November 2020 - CS 395T

TEXAS
The University of Texas at Austin

On Explanations of Question Answering Models

Levi Villarreal, Xi Ye
The University of Texas at Austin

Explanation

Explaining neural models' prediction

Review

the beer was n't what i expected, and i'm not sure it's "true to style", but i thought it was delicious. **a very pleasant ruby red-amber color** with a relatively brilliant finish, but a limited amount of carbonation, from the look of it. aroma is what i think an amber ale should be - a nice blend of caramel and happiness bound together.

Ratings

Look: 5 stars

Smell: 4 stars

(Lei et al., 2016)

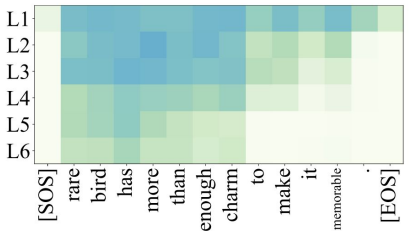
Explaining NLP Models

- Extracted “Rationales” of neural models (Lei et al., 2016)
- Attribution of inputs (Sundararajan et al., 2017; Feng et al., 2018)

Context: QuickBooks sponsored a “Small Business Big Game” contest, in which Death Wish Coffee had a 30-second commercial aired free of charge courtesy of QuickBooks. Death Wish Coffee beat out nine other contenders from across the United States for the free advertisement.

Question:
What company won free advertisement due to QuickBooks contest ?

- Inner-network working mechanism (Guan et al., 2019; Cao et al., 2020)



- Other formats designated to specific tasks (e.g., QA)

Using Explanations

- Understanding the models' behaviours
- Identifying the bias/vulnerability existing in the models
- Improving the generalizability/robustness of models

Explanation of QA Models

- The framework for explanations for general QA

  QED: A Framework and Dataset for Explanations in Question Answering

- The faithfulness of explanation for compositional QA

  Obtaining Faithful Interpretations from Compositional Neural Networks

November 2020 - CS 395T

TEXAS
The University of Texas at Austin

QED: A Framework and Dataset for Explanations in Question Answering

Matthew Lamm, Jennimaria Palomaki, Chris Alberti, Daniel Andor, Eunsol Choi, Livio Baldini Soares, Michael Collins

Levi Villarreal
The University of Texas at Austin

Background

- Modern Question Answering Systems have made much progress in regards to accuracy, but often have no explanation or justification of answers
- Models that that are explainable may have significant trust and debugging benefits (Doshi-Velez and Kim, 2017; Ehsan et al., 2019)

Contributions

1. Introduce QED, a linguistically grounded definition of QA explanations
2. Present a corpus of QED annotations based on the existing Natural Questions dataset (Kwiatkowski et al. 2019)
   - 7638/1353 dev/train examples
3. Propose 4 potential QED related tasks
4. Describe a rater study to show viability of QED to help users discover QA model errors

QED stands for the Latin “quod erat demonstrandum” or “that which was to be shown”

Motivation

Ehsan et al. 2018

"Explainability is important in situations where human operators work alongside autonomous and semi-autonomous systems because it can help build rapport, confidence, and understanding between the agent and its operator. In the event that an autonomous system fails to complete a task or completes it in an unexpected way, explanations help the human collaborator understand the circumstances that led to the behavior, which also allows the operator to make 3 Instances with annotated short answers, omitting table passages. an informed decision on how to address the behavior."

Motivation

- Help users understand, trust and work with a QA system
- Help developers understand, extend and debug QA models
- Mimic known semantic and syntactic categories
- Goal: Define models with faithful explanations

QED Framework

- Given a question and a passage to answer it, a QED explanation is:
  - Identification of a sentence with the answer
  - Identification of matching noun phrases (NP)
  - Confirmation that predicate in sentence matches question predicate

QED Framework - Example

Question: who wrote the film howl's moving castle?
Passage: Howl's Moving Castle is a 2004 Japanese animated fantasy film written and directed by Hayao Miyazaki. It is based on the novel of the same name, which was written by Diana Wynne Jones. The film was produced by Toshio Suzuki.
Answer: Hayao Miyazaki

---

(1) Sentence Selection
Howl's Moving Castle is a 2004 Japanese animated fantasy film written and directed by Hayao Miyazaki.
(2) Referential Equality
the film howl's moving castle = Howl's Moving Castle
(3) Entailment
X is a 2004 Japanese animated fantasy film written and directed by ANSWER. $\vdash$ ANSWER wrote X.

QED Annotation Process

Question: how many seats in university of michigan stadium

Passage: Michigan Stadium, nicknamed “The Big House”, is the football stadium for the University of Michigan in Ann Arbor, Michigan. It is the largest stadium in the United States and the second largest stadium in the world. Its official capacity is 107,601.

QED Annotation Process

1. Single sentence selection

Question: how many seats in university of michigan stadium

Passage: Michigan Stadium, nicknamed “The Big House”, is the football stadium for the University of Michigan in Ann Arbor, Michigan. It is the largest stadium in the United States and the second largest stadium in the world. Its official capacity is 107,601.

