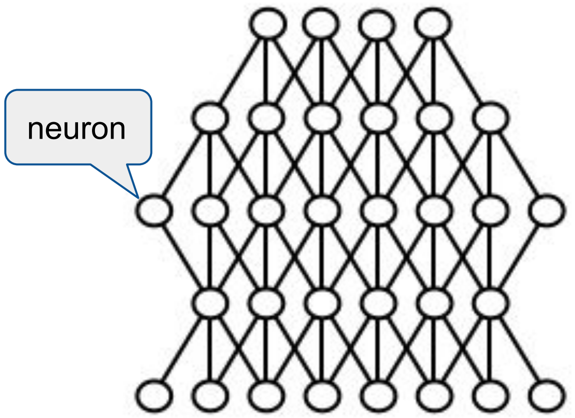
Why did the network make this prediction?

Ankur Taly (ataly@)
go/probe

(Joint work with Mukund Sundararajan, Qiqi Yan, and Kedar Dhamdhere)

Deep Neural Networks

Output
(Image label, next word, next move, etc.)



Input
(Image, sentence, game position, etc.)

Flexible model for learning arbitrary non-linear, non-convex functions

Transform input through a network of neurons

Each neuron applies a non-linear activation function (σ) to its inputs

$$n_3 = \sigma(w_1 . n_1 + w_2 . n_2 + b)$$

Understanding Deep Neural Networks

We understand them enough to:

- Design architectures for complex learning tasks (supervised and unsupervised)
- Train these architectures to favorable optima
- Help them generalize beyond training set (prevent overfitting)

But, a trained network largely remains a black box to humans

Objective

Understanding the input-output behavior of Deep Networks

i.e., *we ask why did it make this prediction on this input?*

Why did the network label this image as **“fireboat”**?

Retinal Fundus Image

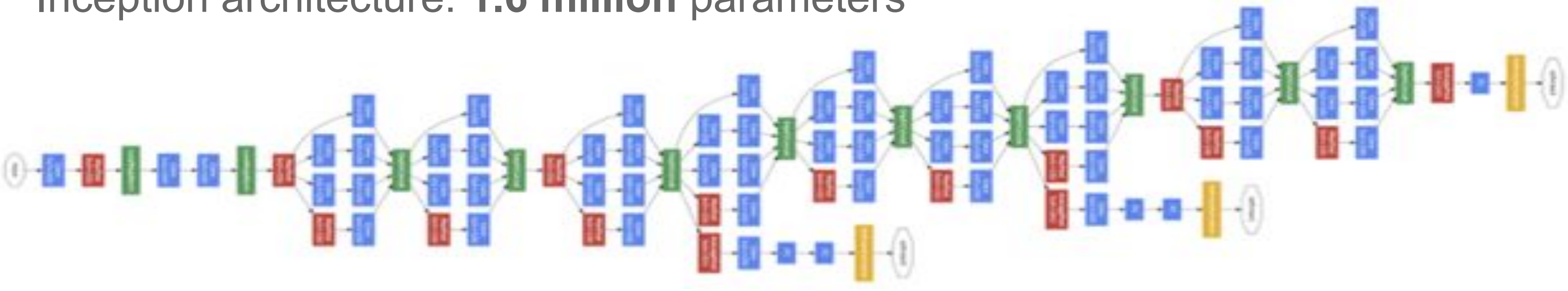
Why does the network label this image with **"mild"** Diabetic Retinopathy?

Why study input-output behavior of deep networks?

- Debug/Sanity check networks
- Surface an explanation to the end-user
- Identify network biases and blind spots
- Intellectual curiosity

Analytical Reasoning is very hard

Inception architecture: **1.6 million** parameters



- Modern architectures are way too complex for analytical reasoning
  - The meaning of individual neurons is not human-intelligible
- Could train a simpler model to approximate its behavior
  - **Faithfulness** vs. **Interpretability**

The Attribution Problem

Attribute a deep network's prediction to its input features, relative to a certain baseline input

- E.g., Attribute an object recognition network's prediction to its pixels
- E.g., Attribute a text sentiment network's prediction to individual words

Need for a baseline

- Every explanation involves an implicit or explicit counterfactual
  - see [Kahneman-Miller 86]
- Ideally, the baseline is an informationless input for the network
  - e.g., black image for image networks
- The baseline may also be an important analysis knob

Outline

- Our attribution method: **Integrated Gradients**
- Applications of the method
- Justifying Integrated Gradients
- Case Study: Neural Programmer
- Discussion

Naive approach: Ablations

Ablate each input feature and measure the change in prediction

Downsides:

- Costly, especially for image networks with (224*224*3) pixel features
- Unrealistic inputs
- Misleading when there are interactive features
    - E.g., Query=”Facebook” AND Domain=”facebook.com”   IMPLIES high click through rate

Gradient-based Attribution

Attribute using gradient of the output w.r.t each input feature

Attribution for feature x_i is $x_i * \partial y / \partial x_i$

- Standard approach for understanding linear models
    - Here, gradients == feature weights
- First-order approximation for non-linear models

Inception on ImageNet

Fireboat (0.9999)	pier (3e-5)	Steel arch bridge (6e-7)	Crane (4e-7)	Liner (4e-2)

Visualizing Attributions

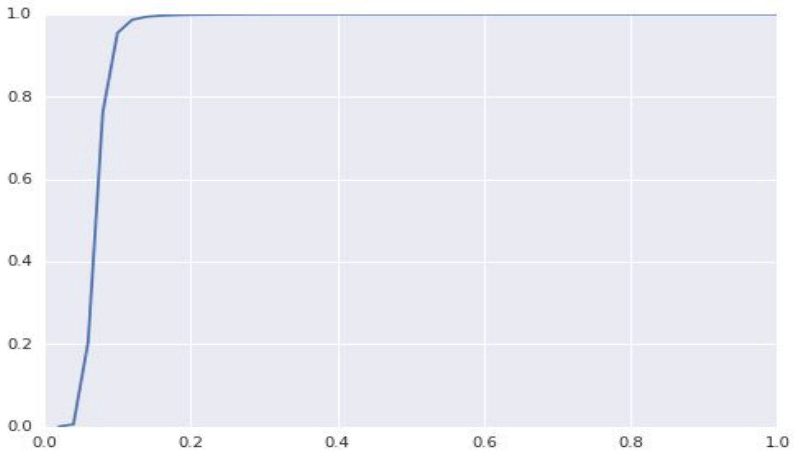
Visualization: Use (normalized) attribution as mask/window over image

Why the sky?

Attribution using gradients

Saturation

Prediction Score



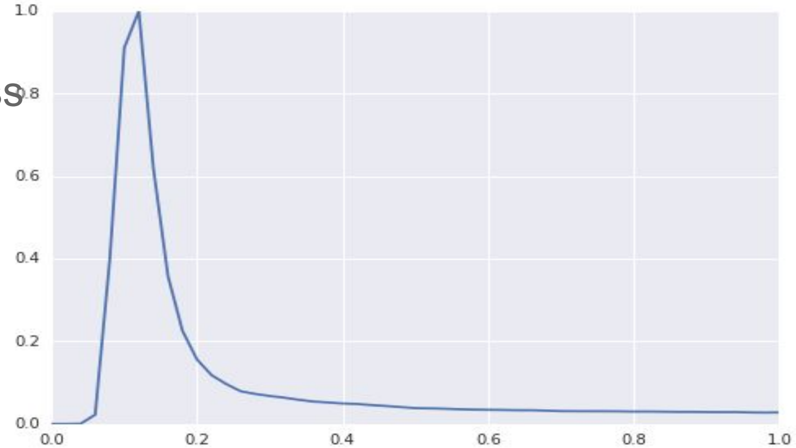
Intensity α

… Scaled inputs ...

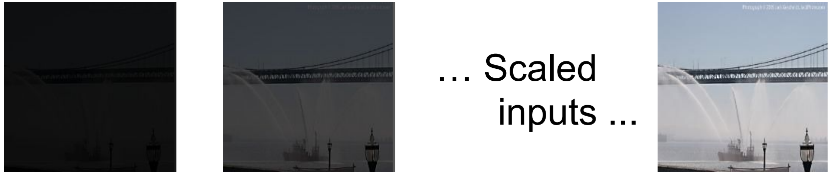
Baseline

Image

Saturation

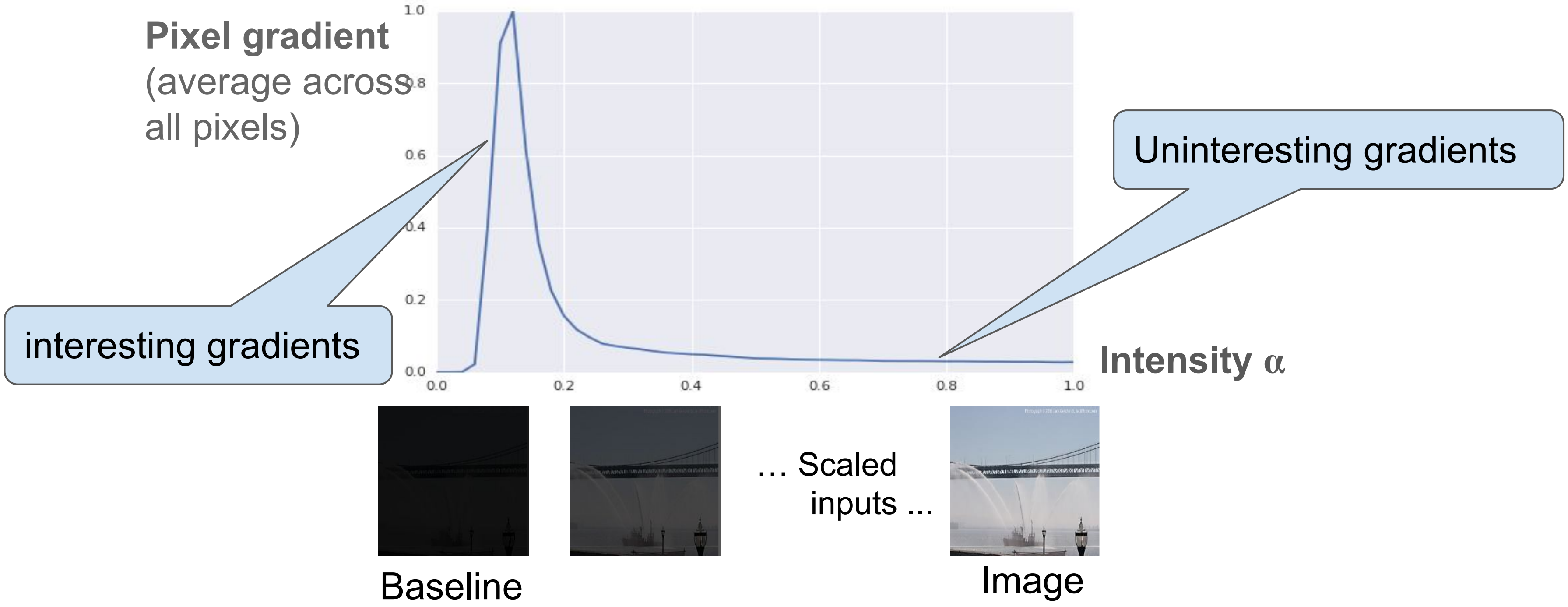


Pixel gradient
(average across all pixels)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Intensity α



