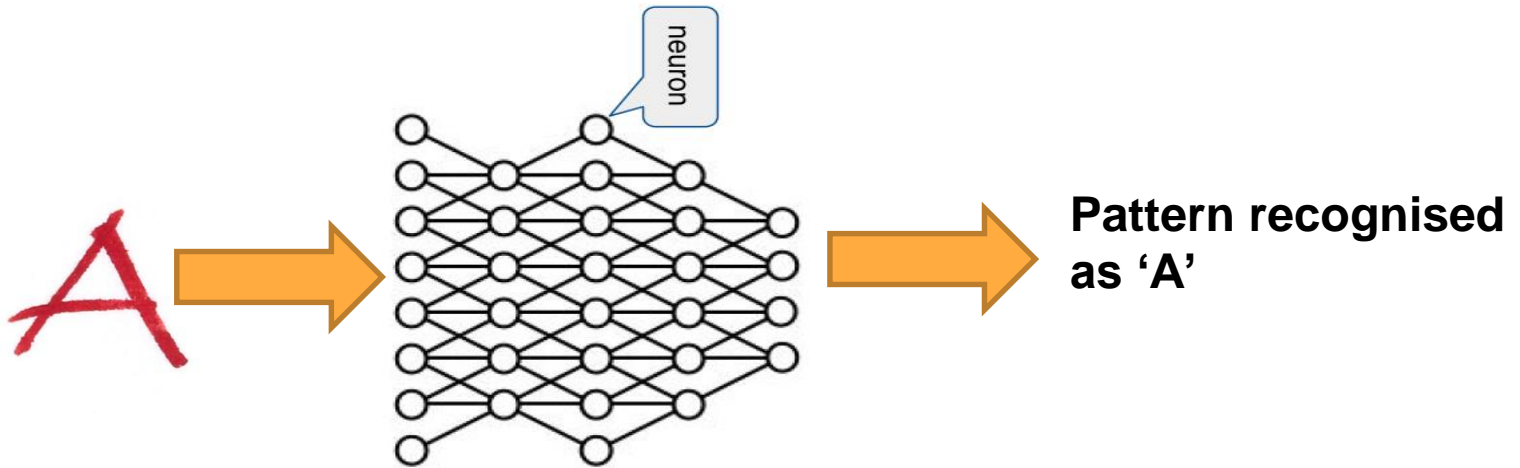
- Letter 'A' is formed from two inclined straight lines meeting at a point with a horizontal line cutting across
- Exception: need not be straight lines; need not meet; the 3rd line need not be horizontal
- Leads to false negative- ERROR OF OMISSION
- Relax condition and have false positive- ERROR OF COMMISSION

So LEARN....!

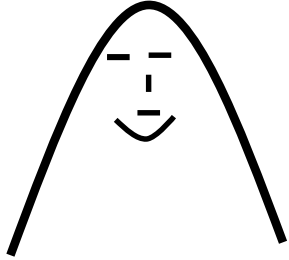
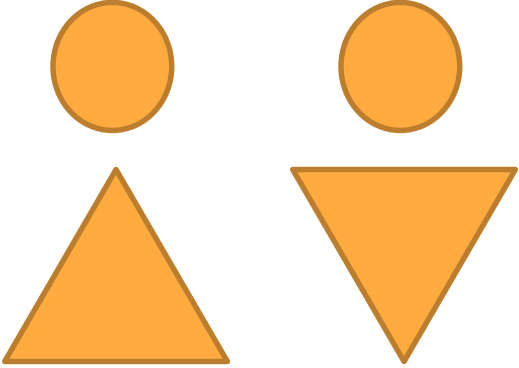
- Have Data
- Have classifier
- With LOTS of data, learn with
 - High precision (small possibility of error of commission)
 - High recall (small possibility of error of omission)
- Many practical applications have **empirically established** superiority of learning over rule engineering
- But depends on human engineered features, i.e., capturing essential properties

Reduce human dependency: DEEP LEARN

- End to end systems; essential properties learnt at intermediate layers



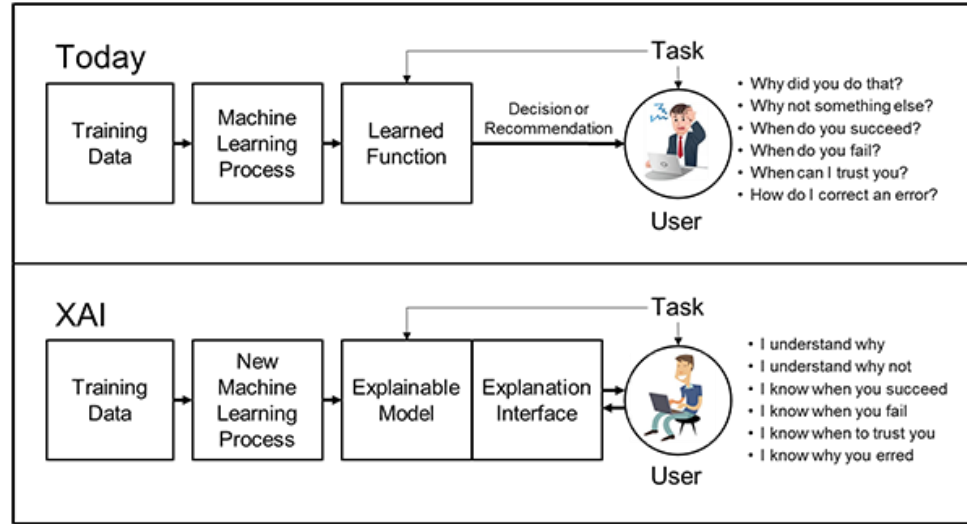
We are extremely good at understanding symbols (Restroom situation)



shutterstock.com • 776460166

Goal of XAI

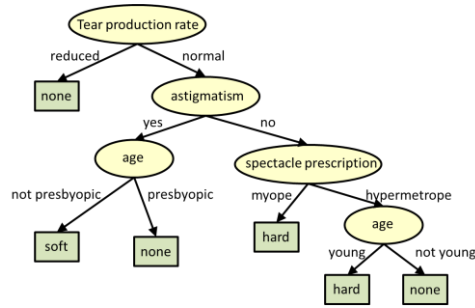
- Explainable AI - need of the hour
- Deep Learning models are mainly black boxes
- Discuss terminology, different formulations of explainability and some techniques



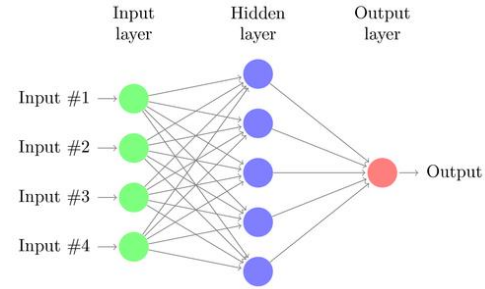
XAI (Gunning, 2017)

Machine Learning Models and Their Explainability

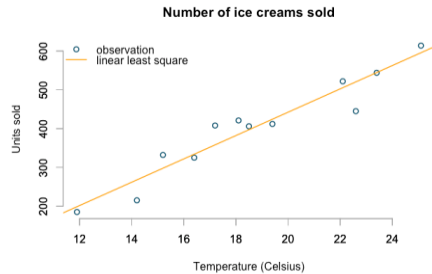
Interpretable Models



Black Boxes

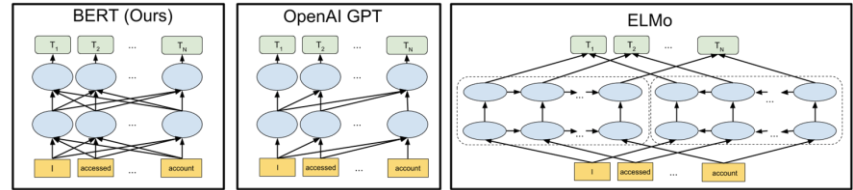


Decision Trees



Linear Models

Simple Neural Networks



Complex Neural Networks

What can Explainability do?

- Increase Trust
 - Does the model know when it is uncertain?
 - Does the model make the same mistakes as a human would in its stead?
 - Are we comfortable replacing a human with the model?
- Reveal Causality
 - What can the model tell us about the natural world?
 - Generally, supervised models are trained to make predictions, but are used to make decisions and take actions accordingly
 - They mainly learn **correlation**, and not cause

Correlation is ALL IMPORTANT! Actually it is more of co-occurrence

- Sentiment: *This is movie is amazing-* decision- **positive** sentiment, strong correlation with '**amazing**'; occurs together frequently
- Machine Translation: *This river bank is crowded* → इस नदी तट पर भीड़ है (is nadii tat par bhid hai)- strong correlation of **river** with **translation of bank as tat**
- Named entity identification (NEI): *vikas paD rahaa hai* → vikas_PN; *bhaarat kaa vikas teji se ho rahaa hai* → vikas_NN (PN means proper noun, NN means ordinary noun)- strong correlation of **vikas_PN** with **paD rahaa hai**; **vikas_NN** with **bhaarat kaa**
- Question answer: Q: *What is the capital of Kenia?* → A: Nairobi- strong correlation of **Nairobi** in corpus with **Kenia** and **capital** (all three occur together frequently)

Causality is still a far cry...

- Chemists, biologist, material scientists are not satisfied with ONLY correlation
- Medical text mining reveals co-occurrences, thereby correlation
- Jaundice is frequently associated with yellow eyes
- Observed clinically, SEEN in medical texts too
- But biologists what to know IS YELLOW EYE THE CAUSE OF JAUNDICE?
- And then long regress: *why is yellowness in eye caused by jaundice?*
- Chain of WHYs can be long, and current explainability is far from that state of competence

Jumping the gun: Investigations into Causality

Problem Statement

- Extracting causal relationships in natural languages.
- To differentiate between **causal** and **correlated** factors.



Eg. Ram and Shyam were **happy** after they **won** the football match.

Causality Vs Correlation

- Correlation + X = Causality
 - X is (cause → effect) relation

Eg.

1. People who **smoke** are likely to suffer from **alcoholism**.
2. People who **smoke** have high chances of **cancer**.

Causality Theories

- Counterfactual theories:
- Probabilistic causation:
- Causal calculus
- Structure learning

Causality Experiment-1

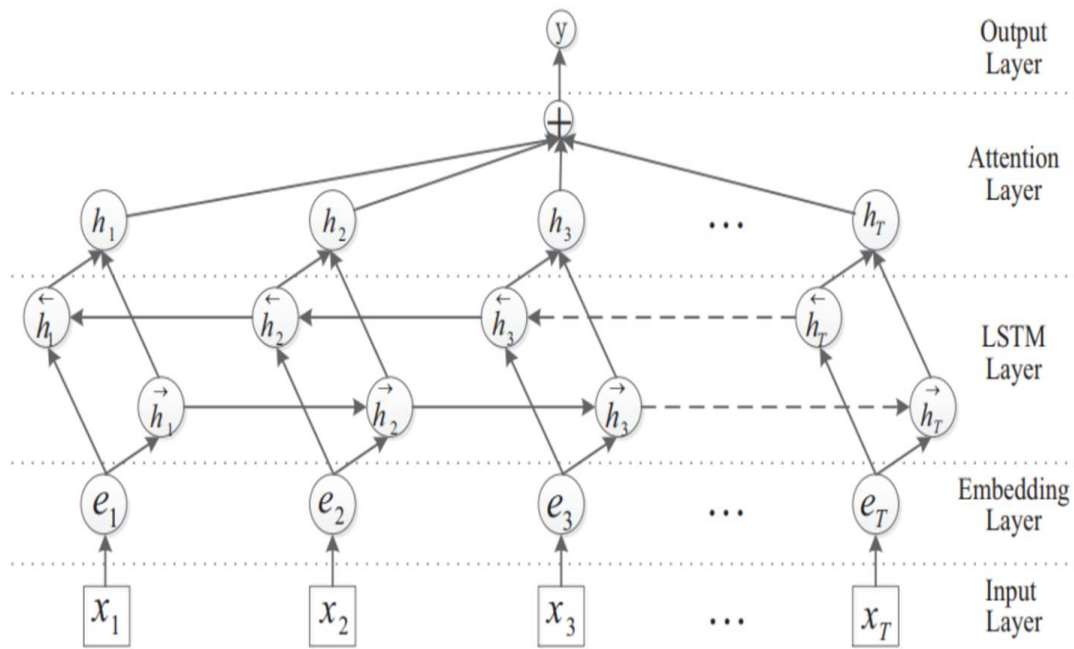
- *Multi-way classification of Relation between pair of entities*
 - SemEval 2010 Task 8
- *Dataset Example:*
 - "We estimate a wind speed associated with the **<e1>devastation</e1>** caused by the **<e2>tornado</e2>**."
Cause-Effect(e2,e1)
- Dataset Statistics