Baseline
… Scaled inputs ...
Image

Saturation



Pixel gradient
(average across all pixels)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Intensity α
interesting gradients
Uninteresting gradients
… Scaled inputs ...
Baseline
Image

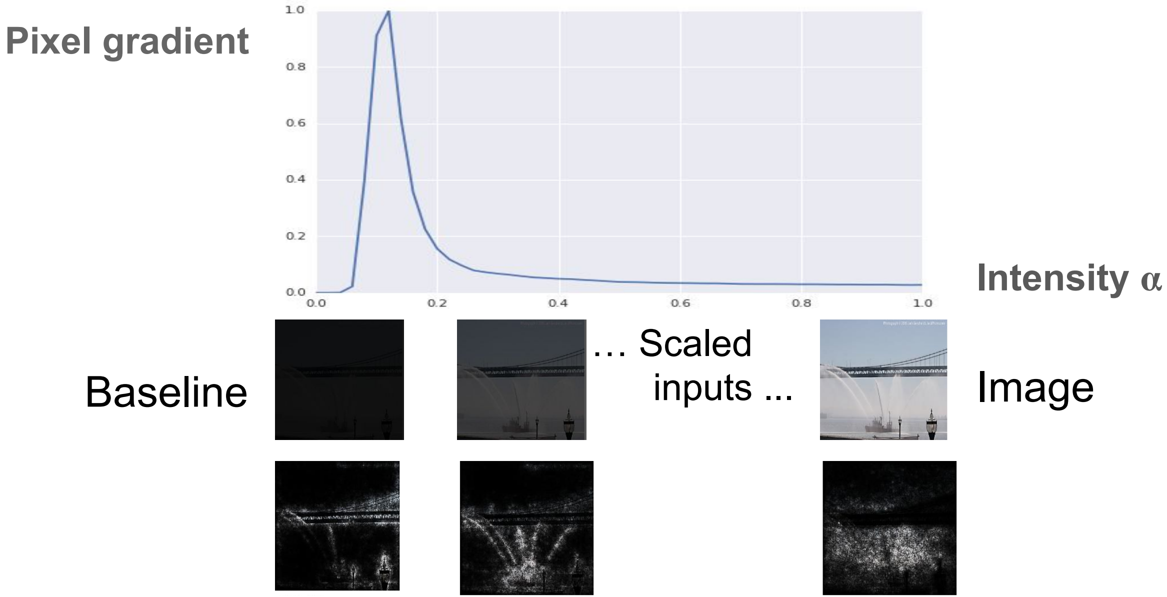
Saturation occurs...

- across images
  - Not just the two images we discussed
- across networks
  - Not just Inception on ImageNet
  - Severity varies

(see [this paper] for details)

The Method: Integrated Gradients

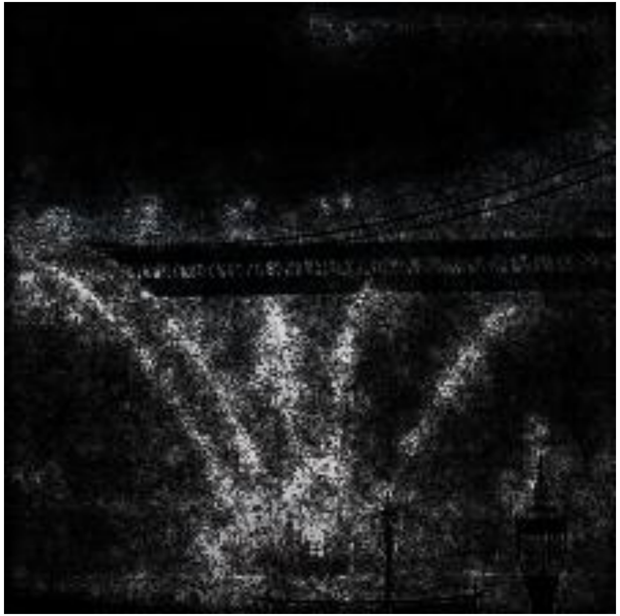
$$\text{IG(input, base)} ::= \text{(input - base)} * \int_{0-1} \nabla F(\alpha\text{*input} + (1-\alpha)\text{*base})\, d\alpha$$



Original image

Gradient at image

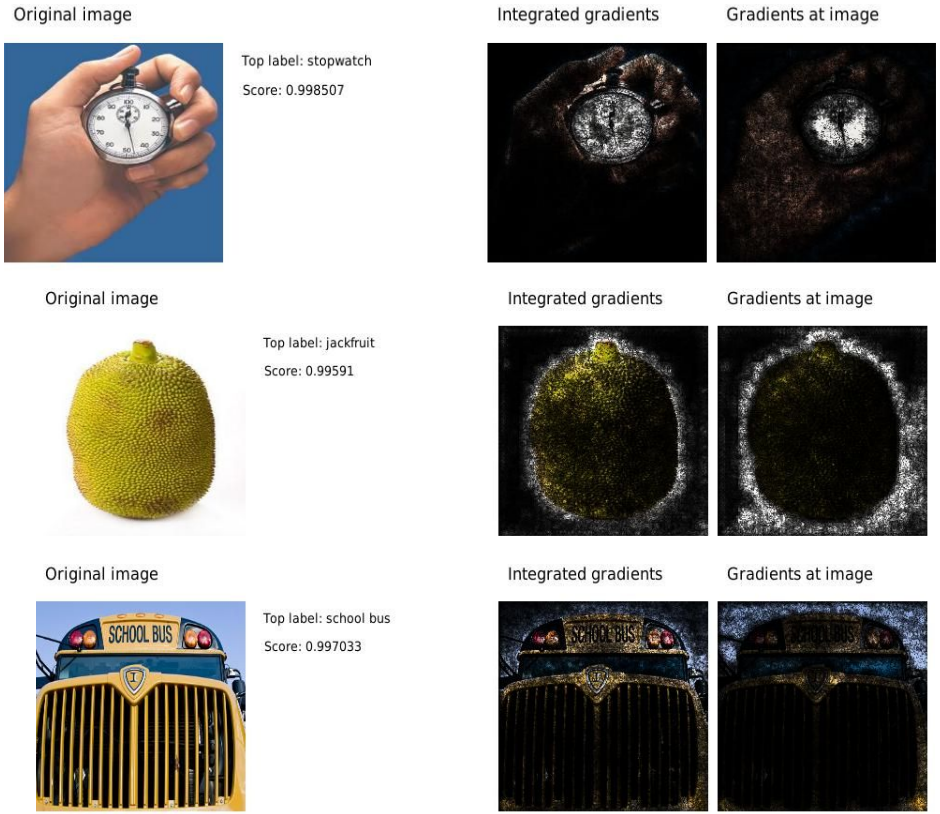
Integrated gradient



Original image (Turtle)

Gradient at image

Integrated gradient



Many more Inception+ImageNet examples [here]

Misconception

Human label: accordion
Network's top label: toaster

Misconception

Human label: accordion
Network's top label: toaster

Integrated gradient

Very few lines of code...

```
def integrated_gradients(inp, base, label, steps=50):
  scaled_inps = [base + (float(i)/steps)*(inp-base) for i in range(0, steps)]
  predictions, grads = predictions_and_gradients(scaled_inputs, label)
  integrated_gradients = (img - base) * np.average(grads, axis=0)
  return integrated_gradients
```

see [this colab]

Baseline matters

Black baseline **White baseline**

Applications

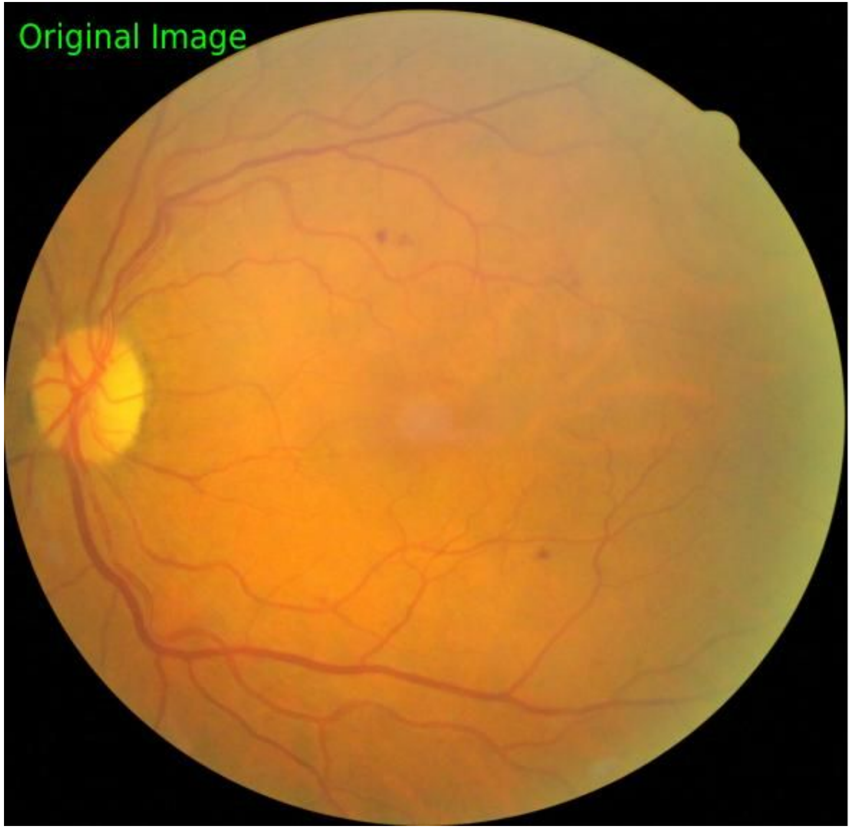
Diabetic Retinopathy

Diabetes complication that causes damage to blood vessels in the eye due to excess blood sugar.