Relation	Train Data	Test Data	Total Data
Cause-Effect	1,003 (12.54%)	328 (12.07%)	1331 (12.42%)
Instrument-Agency	504 (6.30%)	156 (5.74%)	660 (6.16%)
Product-Producer	717 (8.96%)	231 (8.50%)	948 (8.85%)
Content-Container	540 (6.75%)	192 (7.07%)	732 (6.83%)
Entity-Origin	716 (8.95%)	258 (9.50%)	974 (9.09%)
Entity-Destination	845 (10.56%)	292 (10.75%)	1137 (10.61%)
Component-Whole	941 (11.76%)	312 (11.48%)	1253 (11.69%)
Member-Collection	690 (8.63%)	233 (8.58%)	923 (8.61%)
Message-Topic	634 (7.92%)	261 (9.61%)	895 (8.35%)
Other	1,410 (17.63%)	454 (16.71%)	1864 (17.39%)
Total	8,000 (100.00%)	2,717 (100.00%)	10,717 (100.00%)

Causality Experiment-1: Architecture

Architecture

- BiLSTM + Attention



Results

- Test P **0.8213** | R **0.8563** |
macro_F1: **0.8362**
- Test test_max_f1_final: **0.8362**

Confusion Matrix

	Predicted										SUM
	CE	CW	CC	ED	EO	IA	MC	MT	PP	O	
CE	301	0	0	0	7	0	0	1	2	15	326
CW	1	250	5	2	1	9	8	6	1	22	305
CC	0	0	175	5	2	0	1	0	0	6	189
ED	0	2	7	270	0	0	0	1	0	11	291
EO	3	0	2	3	239	2	0	1	2	6	258
IA	0	3	0	2	3	122	0	0	5	20	155
MC	0	4	0	2	2	0	214	1	0	10	233
MT	0	0	0	1	3	0	1	248	0	7	260
PP	4	3	1	3	5	6	1	3	192	12	230
O	10	25	28	25	32	15	40	39	24	216	454
SUM	319	287	218	313	294	154	265	300	226	325	2701

CE	Cause-Effect	IA	Instrument-Agency
CW	Component-Whole	MC	Member-Collection
CC	Content-Container	MT	Message-Topic
ED	Entity-Destination	PP	Product-Producer
EO	Entity-Origin	O	Other

Causality Experiments: Data Bottleneck

Problem

- Only 1003 sentences containing Cause-Effect Relationship
 - TACRED is a large-scale relation extraction dataset with 106,264 examples with 41 relations from Stanford
- Biased dataset: Most of causal relationships indicated by “Caused by”, “Causes” (root word: cause)
 - The best kept secret for avoiding abdominal <e1>weight gain</e1> due to <e2>stress</e2> is the use of adaptogens.

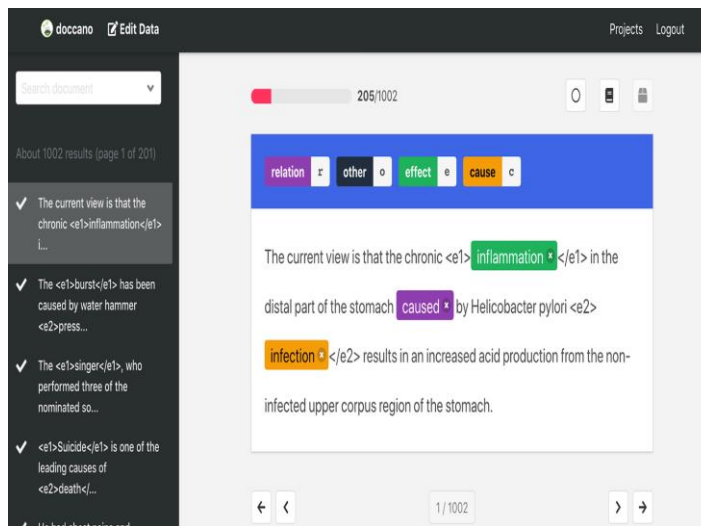
Cause-Effect(e2,e1)

- Nested + multiple causal relationship are not captured
- <e1>Sun</e1> and wind cause <e2>evaporation</e2> of water, causing rains, and this energy can be caught using hydroelectric power.

Solution

Data Annotation Tool

- Auto-labeling (ongoing)
 - Weak-Supervision (eg. Snorkel: StanfordNLP)



Deployed at:
<https://textannotation.herokuapp.com>

Causality Experiments: FewRel Dataset + BERT Based Models

- The Few-Shot Relation Classification Dataset (FewRel)
- This dataset consists of 70K sentences expressing 100 relations annotated by crowdworkers on Wikipedia corpus
 - Can causal relationship be inferred from combination of other relationship present in the text?
 - Till now focus on only causal relationship , To answer this question need to study various kinds of relation in text and how they relate to each other ?

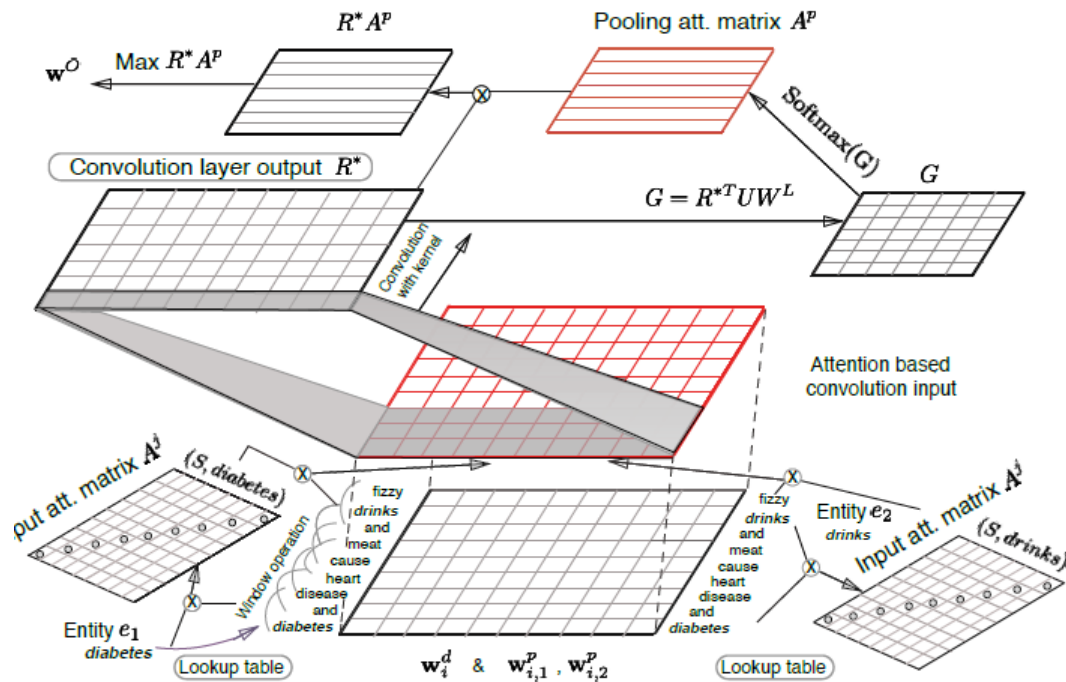
Matching the Blanks: Distributional Similarity for Relation Learning (Baldini Soares et al., 2019)

- No code released
- BERT based models takes days to train/finetune
- Claim that, it gives state of the art on FewRel Dataset without any fine tuning (Generalized relation extractor)

Causality Experiment-2 [Ongoing]

Architecture

- Multi Level CNN + Attention



Back to explainability: What can Explainability do? (contd.)

- Demonstrate Transferability
 - Machine learning model trained in a controlled setting
 - Will it perform in a similar fashion when deployed?
 - In other words, has the model truly learnt to detect underlying phenomenon or is it mimicking the artifacts of the training data?
 - Need sanity checks
- Show Informativeness
 - A model may be trained to make a decision
 - But it could also be used to aid a person in making a decision
 - Can it provide useful information of this kind?

How does Explainability help Data Scientists?

- Identify and mitigate biases
 - All models are biased
 - Cannot eliminate biases completely; reduce them
- Account for context
 - Models cannot account for all the factors that will affect the decision
 - Explainability helps understand the included factors so that one can adjust prediction on additional factors
- Extract knowledge
 - Identify whether learned patterns are true phenomenon or artifacts of the dataset

Desirable Properties of an Explanation

Desirable properties of explanations about individual predictions, without necessarily describing the process for calculating them

- Concise
- Faithful
- Complete
- Comparable
- Global
- Consistent
- Engaging

Desirable Properties of an Explanation (contd.)

- **Concise**
 - An explanation should not overwhelm the user
 - Reason we use ML is to delegate the complexity which our mental models cannot handle
 - An explanation should minimize the cognitive load required of the user
- **Faithful**
 - Explanation should accurately describe the way the model made the prediction
 - Global surrogate models are not faithful
- **Complete**
 - An explanation is complete if it explains all the factors and elements that went into a prediction
 - Trade-off exist between concise and complete

Desirable Properties of an Explanation (contd.)

- **Comparable**

- An explanation should help one compare different models to each other by examining how they handle individual examples
- Handled well by model-agnostic techniques, not so well by model-dependent techniques

- **Global**

- An explanation should indicate how each individual prediction fits into the overall structure of the model
- Understanding how the overall model works
- The global explanation only has to indicate enough of the model's structure to provide reasonable context for each individual prediction

Desirable Properties of an Explanation (contd.)