An Inception-based network for predicting diabetic retinopathy grade from retinal fundus images achieves **0.97 AUC** [JAMA paper]

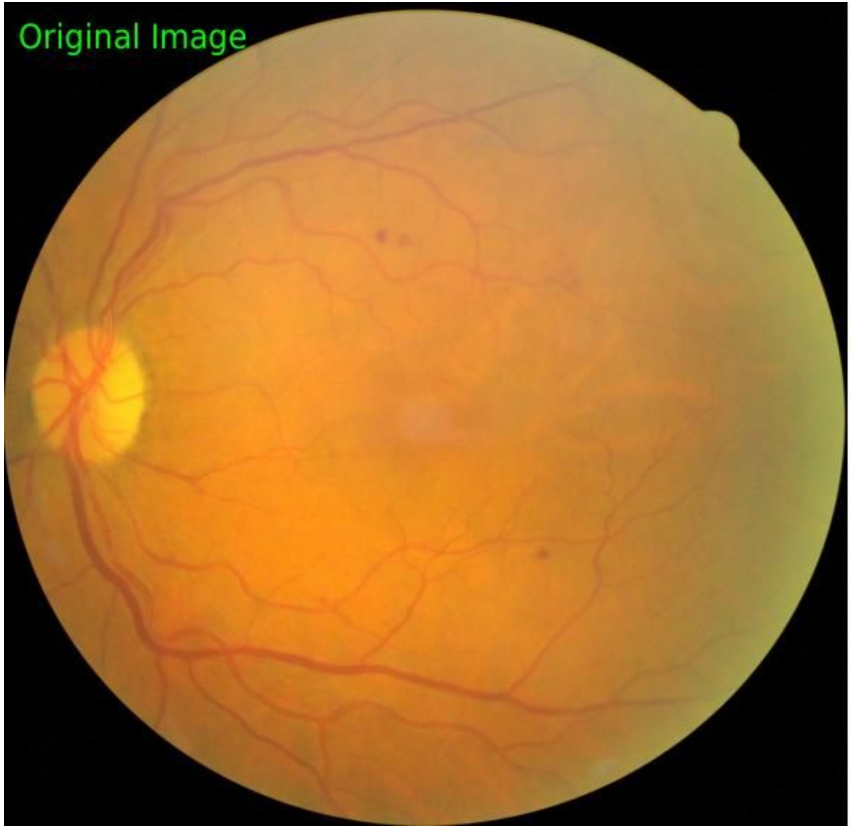
On what basis, does the network predict the DR grade?

A prediction

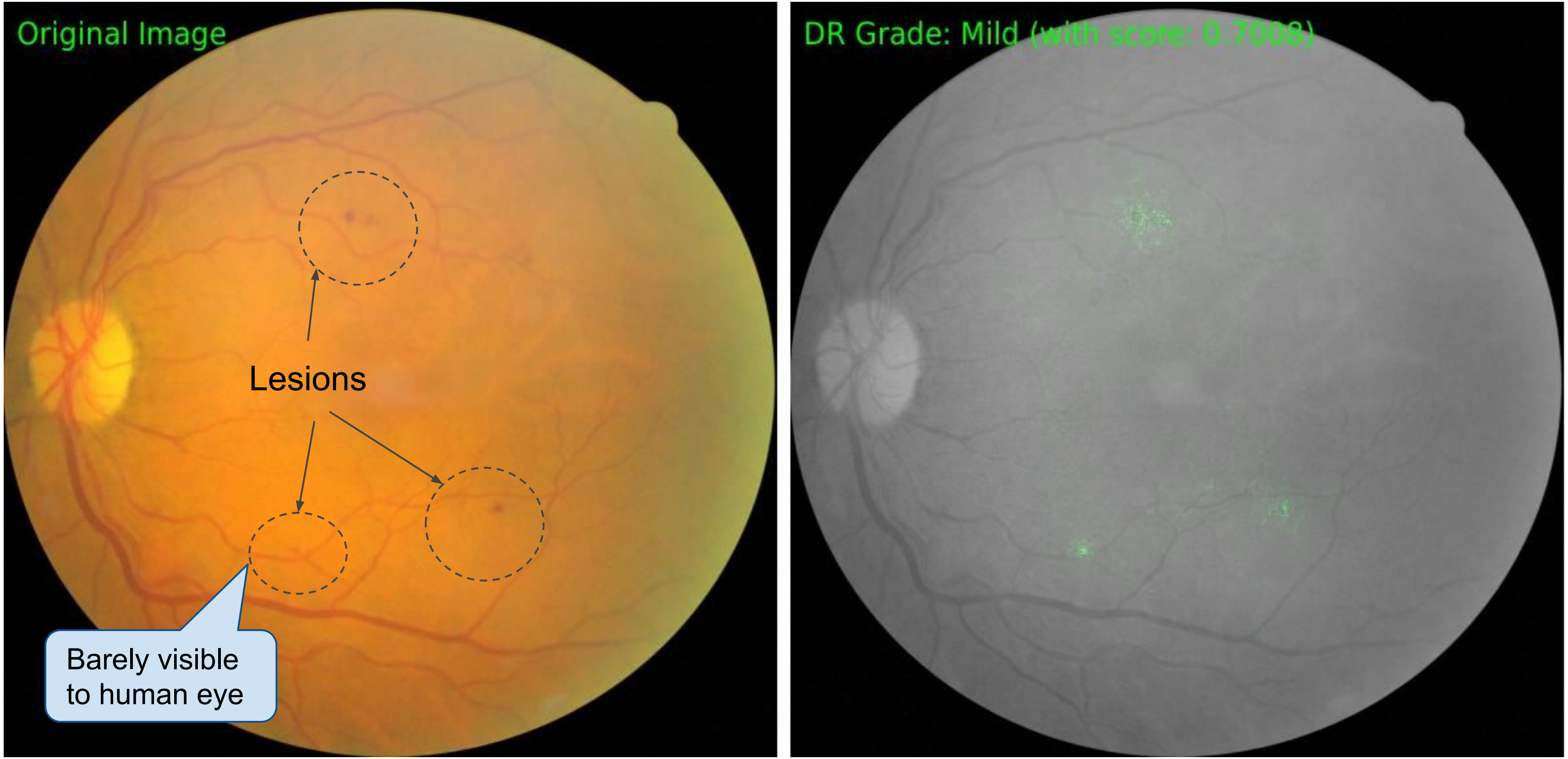


Predicted DR grade: Mild

Surfacing an explanation to the doctor!



Surfacing an explanation to the doctor!



Application: Text Classification

- We have a data set of questions and answers
  - Answer types include numbers, strings, dates, and yes/no
- Can we predict the answer type from the question?
  - Answer: Yes using a simple feedforward network
- Can we tell which words were indicative of the answer type?
  - Enter attributions
- **Key issue**: What is the baseline (analog of the black image)?
  - Answer: the zero embedding vector

Application: Text Classification

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]

Red is positive attribution
Blue is negative attribution
Shades interplolate

Application: Text Class

Several sensible results, can almost harvest these as grammar rules

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]

Overfitting?

Negative signals too

Many Other Applications

- Search Ranking
  - What makes one result rank higher than another?
- Language translation
  - Which input word does this output word correspond to?
- Text sentiment
  - Which input words cause negative sentiment?

Justifying Integrated Gradients

Related Work on Attributions

- Score back-propagation methods
  - DeepLift [ICML'17], Layerwise Relevance Propagation [JMLR'17], Guided BackPropagation [CoRR'14], DeConvNets [CVPR '10]…
- Local Model Approximation
  - E.g., LIME [KDD '16], Anchors [AAAI '18]
- Shapley value based methods
  - E.g., Quantitative Input Influence [S&P '16], SHAP [NIPS '17]
- Gradient-based methods
  - E.g., SmoothGrad [2017], SaliencyMaps [2014]

How do you evaluate an attribution method?

How do you evaluate an attribution method?

- **Eyeball Attributions**
  - Issue: Attribution may “look” incorrect due to unintuitive network behavior
  - Issue: Preference to methods that agree with human reasoning (**confirmation bias**)
- **Ablate top attributed features**
  - Issue: Ablations may change prediction for artifactual reasons

Hard to separate model behavior, attribution errors, eval artifacts

How do you evaluate an attribution method?