- Consistent

- An explanation algorithm is consistent if each successive explanation helps the user to better understand later predictions
- The user should never perceive a contradiction between different explanations

- Engaging

- An explanation should encourage a user to pay attention to the important details
- Users who pay more attention to explanations become familiar with the model faster and ultimately make better decisions
- Seems like a User Experience (UX) issue

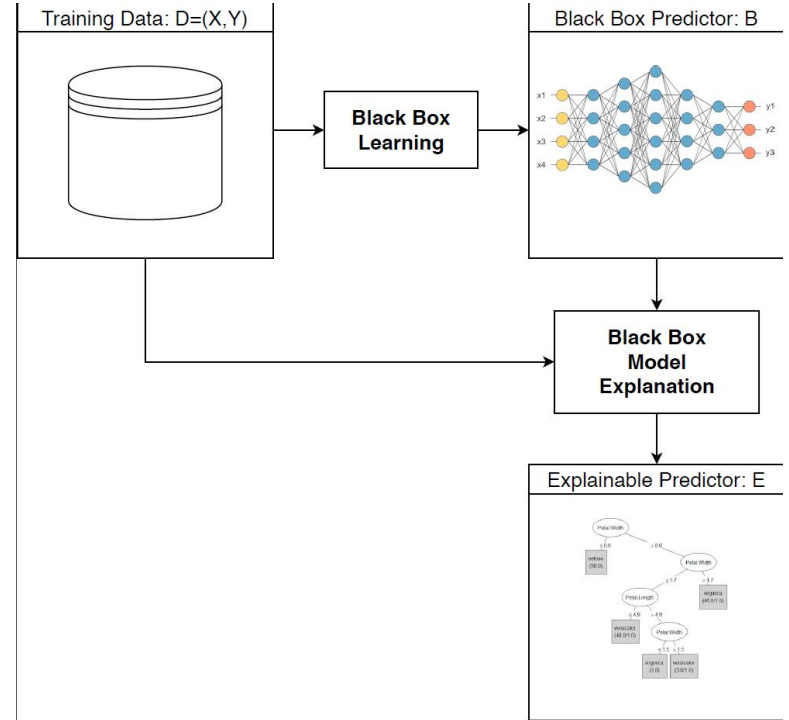
Black Box Explanation : Problem Formulation

Guidotti et. al. (2018) provides the following formulations for model explanations:

- Black Box Model Explanation
- Black Box Outcome Explanation
- Black Box Inspection
- Transparent Box Design

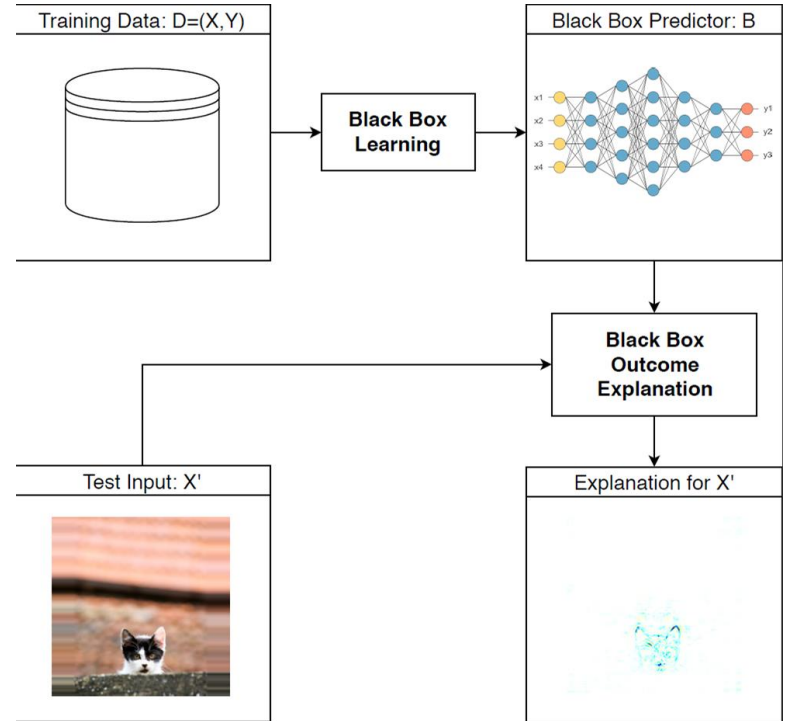
Black Box Model Explanation

- Create surrogate model
- NN \rightarrow DT
- Relatively old paradigm, not much used



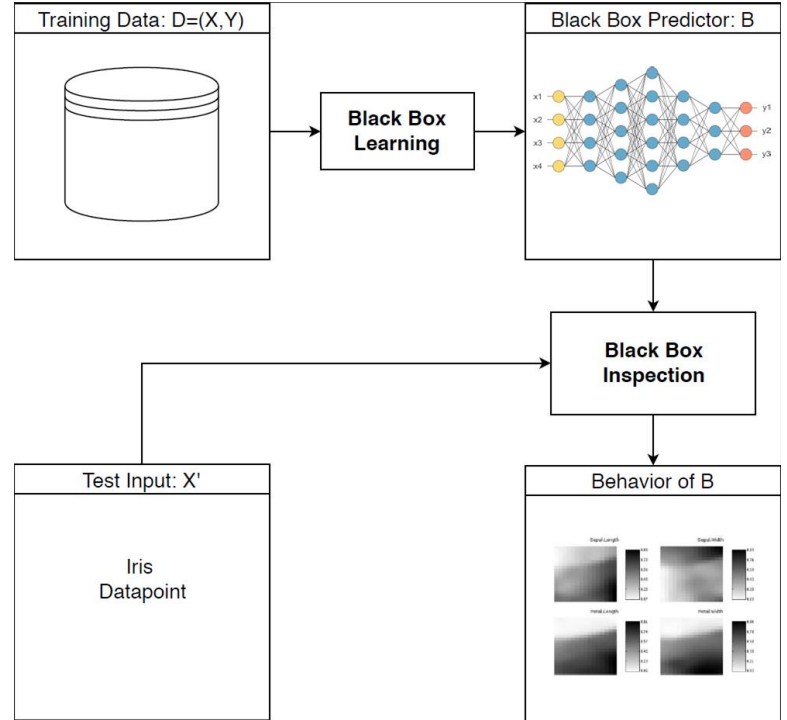
Black Box Outcome Explanation

- Explain behavior for a particular instance
- I.e., explain outcome of a particular instance
- Popular in the current neural network setting



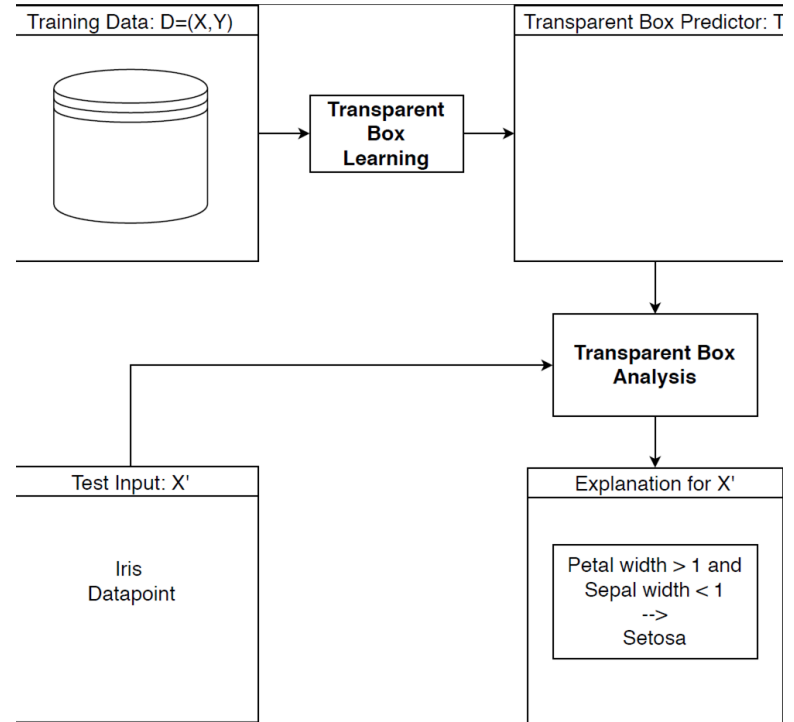
Black Box Inspection

- Inspect attributes
- How does the outcome change if we increase feature x_i
- Popular currently



Transparent Box Design

- Directly use a decision tree to solve given problem
- Often transparency at the cost of power!



Explain by observing the “Training”: How Does a DNN Train?

Morning shows the DAY; Child is the Father of the Man

Two models:

- (a) Perceptron and its Training,
- (b) Feedforward Network and Backpropagation

Perceptron training

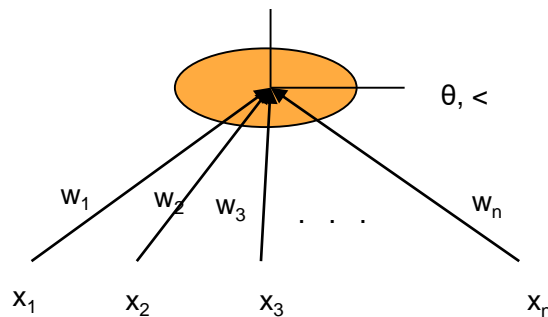
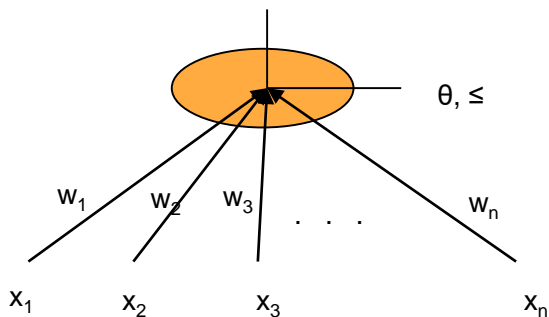
Perceptron Training Algorithm (PTA)

Preprocessing:

1. The computation law is modified to

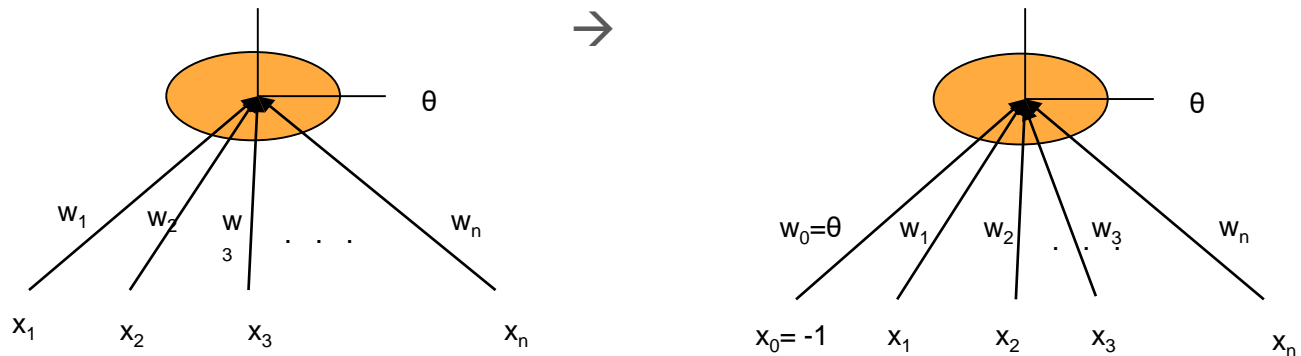
$$y = 1 \text{ if } \sum w_i x_i > \theta$$

$$y = 0 \text{ if } \sum w_i x_i < \theta$$



PTA – preprocessing cont...

2. Absorb θ as a weight



3. Negate all the zero-class examples

Perceptron Training Algorithm

1. Start with a random value of w
ex: $\langle 0, 0, 0 \dots \rangle$
2. Test for $w x_i > 0$
If the test succeeds for $i=1, 2, \dots, n$
then return w
3. Modify w , $w_{\text{next}} = w_{\text{prev}} + x_{\text{fail}}$

NAND on Perceptron

NAND Augmented:				NAND-0 class Negated			
X2	X1	X0	Y		X2	X1	X0
0	0	-1	1	V0:	0	0	-1
0	1	-1	1	V1:	0	1	-1
1	0	-1	1	V2:	1	0	-1
1	1	-1	0	V3:	-1	-1	1

W : $\langle W_2 \ W_1 \ W_0 \rangle$ has to be found such that $W \cdot V_i > 0$

PTA Algo a few steps

Algorithm:

Initialize and Keep adding the failed vectors
until $W \cdot V_i > 0$ is true.

$$\begin{aligned}\text{Step 0: } W &= \langle 0, 0, 0 \rangle \\ W_1 &= \langle 0, 0, 0 \rangle + \langle 0, 0, -1 \rangle \quad \{V_0 \text{ Fails}\} \\ &= \langle 0, 0, -1 \rangle \\ W_2 &= \langle 0, 0, -1 \rangle + \langle -1, -1, 1 \rangle \quad \{V_3 \text{ Fails}\} \\ &= \langle -1, -1, 0 \rangle \\ W_3 &= \langle -1, -1, 0 \rangle + \langle 0, 0, -1 \rangle \quad \{V_0 \text{ Fails}\} \\ &= \langle -1, -1, -1 \rangle \\ W_4 &= \langle -1, -1, -1 \rangle + \langle 0, 1, -1 \rangle \quad \{V_1 \text{ Fails}\} \\ &= \langle -1, 0, -2 \rangle\end{aligned}$$

Continuing this way:

$$W15 = \langle -2, -1, -4 \rangle + \langle -1, -1, 1 \rangle \quad \{\text{V3 Fails}\}$$

$$= \langle -3, -2, -3 \rangle$$

$$W16 = \langle -3, -2, -3 \rangle + \langle 1, 0, -1 \rangle \quad \{\text{V2 Fails}\}$$

$$= \langle -2, -2, -4 \rangle$$

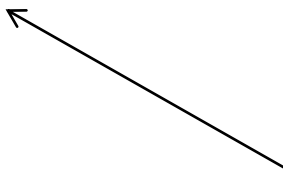
$$W17 = \langle -2, -2, -4 \rangle + \langle -1, -1, 1 \rangle \quad \{\text{V3 Fails}\}$$

$$= \langle -3, -3, -3 \rangle$$

$$W18 = \langle -3, -3, -3 \rangle + \langle 0, 1, -1 \rangle \quad \{\text{V1 Fails}\}$$

$$= \langle -3, -2, -4 \rangle$$

$$W2 = -3, \quad W1 = -2, \quad W0 = \Theta = -4$$



Succeeds for all vectors

FFNN and BP

Gradient Descent Technique

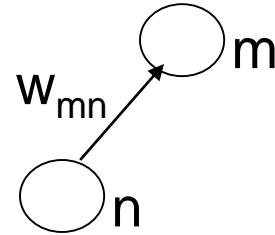
- Let E be the error at the output layer

$$E = \frac{1}{2} \sum_{j=1}^p \sum_{i=1}^n (t_i - o_i)_j^2$$

- t_i = target output; o_i = observed output
- i is the index going over n neurons in the outermost layer
- j is the index going over the p patterns (1 to p)
- Ex: XOR:– $p=4$ and $n=1$

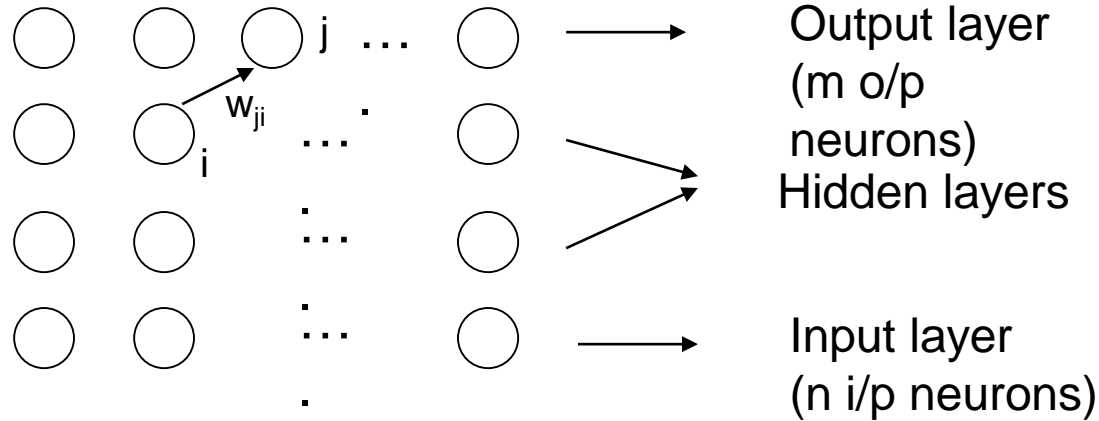
Weights in a FF NN

- w_{mn} is the weight of the connection from the n^{th} neuron to the m^{th} neuron
- E vs \underline{w} surface is a complex surface in the space defined by the weights w_{ij}
- $-\frac{\delta E}{\delta w_{mn}}$ gives the direction in which a movement of the operating point in the w_{mn} coordinate space will result in maximum decrease in error



$$\Delta w_{mn} \propto -\frac{\delta E}{\delta w_{mn}}$$

Backpropagation algorithm



- Fully connected feed forward network
- Pure FF network (no jumping of connections over layers)

Gradient Descent Equations

$$\Delta w_{ji} = -\eta \frac{\delta E}{\delta w_{ji}} \quad (\eta = \text{learning rate}, 0 \leq \eta \leq 1)$$

$$\frac{\delta E}{\delta w_{ji}} = \frac{\delta E}{\delta net_j} \times \frac{\delta net_j}{\delta w_{ji}} \quad (net_j = \text{input at the } j^{\text{th}} \text{ layer})$$

$$\frac{\delta E}{\delta net_j} = -\delta_j$$

$$\Delta w_{ji} = \eta \delta_j \frac{\delta net_j}{\delta w_{ji}} = \eta \delta_j o_i$$

Backpropagation – for outermost layer

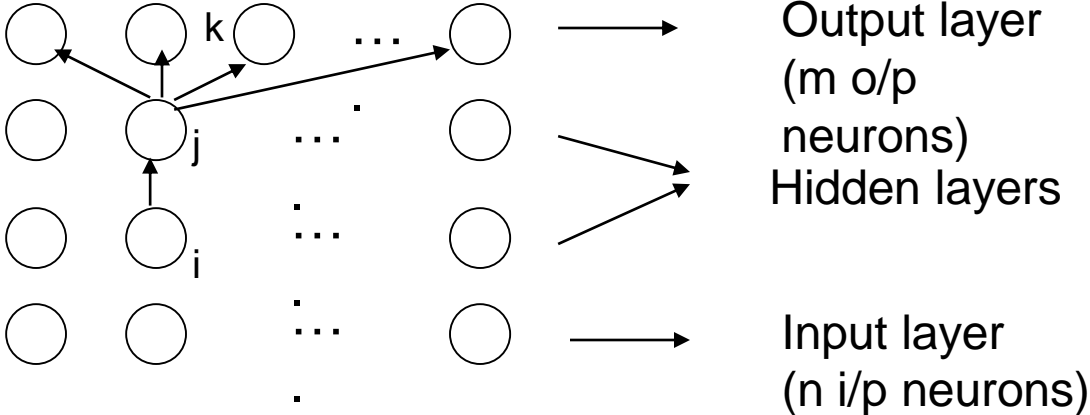
$$\delta_j = -\frac{\delta E}{\delta net_j} = -\frac{\delta E}{\delta o_j} \times \frac{\delta o_j}{\delta net_j} \text{ (} net_j \text{ = input at the } j^{th} \text{ layer)}$$

$$E = \frac{1}{2} \sum_{p=1}^m (t_p - o_p)^2$$

$$\text{Hence, } \delta_j = -(-(t_j - o_j)o_j(1 - o_j))$$

$$\Delta w_{ji} = \eta(t_j - o_j)o_j(1 - o_j)o_i$$

Backpropagation for hidden layers



δ_k is propagated backwards to find value of δ_j

Backpropagation – for hidden layers

$$\Delta w_{ji} = \eta \delta_j o_i$$

$$\delta_j = -\frac{\delta E}{\delta net_j} = -\frac{\delta E}{\delta o_j} \times \frac{\delta o_j}{\delta net_j}$$

$$= -\frac{\delta E}{\delta o_j} \times o_j(1-o_j)$$

This recursion can
give rise to vanishing
and exploding
Gradient problem

$$= -\sum_{k \in \text{next layer}} \left(\frac{\delta E}{\delta net_k} \times \frac{\delta net_k}{\delta o_j} \right) \times o_j(1-o_j)$$

$$\text{Hence, } \delta_j = -\sum_{k \in \text{next layer}} (-\delta_k \times w_{kj}) \times o_j(1-o_j)$$

$$= \sum_{k \in \text{next layer}} (w_{kj} \delta_k) o_j(1-o_j)$$

- General weight updating rule:

$$\Delta w_{ji} = \eta \delta_j o_i$$

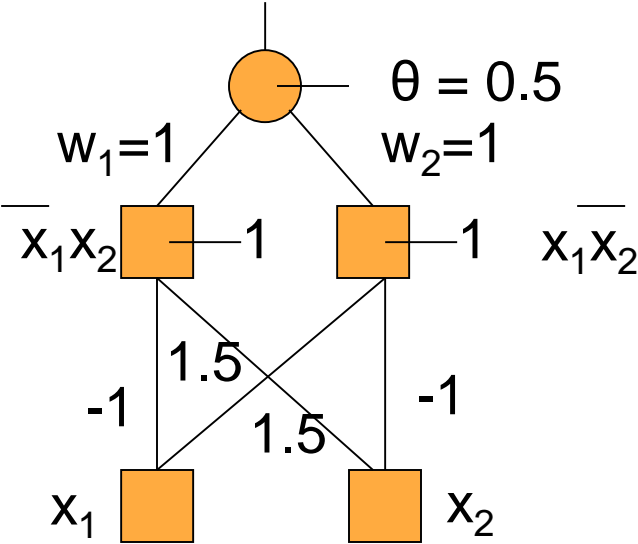
- Where

$$\delta_j = (t_j - o_j) o_j (1 - o_j) \quad \text{for outermost layer}$$

$$= \sum_{k \in \text{next layer}} (w_{kj} \delta_k) o_j (1 - o_j) o_i \quad \text{for hidden layers}$$

How does it work?

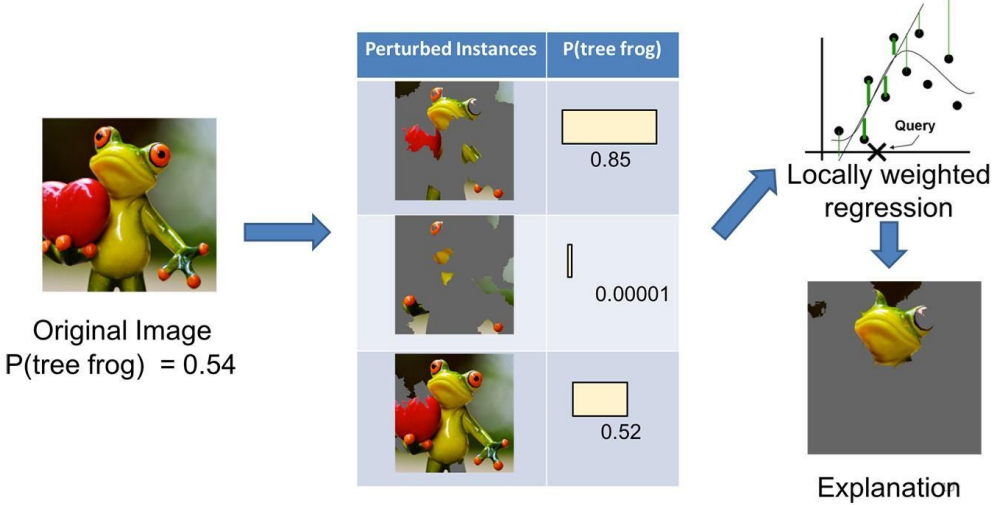
Input propagation forward and error propagation backward (e.g. XOR)



Back to explainability

Black Box Outcome Explanation Techniques

Feature Importance:
Perturbation based
methods

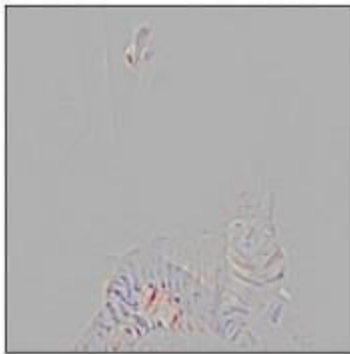


LIME (Ribeiro et. al., 2016)

Black Box Outcome Explanation Techniques (contd.)