- **Eyeball Attributions**
  - Issue: Attribution may “look” incorrect due to unintuitive network behavior
  - Issue: Preference to methods that agree with human reasoning (**confirmation bias**)
- **Ablate top attributed features**
  - Issue: Ablations may change prediction for artifactual reasons

Hard to separate model behavior, attribution errors, eval artifacts

Our approach:

- List **desirable criteria (axioms)** for an attribution method
- Establish a uniqueness result: X is the **only** method that satisfies these desirable criteria

Axiom: **Sensitivity**

A. If starting from baseline, varying a variable changes the output, then the variable should receive some attribution.

B. A variable that has no effect on the output gets no attribution.

(A) not satisfied by:

- Gradient at output
- DeConvNets
- Guided Backpropagation

Axiom: **Implementation Invariance**

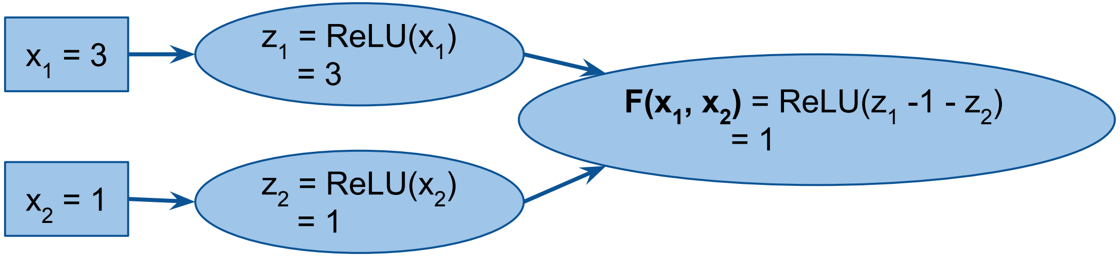
Two networks that compute identical functions <u>for all inputs</u> get identical attributions even if their architecture/parameters differ

E.g. F = x*y + z and G = y*x + z should get the same attributions

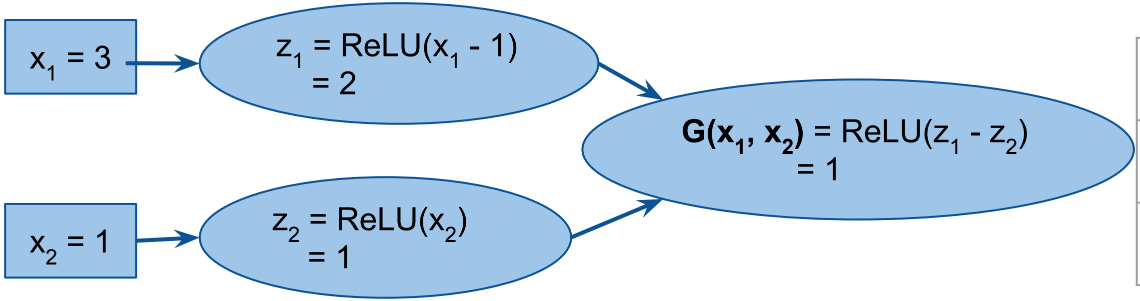
Not satisfied by:

- DeepLift
- Layerwise Relevance Propagation

For all x_1 and x_2: $F(x_1, x_2) == G(x_1, x_2)$



Integrated gradients	**$x_1 = 1.5$, $x_2 = -0.5$**
DeepLift	$x_1 = 1.5$, $x_2 = -0.5$
LRP	$x_1 = 1.5$, $x_2 = -0.5$



Integrated gradients	**$x_1 = 1.5$, $x_2 = -0.5$**
DeepLift	$x_1 = 2$, $x_2 = -1$
LRP	$x_1 = 2$, $x_2 = -1$

Axiom: **Linearity Preservation**

If the function **F** is a linear combination of two functions $\mathbf{F_1}$,$\mathbf{F_2}$ then the attributions for **F** are a linear combination of the attributions for $\mathbf{F_1}$,$\mathbf{F_2}$

I.e., Attributions(α*F1 + ß*F2) = α*Attributions(F1) + ß*Attributions(F2)

Rationale:

- Attributions have additive semantics, good to respect existing linear structure
- E.g., For F = x*y + z, the "optimal" attribution should assign blame independently to ‘z’ and ‘x*y’

Axiom: **Completeness**

Sum(attributions) = F(input) - F(baseline)

Rationale: Attributions apportion the prediction

- Break down the predicted click through rate (pCTR) of an ad like:
    - 55% of pCTR is because it’s at position 1
    - 25% is due to its domain (a popular one)
    - …

Theorem [Friedman 2004]

Every method that satisfies Linearity preservation, Sensitivity and Implementation invariance, and Completeness is a path integral of a gradient.

Axiom: **Symmetry**

Symmetric variables with identical values get equal attributions

Rationale:

- E.g., For F = x*y + z, the "optimal" attribution at x,y,z=1,1,2 should be equal for x and y.

Theorem: [This work]

Integrated Gradients is the unique path method that satisfies these axioms. (there are other methods that take an average over a symmetric set of paths)

Highlights of Integrated Gradients

- Easy to implement
  - Gradient calls on a bunch of scaled down inputs
  - No instrumentation of the network, no new training
- Widely applicable
- Backed by an axiomatic guarantee

References

- Google Data Science Blog: Attributing a deep network's prediction to its input
- Paper [ICML 2017]: Axiomatic Attribution for Deep Networks

Case Study: **Neural Programmer**

(Joint work with Pramod Mudrakarta, Mukund Sundararajan, Qiqi Yan, and Kedar Dhamdhere)

Question-Answering Task

Answer a natural language question on a table (think: spreadsheet)

1999 South Asian Games

Rank	Nation	Gold	Silver	Bronze	Total
1	India	102	58	37	197
2	Nepal	32	10	24	65
3	Sri Lanka	16	42	62	120
4	Pakistan	10	36	30	76
5	Bangladesh	2	10	35	47
6	Bhutan	1	6	7	14
7	Maldives	0	0	4	4

Q: How many gold medals did India win?
A: 102

Q: how many countries won more than 10 gold medals?
A: 3

WikiTables Dataset (WTQ) [Pasupat and Liang 2015]

Dataset of 22,033 **<Question, Table, Answer>** triples (split into train, dev, test)

- Tables scraped from Wikipedia; Questions and Answers by Mechanical Turkers
- Wide variety of questions
    - **[Max/Min]** which lake has the **greatest** elevation?
    - **[A_or_B]** who won more gold medals, brazil **or** china?
    - **[Position]** which location comes **after** kfar yona?
    - **[Count] how many** ships were built after ardent?

Traditional Approach: Semantic Parsing

Stop words

How many countries have won more than 10 gold medals?

Intent word Dimension Filter Metric

- **Annotate** utterances with **typed entities** (metrics, dimensions, filters, etc.)
- **Parse** annotated sentence using a **grammar** into a **logical form**
- Execute logical form to obtain an answer

Relies on human authored grammar, synonym lists, and scoring heuristics

- Good precision but poor recall

Our Protagonist: **Neural Programmer** [ICLR 2016 and ICLR 2017]

- Deep network augmented with a **fixed set of primitive operations**
  - Belongs to the family of Neural Abstract Machine architecture
- Learns to compose operators and apply them to the table to obtain an answer
- Trained end-to-end on <question, table, answer> triples

Eliminates the need for hand-crafted grammars, synonym lists and other heuristics. Instead, learns these from data!

Understanding Neural Programmer (NP)

- What triggers various operator and column selections?
- Can we extract rules from NP that we could use in a hand-authored system?
  - Can we extract a grammar from NP?
- How robust is NP's reasoning?
  - Can we craft adversarial examples to fool it?

Example 1

Rank	Athlete	Nationality	Time	Notes
	Valeriy Borchin	Russia	1:19:56	
	Vladimir Kanaykin	Russia	1:20:27	
	Luis Fernando López	Colombia	1:20:38	SB
4	Wang Zhen	China	1:20:54	
5	Stanislav Emelyanov	Russia	1:21:11	
6	Kim Hyun-sub	South Korea	1:21:17	
7	Ruslan Dmytrenko	Ukraine	1:21:31	SB
8	Yusuke Suzuki	Japan	1:21:39	
9	Alex Schwazer	Italy	1:21:50	SB
10	Erick Barrondo	Guatemala	1:22:08	
11	Chu Yafei	China	1:22:10	
12	Sergey Morozov	Russia	1:22:37	
13	Wang Hao	China	1:22:49	

Q: Wang Zheng and Wang Hao are from which **country**?

Neural Programmer: China

Example 1

Rank	Athlete	Nationality	Time	Notes
	Valeriy Borchin	Russia	1:19:56	
	Vladimir Kanaykin	Russia	1:20:27	
	Luis Fernando López	Colombia	1:20:38	SB
4	Wang Zhen	China	1:20:54	
5	Stanislav Emelyanov	Russia	1:21:11	
6	Kim Hyun-sub	South Korea	1:21:17	
7	Ruslan Dmytrenko	Ukraine	1:21:31	SB
8	Yusuke Suzuki	Japan	1:21:39	
9	Alex Schwazer	Italy	1:21:50	SB
10	Erick Barrondo	Guatemala	1:22:08	
11	Chu Yafei	China	1:22:10	
12	Sergey Morozov	Russia	1:22:37	
13	Wang Hao	China	1:22:49	

Q: Wang Zheng and Wang Hao are from which **country**?

Neural Programmer: China

Operator Selection:

Select (Athlete)	First	Print (**Nationality**)

What triggered the "**Nationality**" column?

Example 2

Rank	Nation	Gold	Silver	Bronze	Total
1	Cuba	4	3	2	9
2	Canada	4	2	1	7
3	United States	2	0	2	4
4	Mexico	1	1	0	2
5	Ecuador	1	0	0	1
6	Argentina	0	4	3	7
7	Brazil	0	2	2	4
8	Chile	0	0	1	1
8	Venezuela	0	0	1	1
Total	Total	12	12	12	36

Q: Which nation earned the most gold medals?

Neural Programmer: Cuba

Example 2

Rank	Nation	Gold	Silver	Bronze	Total
1	Cuba	4	3	2	9
2	Canada	4	2	1	7
3	United States	2	0	2	4
4	Mexico	1	1	0	2
5	Ecuador	1	0	0	1
6	Argentina	0	4	3	7
7	Brazil	0	2	2	4
8	Chile	0	0	1	1
8	Venezuela	0	0	1	1
Total	Total	12	12	12	36

Q: Which nation earned the most gold medals?