Gradient based methods: Gradient-weighted Class Activation Mapping (**Grad-CAM**)

Guided Grad-CAM for "Cat"

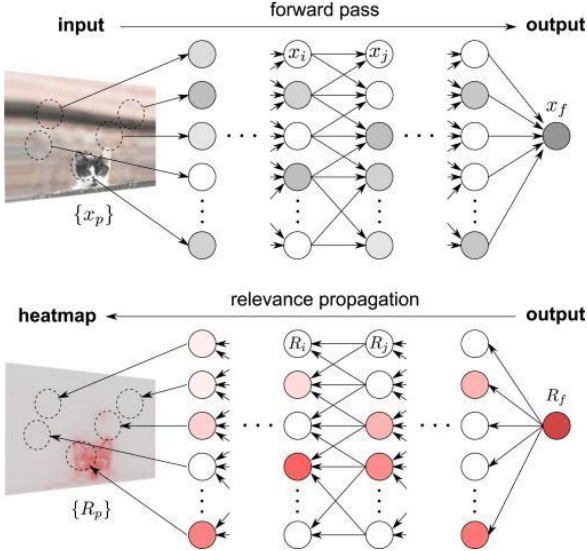


Guided Grad-CAM for "Dog"



Black Box Outcome Explanation Techniques (contd.)

Decomposition based methods



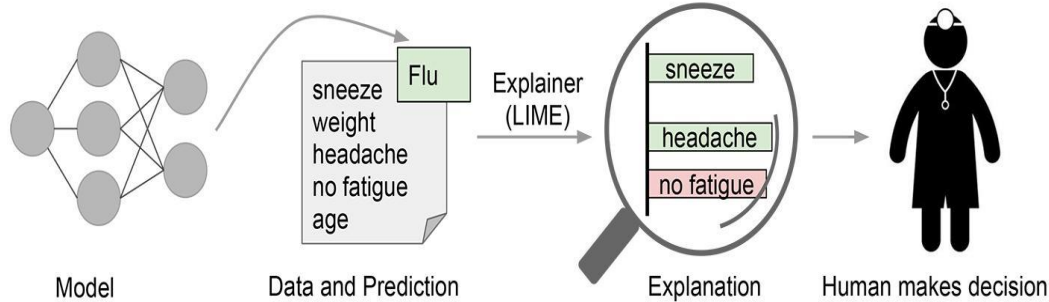
LRP (Bach et. al., 2015)

Black Box Outcome Explanation Techniques (contd.)

Prototype Explanations: Given an input, return as explanation a set of prototypical datapoints from the training data, a comparison with whom can explain the current decision (Kim et. al. 2016) (*“the movie was amazing”* has positive sentiment, because it is similar to *“the movie was excellent”* which is itself positive)

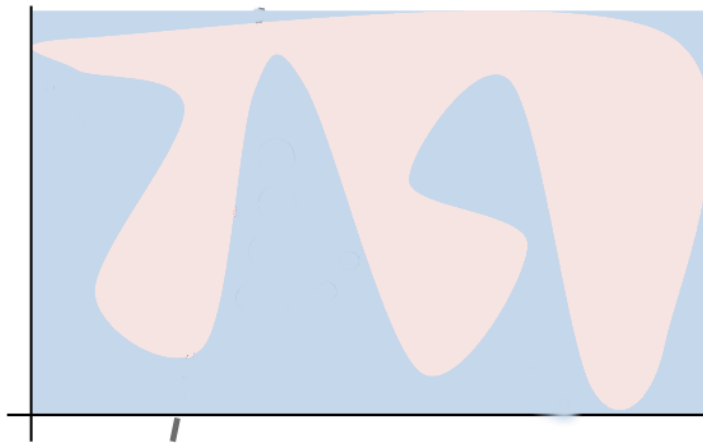
Contrastive Explanations: Given an input, return as explanation a set of features whose presence influenced the decision, and a set of features whose absence influenced the decision (Dhurandhar et. al. 2018)

Locally Interpretable Model Agnostic Explanations (LIME)



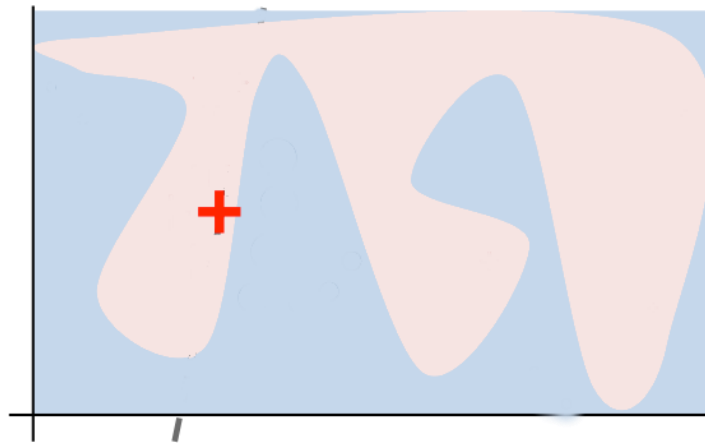
- By Riberio et. al. (2016)
- Gives feature importance explanations
- Learns a local surrogate model
- Is a model-agnostic explanation technique
 - Can work with different kinds of models
- Is perturbation based

LIME: Working



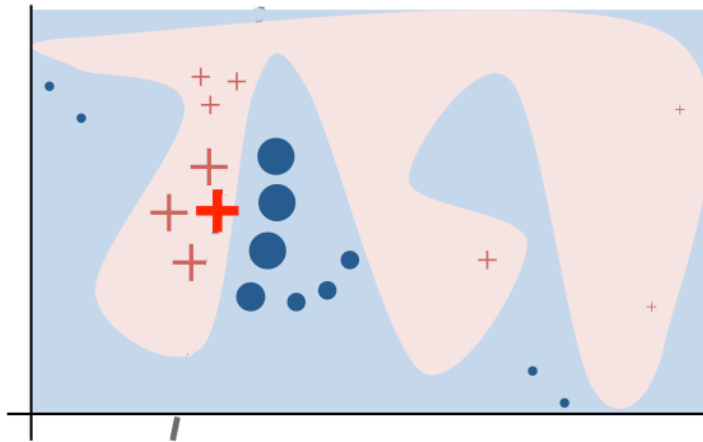
Consider a binary classifier with non-linear decision boundary

LIME: Working (contd.)



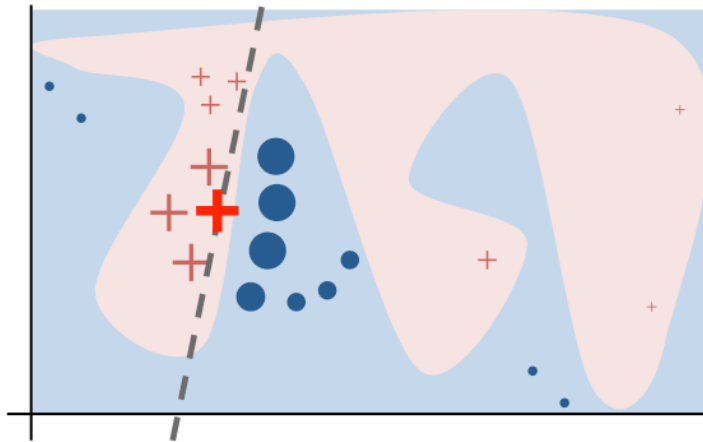
This particular datapoint has been assigned to the red class, and needs to be explained

LIME: Working (contd.)



LIME generates these additional samples, and uses the model to classify them

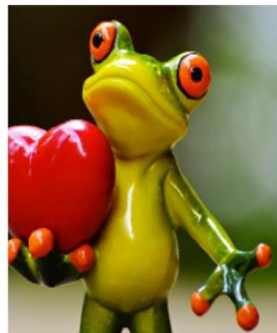
LIME: Working (contd.)



Finally it learns a classifier that is linear in the vicinity of the datapoint, and returns the weights of this classifier as feature importance explanation for the particular datapoint

LIME: Working (contd.)

- Works on Interpretable data representation, regardless of actual features used by the model
- Example: interpretable representation for text classification is a binary vector indicating presence or absence of a word, even though the classifier may use word embeddings



Original Image



Interpretable Components







LIME: Working (contd.)

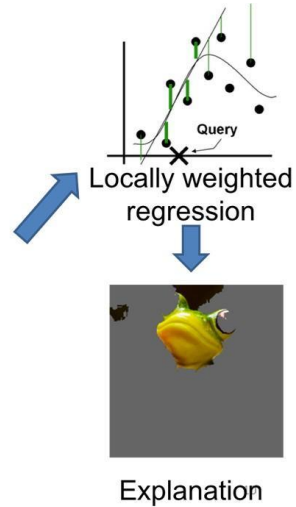
- Objective function aims to balance fidelity and interpretability
- LIME tries to minimize the locality aware loss, without making any assumption about f , since we want it to be model-agnostic.
- So L 's local behavior is approximated by drawing samples, weighted by similarity-kernel



Original Image
 $P(\text{tree frog}) = 0.54$



Perturbed Instances	$P(\text{tree frog})$
	 0.85
	 0.00001
	 0.52



LIME: Usage

- Python libraries exist
- Can be used with multiple models and libraries
- Relatively slow
- Sentiment Classification Demo: <https://sst5-explainer.herokuapp.com/>

Integrated Gradients (IG)

- By Sundarajan et. al. (2017)
- Gives feature importance explanations
- Uses sensitivity analysis
- Is a model-agnostic explanation technique
 - Can work with different kinds of models
- Is gradient based

IG: Axioms

- Sensitivity:
 - If every input differs from the baseline in exactly one feature and has different predictions, the differing feature is given a non-zero attribution
 - If the function (implemented by the model) does not depend (mathematically) on some variable in the input, then the attribution to that variable should always be zero
- Implementation Invariance:
 - Two networks are functionally equivalent if their outputs are equal for all inputs, despite very different structures
 - A method satisfies Implementation Invariance when the attributions are always identical for two functionally equivalent networks

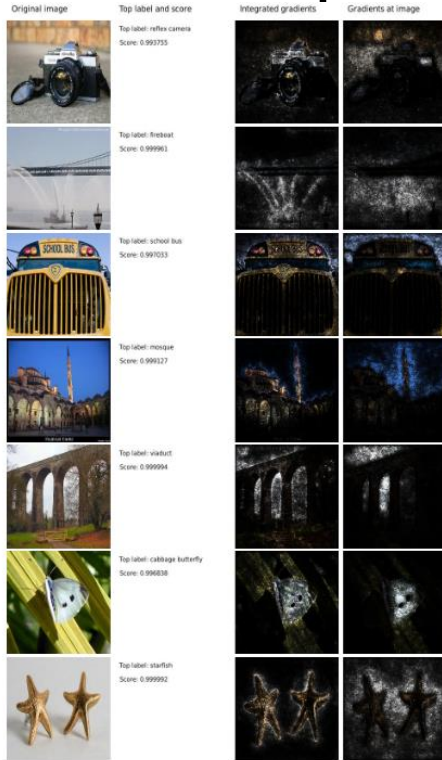
IG: Working

- Considers all points on a path between the current input and a baseline
- Computes prediction at each point
- Computes the gradients for each point with respect to the baseline
- Returns the weighted sum of these gradients as relevance assignment over the input space

$$e_i = (x_i - x'_i) \times \sum_{k=1}^m \frac{\partial F(x' + \frac{k}{m}(x - x'))}{\partial x_i} \times \frac{1}{m}$$

- \mathbf{x} - Input to be explained
- \mathbf{x}' - Baseline point
- \mathbf{m} - no. of points between \mathbf{x} and \mathbf{x}'
- \mathbf{F} - function learnt by the network
- \mathbf{e} - Final feature importance output
- \mathbf{j} - i th dimension of feature vector