Neural Programmer: Cuba

Operator Selection:

Prev (Team)	First	Print (Team)

What triggered operator **Prev?**

What triggered operator **First?**

Example 3

	Place	Team	Matches	Won	Drawn	Lost	Difference	Points
0	1	Canada	6	6	0	0	62–6	12
1	2	Sweden	6	4	1	1	33–14	9
2	3	Switzerland	6	4	1	1	28–12	9
3	4	Norway	6	2	0	4	10–27	4
4	5	Great Britain	6	1	1	4	18–42	3
5	6	United States	6	1	1	4	14–42	3
6	7	Finland	6	1	0	5	15–37	2

Q: which **country performed better** during the 1951 world ice hockey championships, **switzerland** or **great britain**?

Neural Programmer: Switzerland

Example 3

	Place	Team	Matches	Won	Drawn	Lost	Difference	Points
0	1	Canada	6	6	0	0	62–6	12
1	2	Sweden	6	4	1	1	33–14	9
2	3	Switzerland	6	4	1	1	28–12	9
3	4	Norway	6	2	0	4	10–27	4
4	5	Great Britain	6	1	1	4	18–42	3
5	6	United States	6	1	1	4	14–42	3
6	7	Finland	6	1	0	5	15–37	2

Q: which **country performed better** during the 1951 world ice hockey championships, **switzerland** or **great britain**?

Neural Programmer: Switzerland

Operator Selection

Select (Team)	First	Print (Team)

What triggered this non-robust selection?

Basic Questions

- Which inputs and outputs should we focus on?
    - **Not immediately clear**:
        - Several inputs comprising of question/table features, masks, labels, etc.
        - Answer computation logic is partly continuous and partly discrete
- What is the right baseline?

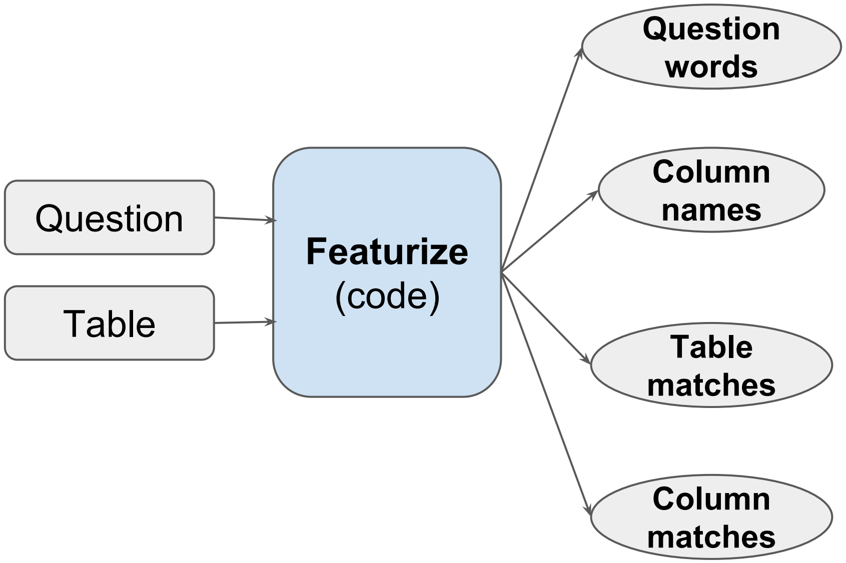
Basic Questions

- Which inputs and outputs should we focus on?
  - **Not immediately clear**:
    - Several inputs comprising of question/table features, masks, labels, etc.
    - Answer computation logic is partly continuous and partly discrete
- What is the right baseline?

Take inspiration from program debugging,

- Abstract out uninteresting details
- Focus on parts that are most mysterious or error-prone

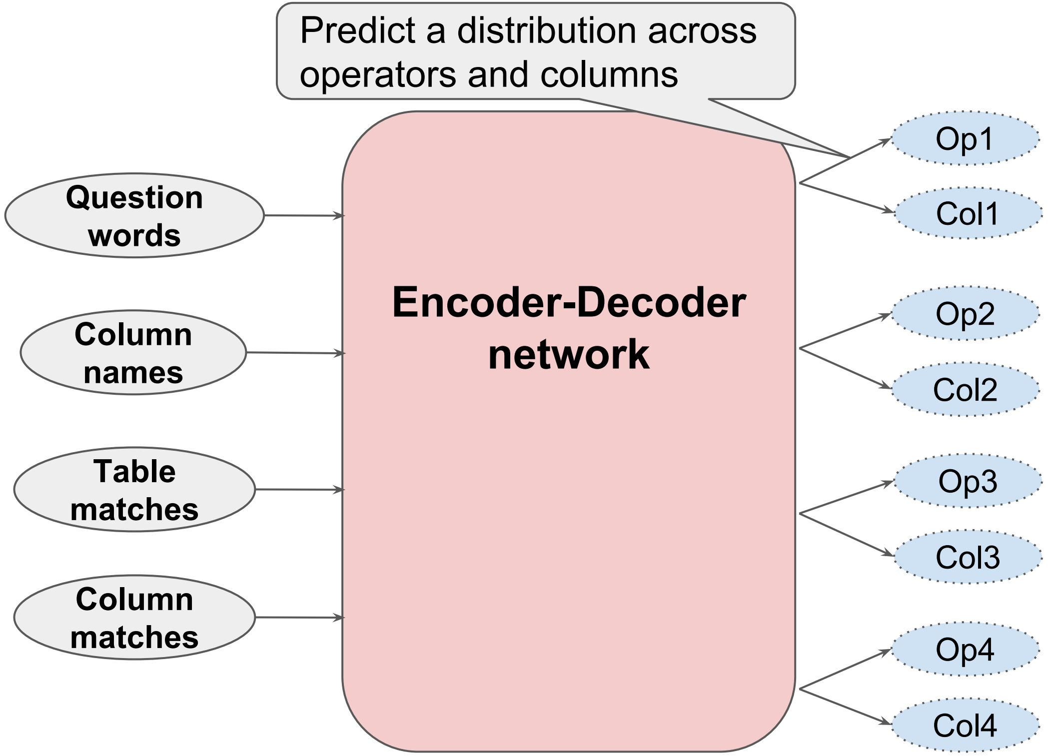
Question and Table Featurization



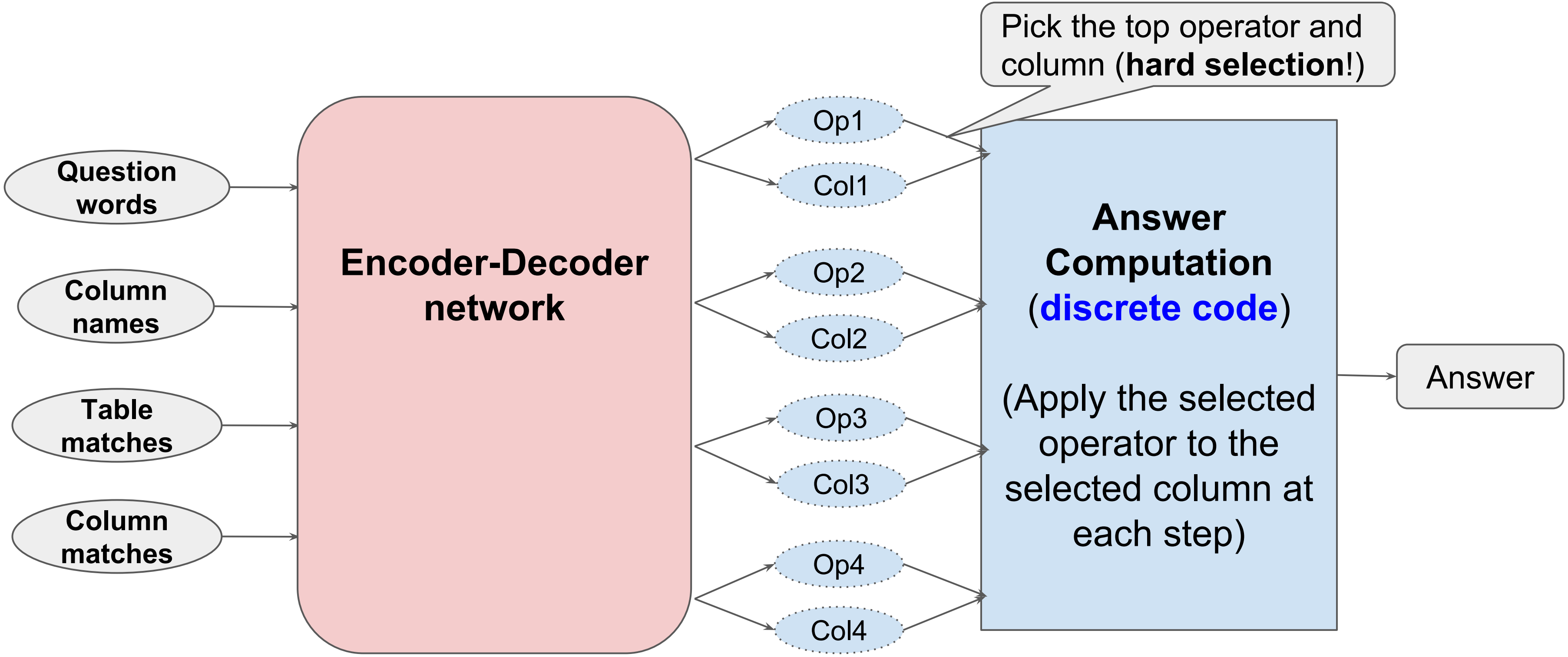
- **Column matches**: Boolean tensor indicating which column names share a word with the question
- **Table matches**: Boolean tensor indicating which table cells share a word with the question
- Special tokens <tm_token>, <cm_token> are added to the question when above tensors are non-zero