IG: Example Usage



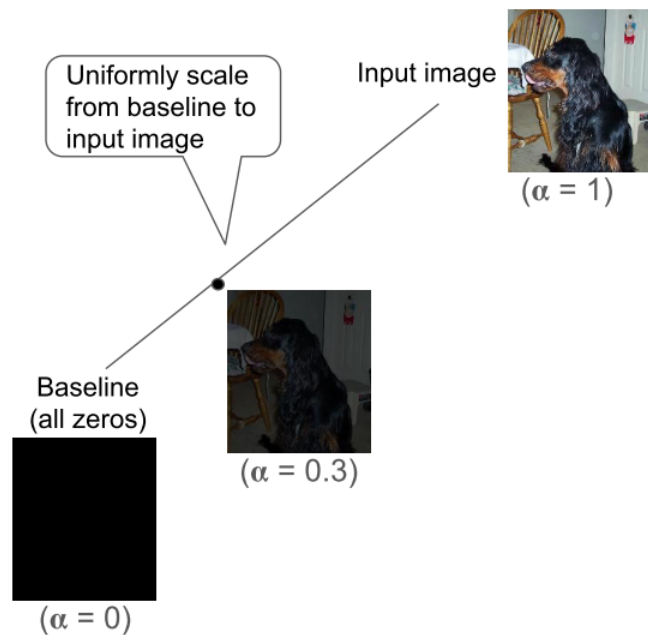
how many townships have a population above 50 ? [prediction: NUMERIC]
what is the difference in population between fora and masilo [prediction: NUMERIC]
how many athletes are not ranked ? [prediction: NUMERIC]
what is the total number of points scored ? [prediction: NUMERIC]
which film was before the audacity of democracy ? [prediction: STRING]
which year did she work on the most films ? [prediction: DATETIME]
what year was the last school established ? [prediction: DATETIME]
when did ed sheeran get his first number one of the year ? [prediction: DATETIME]
did charles oakley play more minutes than robert parish ? [prediction: YESNO]

IG Explanations for Question Classification. Red indicates positive contribution; blue indicates negative contribution; grey indicates neutral. Class is mentioned in square brackets

IG working

(CNTD. ack: http://theory.stanford.edu/~ataly/Talks/sri_attribution_talk_jun_2017.pdf)

The method: Integrated gradients



Mathematically,

$$IG_i(\text{image}) = \text{image}_i * \int_{0-1} \nabla F_i(\alpha * \text{image}) d\alpha$$

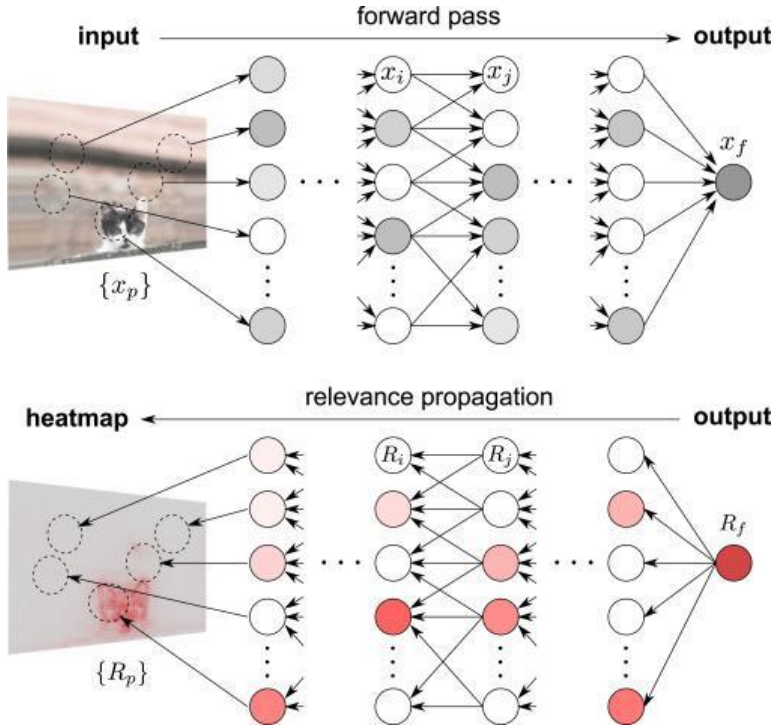
where:

- F is the prediction function for the label
- image_i is the intensity of the i^{th} pixel
- $IG_i(\text{image})$ is the integrated gradient w.r.t. the i^{th} pixel, i.e., **attribution for i^{th} pixel**

IG: Usage

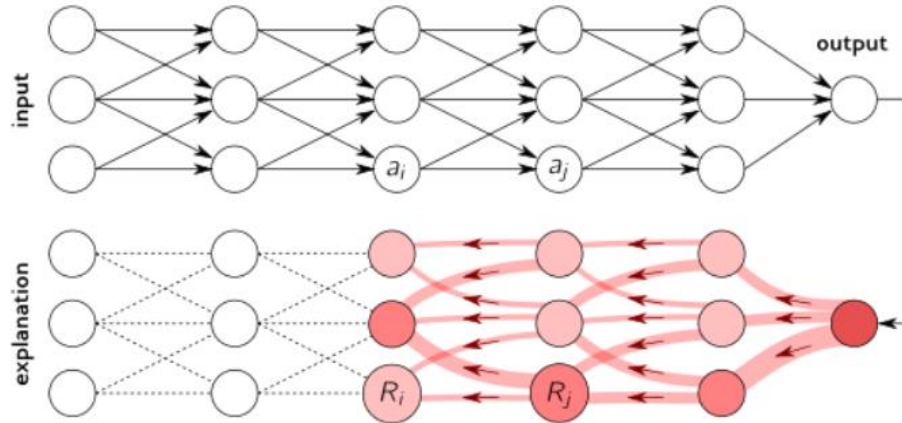
- No officially available code
- Simple logic
- Implementable easily using modern libraries
- Input-output dependent logic
- No need for network code instrumentation
- This also prevents one from investigating different layers of the network
 - Model-agnosticness is a double edged sword

Layerwise Relevance Propagation (LRP): Overview



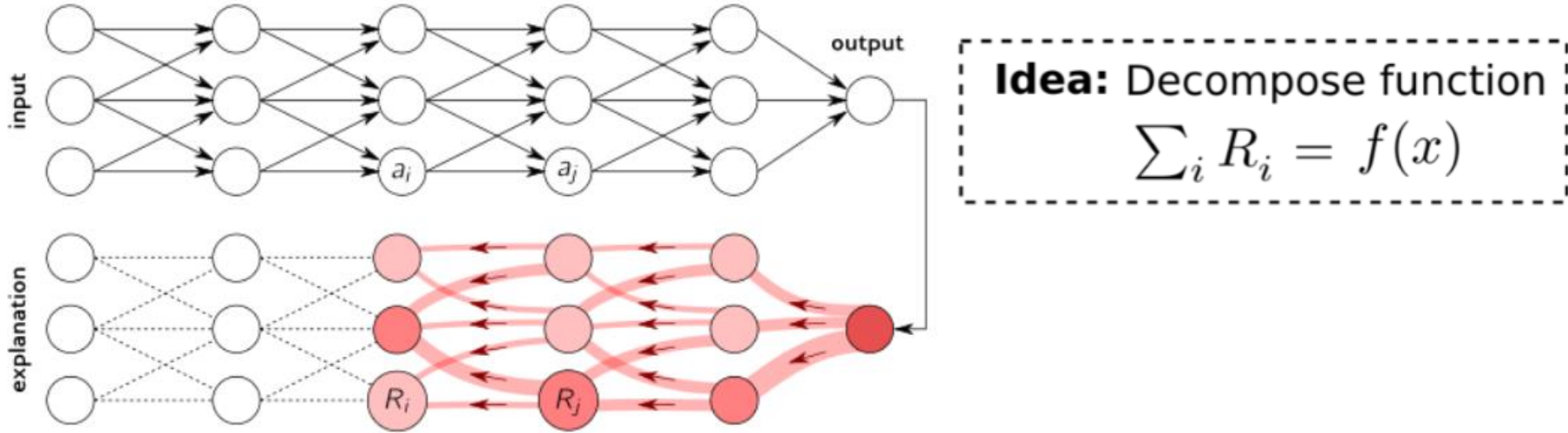
Layerwise Relevance Propagation (LRP)

- A model-dependent technique proposed by Bach et. al. (2015)
- Explains individual predictions by redistributing the final prediction output back in the network



- Assigns relevance scores to each input variable
 - Input neurons that contribute the most to higher layer get max relevance

Layerwise Relevance Propagation (LRP) (contd.)



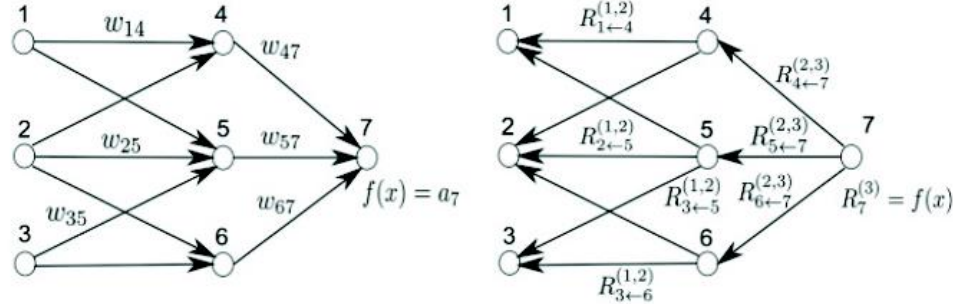
$$R_i = \sum_j \frac{a_j w_{ij}^+}{\sum_i a_i w_{ij}^+} R_j$$

Relevance Propagation Rule

Layer-wise relevance conservation

$$\sum_i R_i = \dots = \sum_i R_i^{(l)} = \sum_j R_j^{(l+1)} = \dots = f(x)$$

LRP: Working

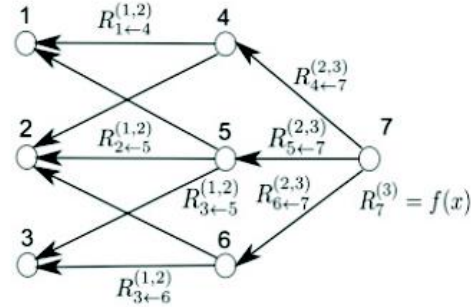
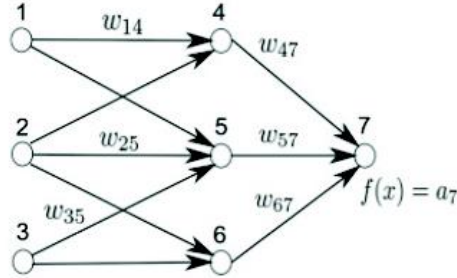


- Output is considered as relevance of final layer: $R_7^{(3)} = f(x)$
- Relevance conserved at each layer:

$$R_4^{(2)} + R_5^{(2)} + R_6^{(2)} = R_1^{(1)} + R_2^{(1)} + R_3^{(1)}$$

- Relevance for layer l comes from the layer $l+1$ via messages

LRP: Working



Two main equations

$$R_i^{(l)} = \sum_{k:i \text{ is input for neuron } k} R_{i \leftarrow k}^{(l,l+1)}$$

k:i is input for neuron k

$$R_{i \leftarrow k}^{(l,l+1)} = R_k^{(l+1)} \frac{a_i w_{ik}}{\sum_h a_h w_{hk}}$$

Example

$$(1) \quad R_3^{(1)} = R_{3 \leftarrow 5}^{(1,2)} + R_{3 \leftarrow 6}^{(1,2)}$$