Network nevers sees the table contents; it sees only the table matches

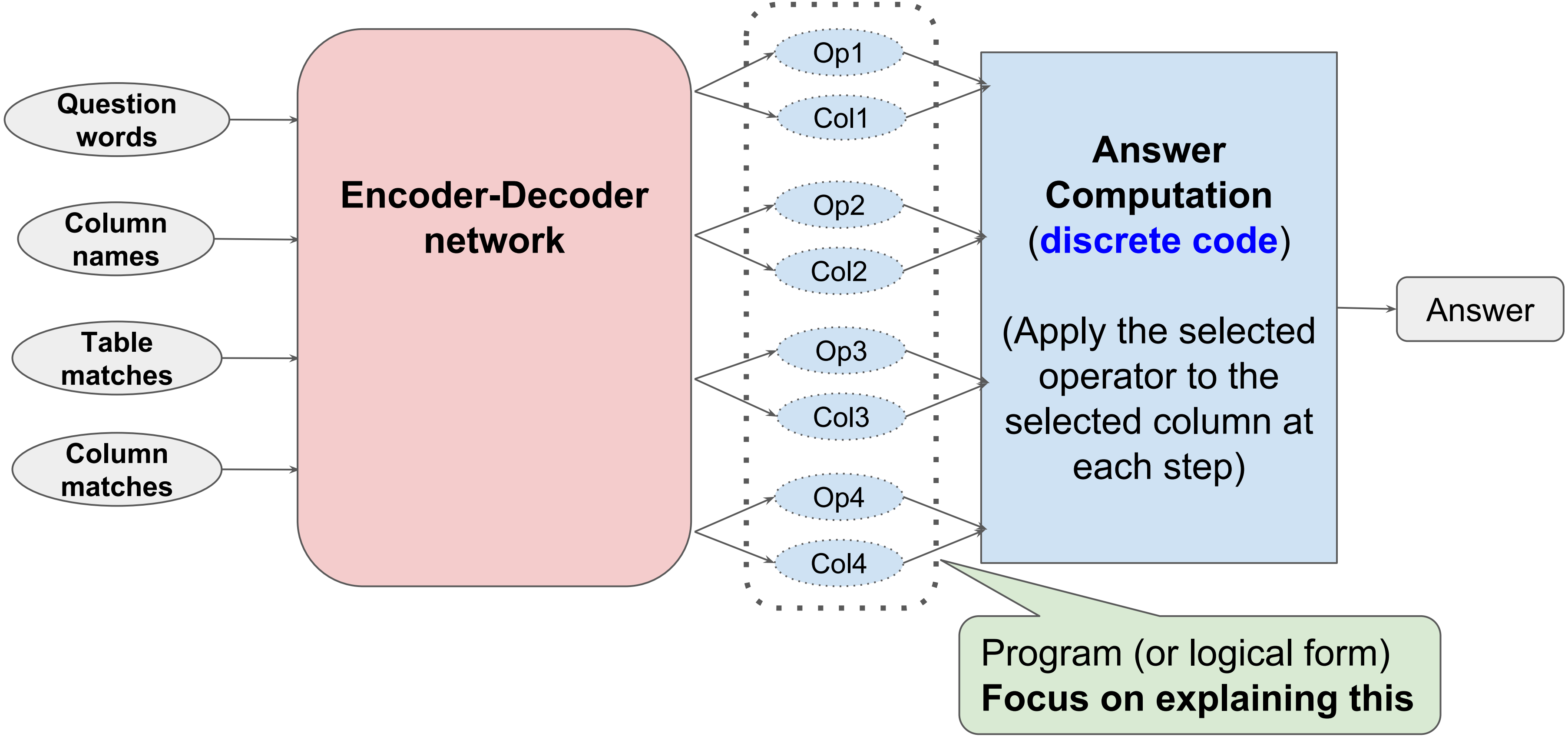
Answer Computation (during inference)



Answer Computation (during inference)



Answer Computation (during inference)



Currying

col-names $\longrightarrow$ **< ques-words, table-matches, col-matches >** $\longrightarrow$ $\mathbf{R}^{\#\mathbf{operators}}$

(analogous function for column selection)

Split the analysis:

1. Understand the influence of table inputs (column names)
2. Understand the influence of question inputs given the table

Step 1: Understanding Table Influence

We invoked the network on a given set of column names but **empty question** **(i.e., ques-words = [], table-matches = 0, column-matches = 0)**

- We expected this to return uniform operator and column distributions
- Instead, the distributions were quite skewed ⇒ network has a bias per table
- We call the (skewed) selections **Table-Default Programs**

Next step: Attribute table-default programs to column names

Table-Default Programs

Operator selections	Num. tables	Attributions to *cnames*
reset, reset, max, print	108	UNK, year, date, name, points, position, competition, notes, team, no
reset, prev, max, print	67	UNK, rank, total, gold, silver, bronze, nation, year, name, no
reset, reset, first, print	29	UNK, name, notes, year, nationality, rank, date, location, previous, comments
reset, mfe, first, print	26	year, date, UNK, notes, title, role, genre, opponent, score, surface
reset, reset, min, print	16	year, UNK, name, height, location, jan, may, jun, notes, floors
reset, mfe, max, print	14	opponent, date, result, site, rank, year, attendance, location, notes, city
reset, next, first, print	10	UNK, name, edition, year, death, time, type, men, birth, women
reset, reset, last, print	10	UNK, year, date, location, album, winner, score, type, opponent, peak
reset, prev, last, print	5	date, votes, candidate, party, season, report, UNK, city, west, east

(similar table for column selections)

Table-Default Programs

Sports tables?

Operator selections	Num. tables	Attributions to *cnames*
reset, reset, max, print	108	UNK, year, date, name, points, position, competition, notes, team, no
reset, prev, max, print	67	UNK, rank, total, gold, silver, bronze, nation, year, name, no
reset, reset, first, print	29	UNK, name, notes, year, nationality, rank, date, location, previous, comments
reset, mfe, first, print	26	year, date, UNK, notes, title, role, genre, opponent, score, surface
reset, reset, min, print	16	year, UNK, name, height, location, jan, may, jun, notes, floors
reset, mfe, max, print	14	opponent, date, result, site, rank, year, attendance, location, notes, city
reset, next, first, print	10	UNK, name, edition, year, death, time, type, men, birth, women
reset, reset, last, print	10	UNK, year, date, location, album, winner, score, type, opponent, peak
reset, prev, last, print	5	date, votes, candidate, party, season, report, UNK, city, west, east

(similar table for column selections)

Bias can be useful

- When question has OOV words, final program == table-default program
- For 6% of dev data instances, the table-default program is the final program

There is a **global default for empty table, empty question** too!

Reset (prob: 0.41)	**Prev** (prob: 0.37)	**Max** (prob: 0.50)	**Print** (prob: 0.97)

Step 2: Understanding Question Influence

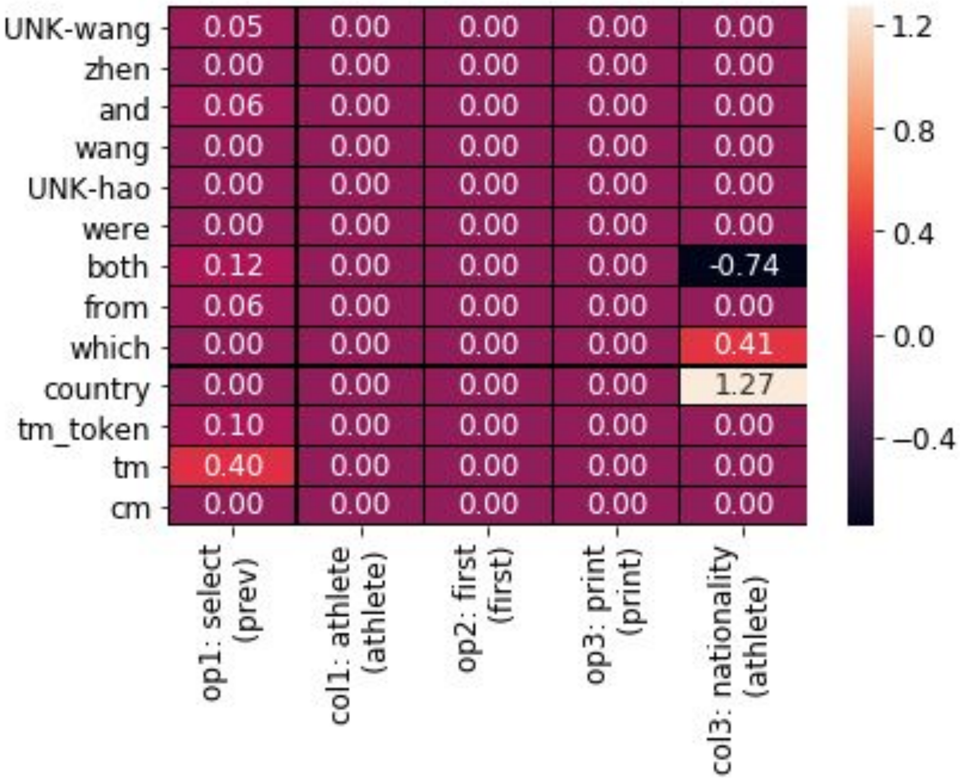
$$\text{col-names} \rightarrow \left\{ < \textbf{ques-words, table-match, col-match} > \rightarrow \mathbf{R}^{\#\mathbf{operators}} \right\}$$

Use Integrated Gradients to attribute selections to **question words**, **table-matches** and **column-matches**

- **Baseline**: empty question
- Attributions will be meaningful only for selections different from those in the table-default program

Visualizing Attributions

Wang zhen and Wang Hao are both from which country?



Visualizing Attributions

Wang zhen and Wang Hao are both from which country?