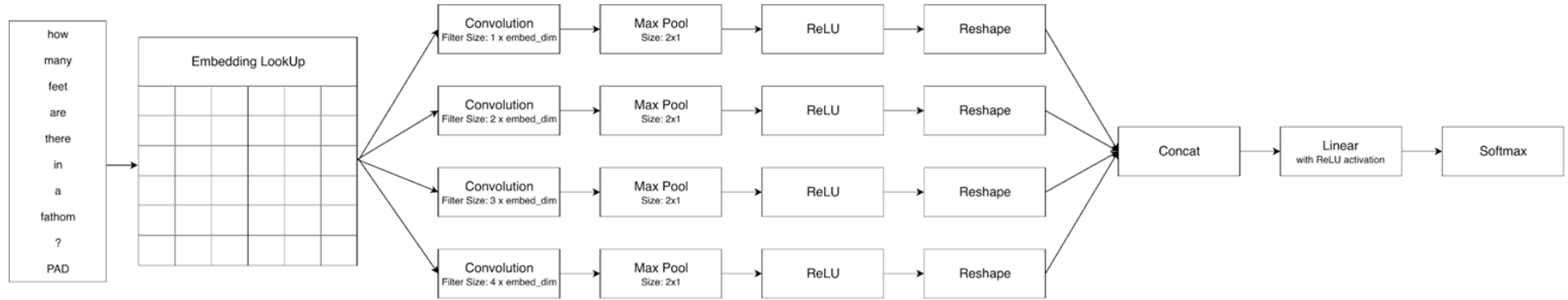
$$(2) \quad R_{3 \leftarrow 5}^{(1,2)} = R_5^{(2)} \frac{a_3 w_{35}}{\sum_h a_h w_{h5}}$$

A_i : activation of i^{th} neuron; l - layer

LRP: Usage

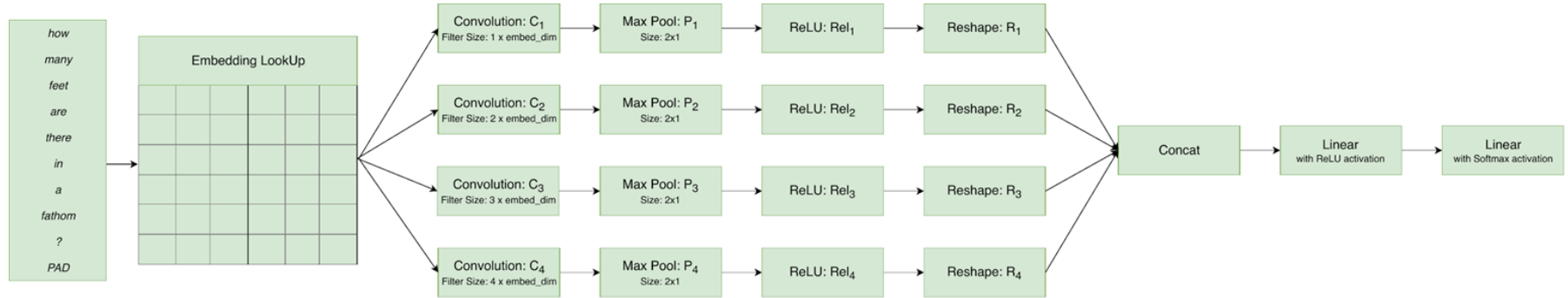
- Extra instrumentation needed in network code by practitioners
- Library specific packages released by developers
- Allows one to not only investigate relevance at input layers, but also at different intermediate layers

LRP: Model Specific Usage Example



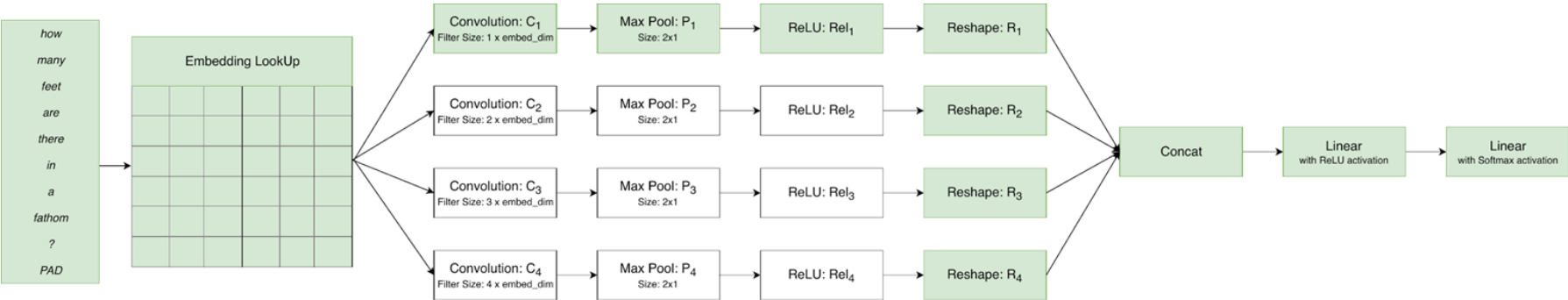
A Sample CNN Architecture for Question Classification

LRP: Model Specific Usage Example



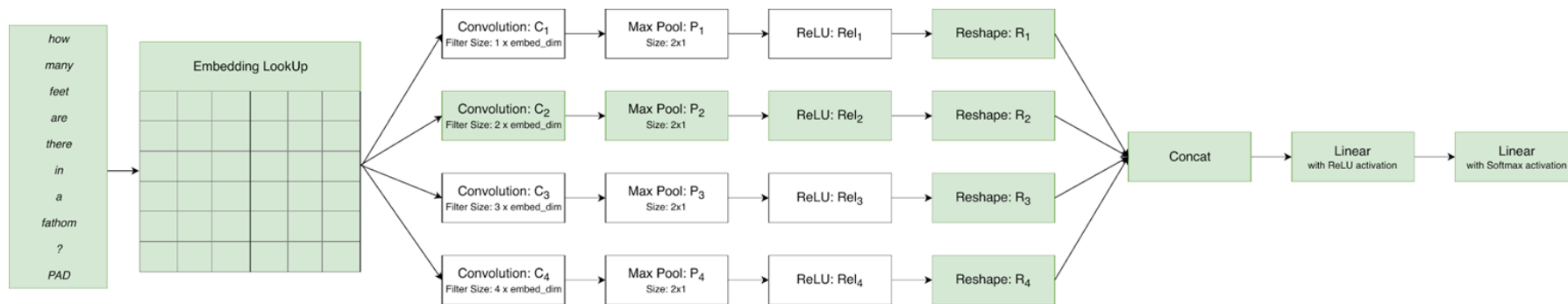
Overall Relevance Flow

LRP: Model Specific Usage Example



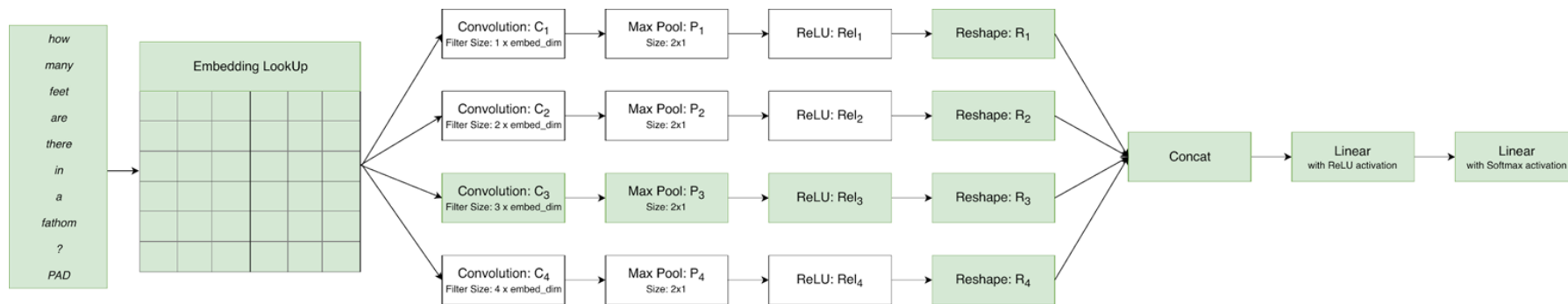
1-gram Relevance Flow

LRP: Model Specific Usage Example



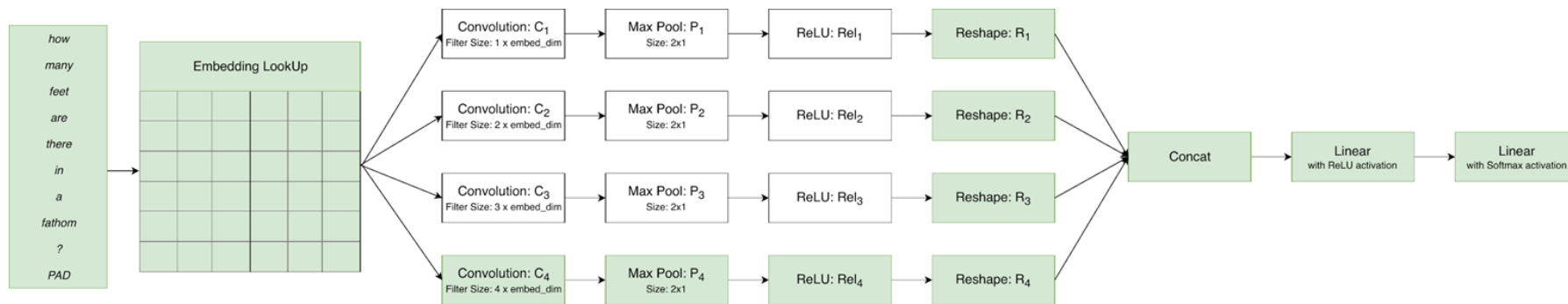
2-gram Relevance Flow

LRP: Model Specific Usage Example



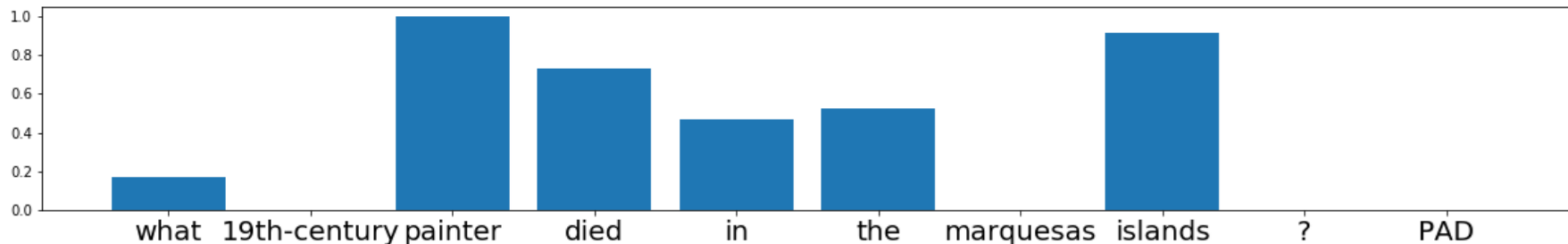
3-gram Relevance Flow

LRP: Model Specific Usage Example



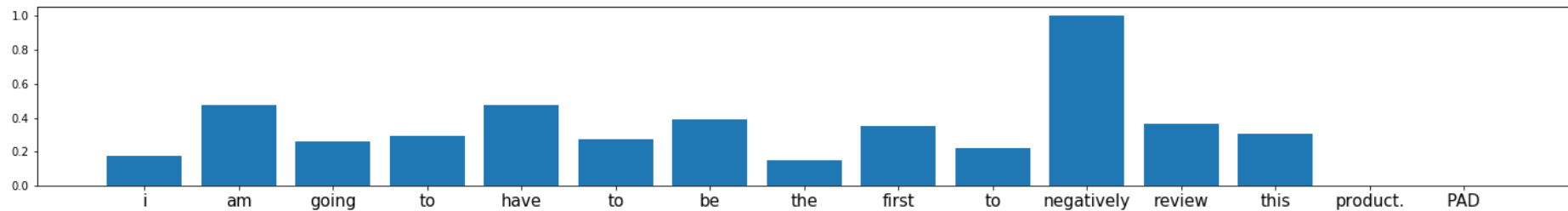
4-gram Relevance Flow

LRP: Question Classification Example



True Label: Human, Pred Label: Human

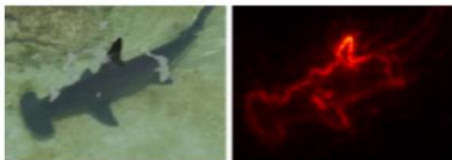
LRP: Sentiment Classification Example



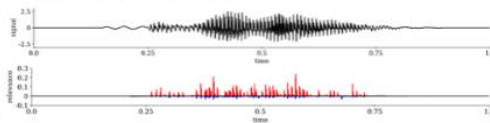
True Label: Negative, Pred Label: Negative

LRP: Other Examples

General Images (Bach' 15, Lapuschkin'16)



Speech (Becker'18)



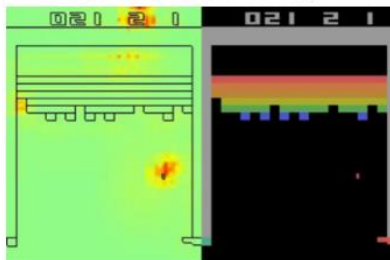
Text Analysis (Arras'16 &17)

do n't waste your money
neither funny nor susper

Morphing Attacks (Seibold'18)

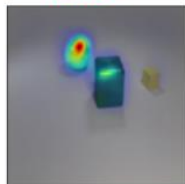


Games (Lapuschkin'19)



VQA (Samek'19)

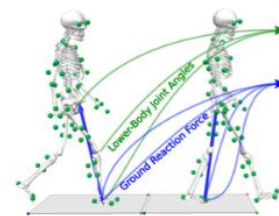
there is a metallic cube ; are
there any large cyan metallic
objects behind it ?



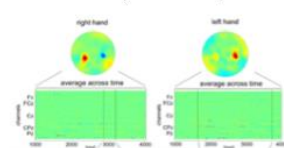
Video (Anders'19)



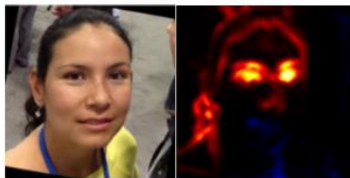
Gait Patterns (Horst'19)



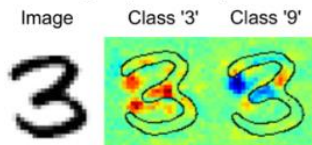
EEG (Sturm'16)



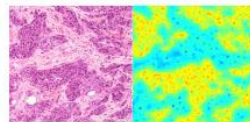
Faces (Lapuschkin'17)



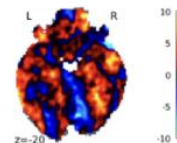
Digits (Bach' 15)



Histopathology (Hägele'19)



fMRI (Thomas'18)



Explainability and Prediction:

Using Linguistically Grounded Explanations for
Improving Neural NLP

Kevin Patel and Pushpak Bhattacharyya (under review)

Background and Motivation

- Feature Importance Explanations:
 - Given a model \mathbf{M} trained on data \mathbf{D} , the input \mathbf{x} is classified in class \mathbf{y} because of the presence of features $\mathbf{f}_i, \dots, \mathbf{f}_k$
 - Extremely helpful in cases where the features themselves are interpretable
 - \mathbf{f}_i is Current bank balance, \mathbf{f}_j is Previous loan fulfilment delay, etc. for deciding whether to grant loan or not
 - Becomes a bit subjective in case of uninterpretable features
 - \mathbf{f}_i is pixel at location (\mathbf{p}, \mathbf{q}) , \mathbf{f}_j is pixel at location (\mathbf{r}, \mathbf{s})
 - \mathbf{f}_i is value of \mathbf{p}^{th} dimension of the embedding of word at location \mathbf{q} , \mathbf{f}_j is value of \mathbf{r}^{th} dimension of the embedding of word at location \mathbf{s}

Problem Statement

Investigate methods to incorporate incongruity between gold feature importance and model obtained feature importance to improve neural models in natural language setting.

Why is NLP setting harder?

- Removing or replacing a word can have other unintended consequences
 - Example: Not only does the sentiment change, but also the grammar is messed up.
- May need to give an entire phrase importance, instead of just words
 - Example: Cannot focus on just *not* or *good* in the sentence *the movie was not good*.

Problem Statement, explained

- Feature importance explanations can help us identify potential issues in trained models
 - For instance, some sentiment analysis systems were found to be also focusing on gender and race terms while making predictions, thereby revealing inherent bias in their training.
- Can we penalize such incorrect feature importance by adding this incongruity as a loss in the training process?
 - For sentiment analysis, during training, if the network gives importance to sentiment bearing words, do nothing. But if it gives importance to gender and race terms, penalize it.
- Ross et. al. (2017) demonstrated one such loss function that used gradients of the network with respect to input as feature importance.
 - They showed it for interpretable features and images.

Example of Incongruity Capturing Loss Functions

$$\begin{aligned}
 L(\theta, X, y, A) &= \underbrace{\sum_{n=1}^N \sum_{k=1}^K -y_{nk} \log(\hat{y}_{nk})}_{\text{Right Answers}} \\
 &+ \underbrace{\lambda_1 \sum_{n=1}^N \sum_{d=1}^D \left(A_{nd} \frac{\partial}{\partial x_{nd}} \sum_{k=1}^K \log(\hat{y}_{nk}) \right)^2}_{\text{Right Reasons}} \\
 &+ \underbrace{\lambda_2 \sum_i \theta_i^2}_{\text{Regularization}}
 \end{aligned}$$

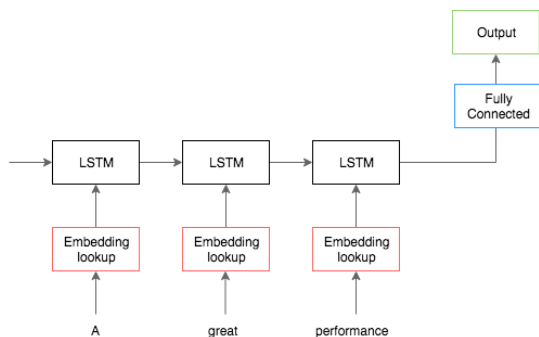
- A_{nd} is gold feature importance
- The remaining portion is predicted feature importance

x_n	x_{n1}	x_{n2}	x_{n3}	x_{n4}
A_n	1	0	0	1

Gold importance for an example with 2nd and 3rd features being important

Applying to Sentiment Analysis

- Given a review, predict whether its sentiment is positive or negative.
- Gold Importance: Sentiment bearing words obtained from SentiWordNet
- Feature Importance Method: Gradients of network with respect to input (Sensitivity Analysis)
- Model: LSTM based classifier



	MR (10%)	MR (full)	SST (10%)	SST (full)
CE	70.87	76.77	75.83	82.45
RfRR	74.34	79.57	78.62	86.68

Applying Loss for Classification Tasks (contd.)

- Tested the robustness of the resulting networks using challenge dataset that we created using variations of the reviews

Perturbation	CE	RfRR
Intensification	73.71	79.67
Negation	23.75	17.44
Tense Modification	77.35	84.70
Passivization	72.97	75.66
Exclamation	76.02	79.93

Performance on Challenge MR

	MR	SST
Original	10,662	79,654
Intensification	194,640	499,211
Tense Modification	126,873	408,362
Negation	5,802	28,700
Passivization	2,083	11,755
Exclamation	997	3,747
Total	330,395	951,775

Statistics of Challenge Dataset

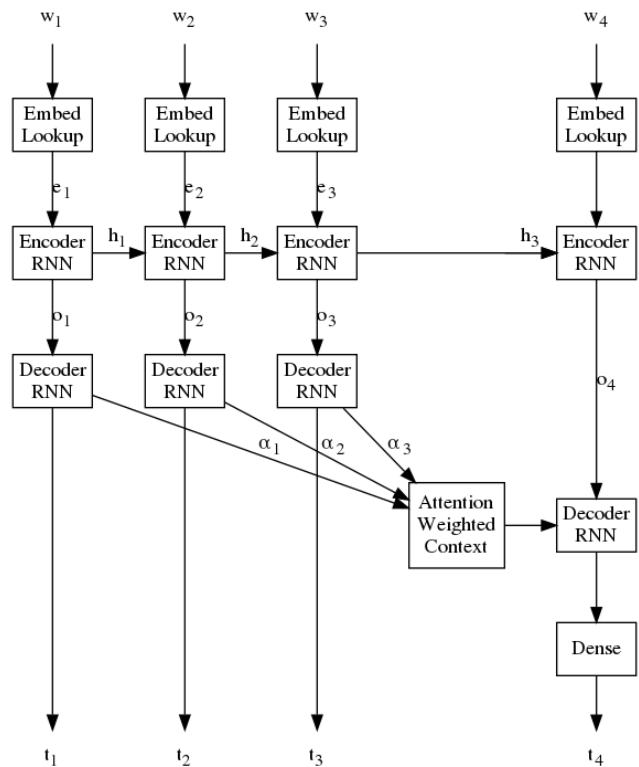
Applying Loss for Sequence Labeling Tasks

$$\begin{aligned} L(\theta, X, y, A^g) &= \underbrace{\sum_{n=1}^N \sum_{k=1}^K -y_{nk} \log(\hat{y}_{nk})}_{\text{Right Answers}} \\ &+ \underbrace{\lambda_1 \sum_{n=1}^N \sum_{d=1}^D (A_{nd}^g - A_{nd}^p)^2}_{\text{Right Reasons}} \\ &+ \underbrace{\lambda_2 \sum_i \theta_i^2}_{\text{Regularization}} \end{aligned}$$

- A_{nd}^g : Gold Attention
- A_{nd}^p : Predicted Attention

Applying Loss for Sequence Labeling Tasks (contd.)

- For instance consider the task of Part of Speech Tagging
 - Generally, in order to predict current tag, the current word and the previous two tags are enough
- Designed an encoder decoder network with an attention layer on top of decoder
 - So that the network can attend to past tags while decoding for the current tag



Applying Loss for Sequence Labeling Tasks (contd.)

- Here, as per our intuition, gold attention is to attend on the previous two tags
 - Accuracy of the system without incongruity loss - 74%
 - Accuracy of the system with incongruity loss - 84%
- This establishes that such incongruity based loss could also be used for sequence labeling tasks
- Could be applied to Machine Translation, Named Entity Recognition, *etc.*
 - Gold Attention for Machine Translation: Alignment from Giza++
 - Gold Attention for Named Entity Recognition: Features such as capitalization, *etc.*

Summary

- Gave a perspective to explainability
- Discussed advantages of explainability
- Explainability problem formulations
- Feature importance explanation techniques
 - LIME, IG and LRP
- Explainability improving Prediction (model performance)
 - Introduces helpful inductive bias
 - Reduces blind groping in the dark
 - Reduces spurious correlations

Conclusion and Future work

Deep Learning has entered day to day life; its decisions are affecting and will affect our lives. For example, why did the NN label the eye image as onset of diabetic retinopathy?

Will need explanation as to why the DNN concluded the way it did

Currently explanation is essentially **correlation capture**;
followed by **sensitivity analysis**

In future: Have to tackle causality and explore relation between explainability and prediction



References I

- Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7), e0130140.
- Dhurandhar, A., Chen, P. Y., Luss, R., Tu, C. C., Ting, P., Shanmugam, K., & Das, P. (2018). Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *Advances in Neural Information Processing Systems* (pp. 592-603).
- Gunning, D. (2017). Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA), nd Web, 2.
- Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5), 93.
- Kim, B., Khanna, R., & Koyejo, O. O. (2016). Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems* (pp. 2280-2288).

References II

- Lipton, Z. C. (2016). The mythos of model interpretability. arXiv preprint arXiv: 160603490.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). Why should i trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144). ACM.
- Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 618-626).
- Sundararajan, M., Taly, A., & Yan, Q. (2017, August). Axiomatic attribution for deep networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70 (pp. 3319-3328). JMLR. org.