Attribution is set to 0.0 when selection is same as table-default

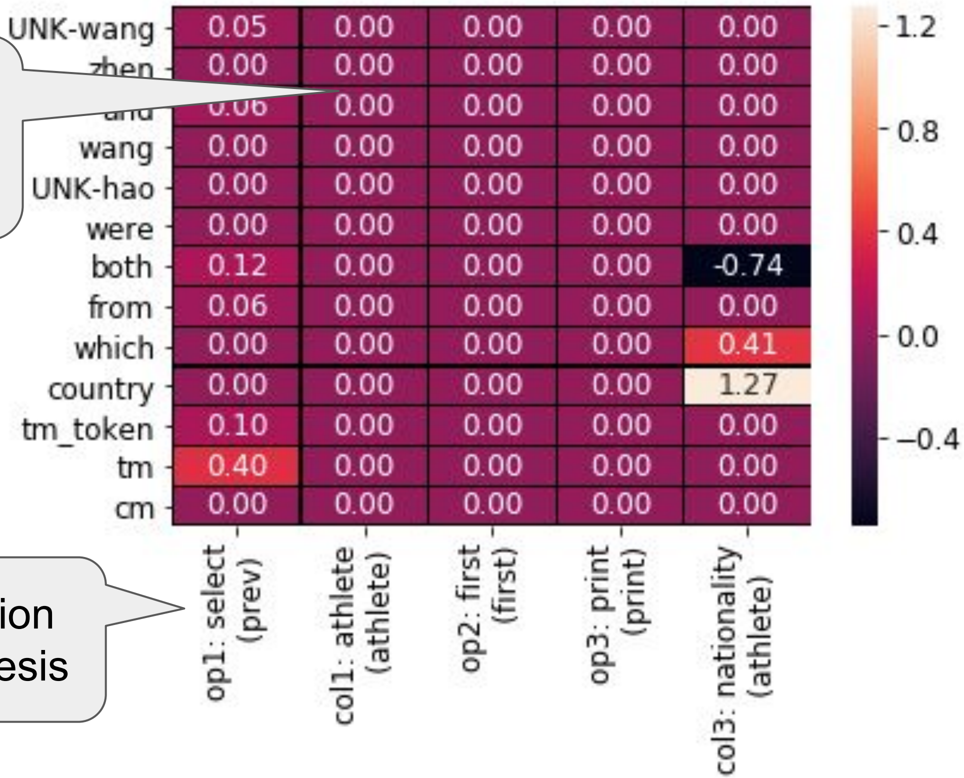
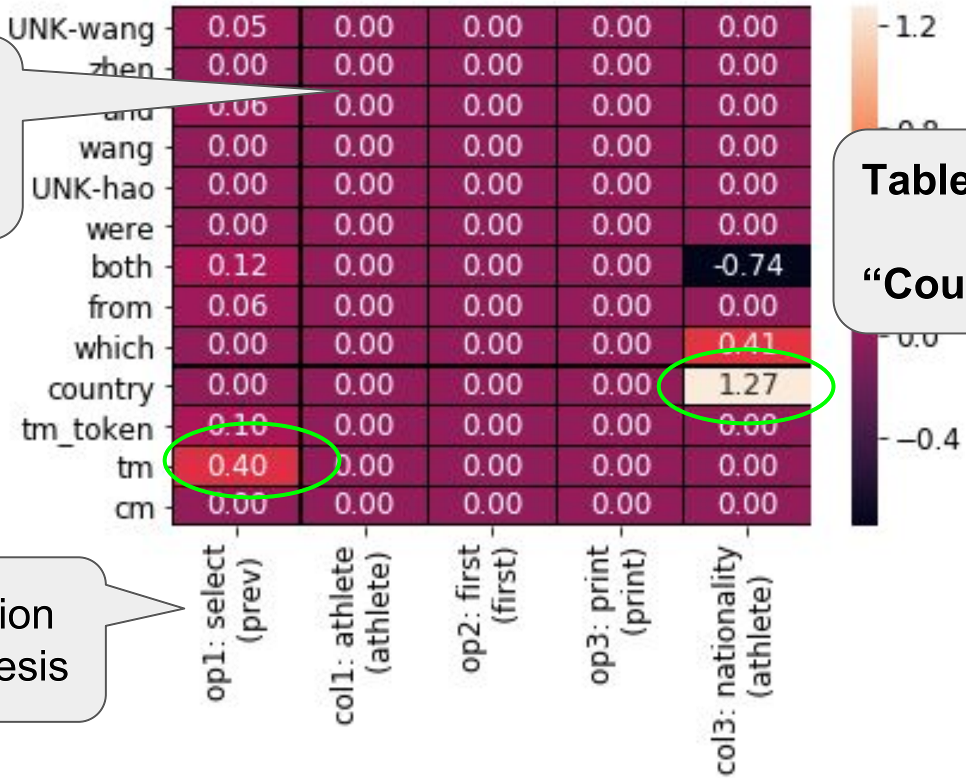


Table-default selection is shown in parenthesis

Visualizing Attributions

Wang zhen and Wang Hao are both from which country?



Attribution is set to 0.0 when selection is same as table-default

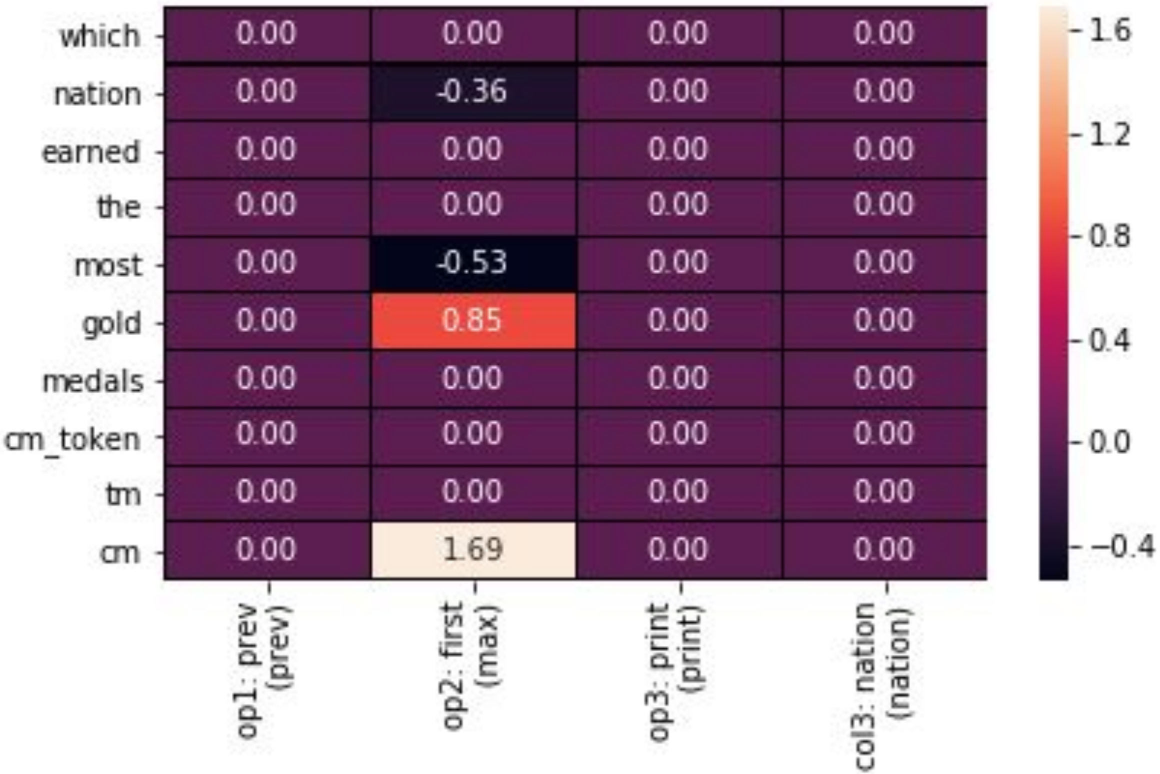
Table-match → Select

“Country” → “Nationality”

Table-default selection is shown in parenthesis

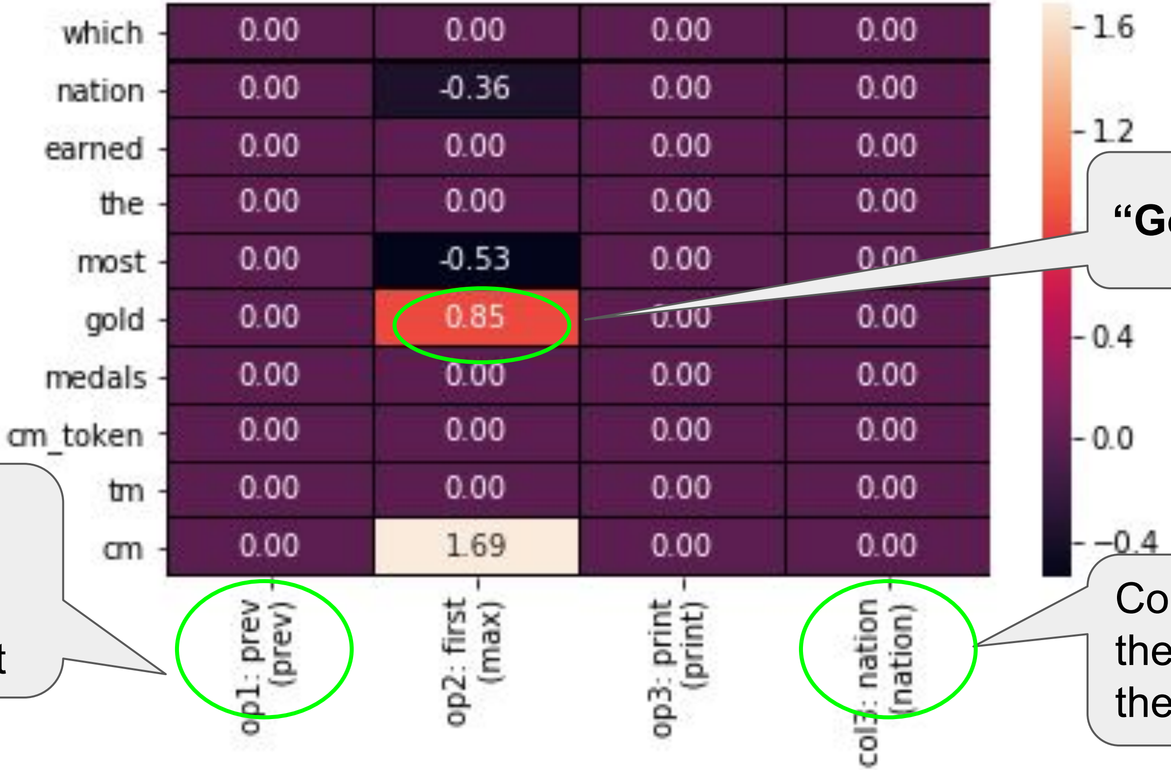
Example 2

Which **nation** earned the most **gold** medals?



Example 2

Which **nation** earned the most **gold** medals?



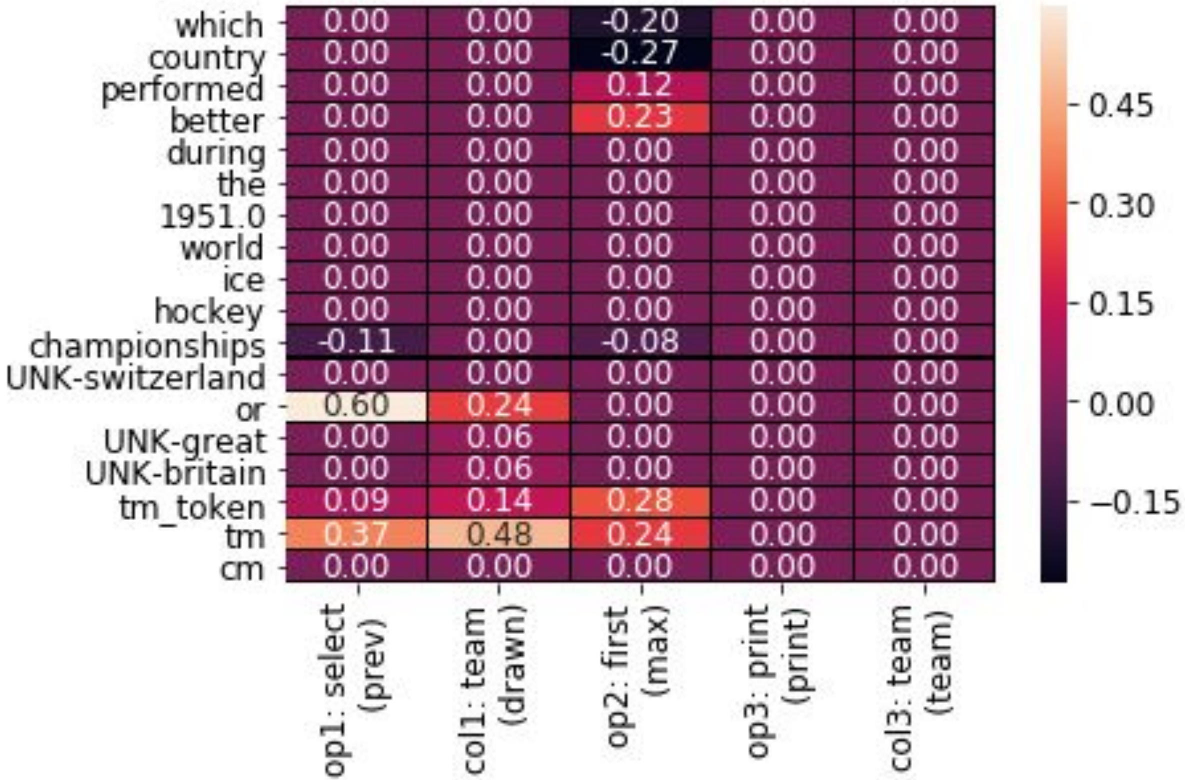
"Gold" → First

Operator Prev comes from the table default

Column "Nation" at the last step also from the table default

Example 3

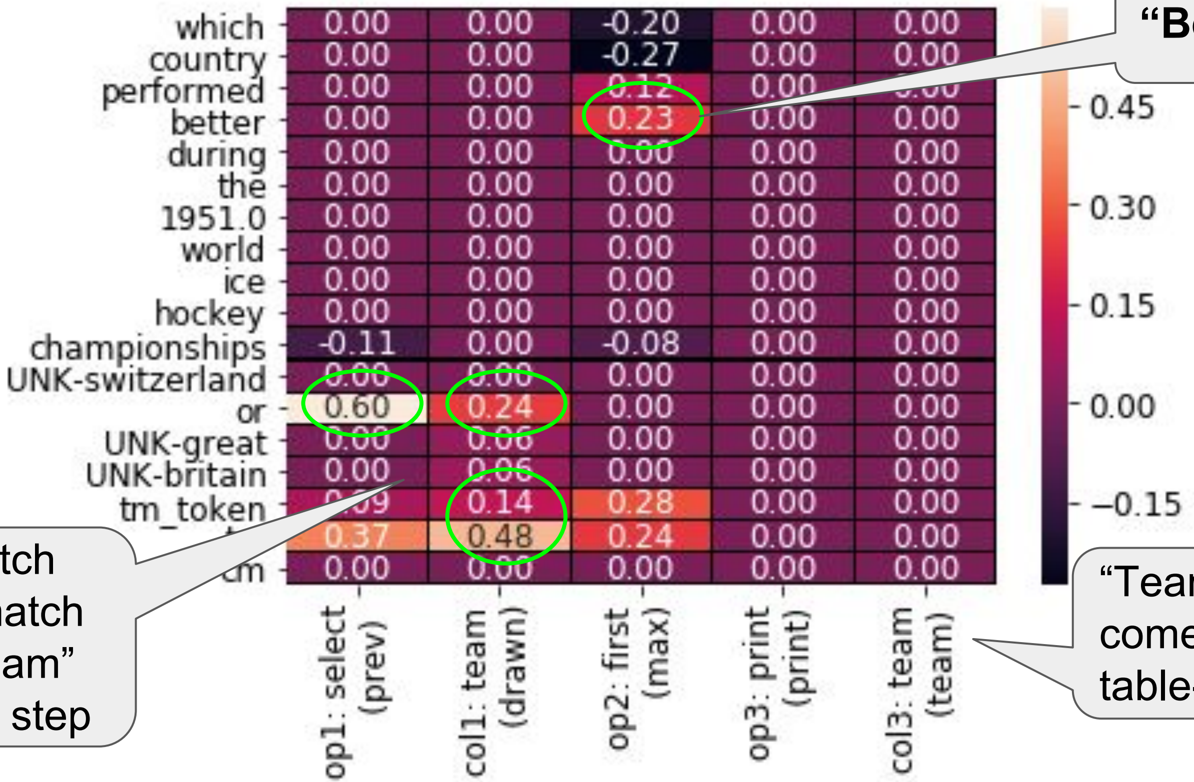
Which country performed **better** during the 1951 word ice hockey championships, switzerland **or** great britain?



Example 3

Which country performed **better** during the 1951 word ice hockey championships, switzerland **or** great britain?

“Better” → First



“Or”, table-match and column-match trigger the “Team” column at first step

“Team” at the last step comes from table-default

Crafting Adversarial Inputs

Can we use (mis-) attributions to craft adversarial inputs against Neural Programmer?

Operator triggers

For each operator, aggregate the top attributed words across questions

Operator	Trigger words
select	[tm_token, how, many, number, of, after, or, total, before, c
count	[how, many, number, of, total, times, is, players, games, difference]
first	[tm_token, first, before, who, listed, after, top, previous, or, most]
reset	[total, many, how, number, the, last, of, listed, first, are]
last	[last, after, tm_token, next, chart, is, the, listed, or, in]
next	[after, tm_token, next, same, listed, comes, not, below, finished, cm_token]
prev	[before, previous, listed, tm_token, above, most, is, what, largest, who]
min	[the, least, amount, which, has, smallest, no, who, school, team]
mfe	[most, cm_token, tm_token, the, competitions, singles, other, many, locomotives, year]
geq	[at, many, had, least, more, number, than, have, players, a]
max	[most, taller, highest, what, area, or, other, building, larger,
print	[cm_token, tm_token, each, who, chart]

Fluff words?

Irrelevant?

Attack 1: Fluff word deletion

- We deleted fluff words from all dev data questions
- Dev accuracy falls from **33.62%** to **28.60%**

Attack 2: Question phrase concatenation

Stick a content-free phrase comprised of semantically-irrelevant trigger words to all questions in the dev set[1].

Original Accuracy: 33.62%

Attack Phrase	Prefix	Suffix
"in not a lot of words"	−12.92%	−23.91%
"in this chart"	−2.89%	−4.23%
"among these rows listed"	−3.42%	−7.31%
"if its all the same"	−11.62%	−15.65%
"above all"	−7.17%	−14.02%
"at the moment"	−2.47%	−7.62%

Union of the 6*2 = 12 attacks drops accuracy from **33.62%** to **5.01%**

[1]Related work: Adversarial examples for evaluating reading-comprehension systems [Jia and Liang, 2017]

Other Research Directions

On Understandability

- Extract rules from a DNN
  - E.g., Can we extract contextual synonyms from Neural Programmer?
- Understand individual dataflow paths
  - For e.g., what influence does the attention path have on the predictions?
  - Allows extracting more focussed rules
- Understand feature interactions
  - Can we automatically extract feature crosses from a deep network?
  - Hessians instead of Gradients?
- Steer DNNs toward **robust** behavior
  - Training data augmentation
  - Intervene with rules, e.g., only attend to non-stop words?

Questions?