Selection: Its official capacity is 107,601

QED Annotation Process

1. Single sentence selection
2. Answer selection

Question: how many seats in university of michigan stadium

Selection: Its official capacity is 107,601$_{A}$

QED Annotation Process

1. Single sentence selection
2. Answer selection
3. Identification of question-sentence noun phrase equalities

Question: how many seats in university of michigan stadium$_1$

Selection: Its$_1$ official capacity is 107,601$_A$

QED Annotation Process

1. Single sentence selection
2. Answer selection
3. Identification of question-sentence noun phrase equalities
4. (Automatic) Extraction of an entailment pattern

Question: how many seats in university of michigan stadium$_1$

Selection: Its$_1$ official capacity is 107,601$_A$

how many seats in X
X's official capacity is ANSWER

QED Annotations for the Natural Questions Corpus

- Natural Questions (NQ) Corpus
  - Google dataset of real search queries and corresponding Wikipedia snippet answers
- Focus on questions with a passage and short answer
- Exclude tables
- Before performing QED annotations, classified examples into 3 groups
  1. Valid short answer and appropriate for QED annotations
  2. Valid short answer but not appropriate for QED annotations
  3. No valid short answer in text (NQ error)

QED Annotations for the Natural Questions Corpus

- 3 expert annotators
- 7638 training and 1353 dev examples
- On a common set of 100 examples from dev set
  - Classification accuracy of 73.9%
  - Average pairwise F1 on mention identification/mention alignment was 88.4 and 84.1 respectively

Analysis of Referential Expressions

Types of referential expressions

- Proper names *e.g. The Office*
- Non-anaphoric definite NPs *e.g. POTUS*
- Anaphoric definite NPs *e.g. The series*
- Generics *e.g. a dead zone*
- Pronouns *e.g. They*
- Bridging *e.g. was the winner of it*
- Misc

Analysis of Referential Expressions

	Referential Link Count			
	0	1	2	3
Instances	54	649	294	6

Qu. \ Ps.	P	N	A	G	Pn	B	M	T
Proper	44	0	16	0	9	4	0	73
Def. (Non-Ana)	4	6	4	0	0	1	1	16
Def. (Ana)	0	1	1	0	0	0	0	2
Generic	0	0	0	6	0	0	0	6
Pronoun	0	0	0	0	0	0	0	0
Bridge	0	0	0	0	0	0	0	0
Misc	2	0	0	0	0	0	1	3
Total	50	7	21	6	9	5	2	100

Proposed Tasks and Baselines

- Intention of the QED dataset is to spur research into QED based tasks and models
- Introduce four potential modeling tasks using QED data and describe baseline approaches for the first two

Proposed Tasks and Baselines

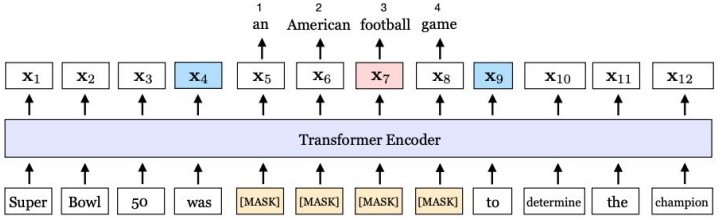
Terminology

- q: question from NQ corpus
- d: wikipedia page
- c: long answer within d
- a: short answer within c
- e: QED explanation

Four tasks

1. Predict QED explanation
   - $\hat{e} = f(q,d,c,a)$
2. Predict answer and QED explanation
   - $(\hat{a},\hat{e}) = f(q,d,c)$
3. Predict a long and short answer, and QED explanation
   - $(\hat{c},\hat{a},\hat{e}) = f(q,d)$
4. Predict a long and short answer, and QED explanation faithful to the underlying model
   - $(\hat{c},\hat{a},\hat{e}) = f(q,d)$
   - Requires a faithfulness measure

QED Explanation Task Baseline



- SpanBERT - Joshi et al (2019)
  - Pre-training method to represent and predict spans of text
  - Extends and outperforms BERT on span selection
  - Mask random contiguous spans rather than individual tokens
  - Predict the entire content of a masked span
- End-to-end Neural Coreference Resolution - Lee et al (2017)
  - End-to-end model that classifies entity mentions spans
  - Learn a conditional probability distribution $P(y_1,\ldots,y_n \mid D)$ which is most likely to produce the correct clustering
  - Interpretable model

$$P(y_1,\ldots,y_N \mid D) = \prod_{i=1}^{N} P(y_i \mid D)$$

$$= \prod_{i=1}^{N} \frac{\exp(s(i,y_i))}{\sum_{y' \in \mathcal{Y}(i)} \exp(s(i,y'))}$$

QED Explanation Task Baseline

- Model input consists of question **q**'s tokens, passage **c**'s tokens, and page title **t**'s tokens
- Model is tasked with predicting the referential equality annotations
- Post-process pair clusters of questions and passages based on positionality
  - Links between noun phrases in the question and passage
  - Links between noun phrases in the question and implicit argument in the passage

QED Explanation Task Baseline Results

- **Zero-shot** - Used CoNLL OntoNotes coreference dataset (Pradhan et al., 2012) to train SpanBERT model
- **Fine-tuned** - further trained the model with the training portion of QED data converted into coreference format

	Mention Identification			Mention Alignment		
	P	R	F1	P	R	F1
zero-shot	59.0	35.6	44.4	47.7	28.8	35.9
fine-tuned	76.8	68.8	72.6	68.4	61.3	64.6

Answer and QED Explanation Task Baseline

- Builds upon model from task one
- Compose an existing QA model with an answer agnostic model

$$
\begin{aligned}
& p(a, e|q, d, c; \theta) \\
= \; & p^{(1)}(a|q, d, c; \theta^{(1)}) p^{(2)}(e|a, q, d, c; \theta^{(2)})
\end{aligned}
$$

- Train $p^{(1)}$ and $p^{(2)}$ in a multitask fashion, by minimizing the weighted sum of the QA and coreference cross entropy losses

Answer and QED Explanation Task Baseline Results

- **QED-only** - fine-tunes $p^{(2)}$ on the QED training set only
- **QA-only** - fine-tunes $p^{(1)}$ on all the paragraphs of the NQ dataset that contain a short answer
- **QA+QED** - fine-tunes both $p^{(1)}$ and $p^{(2)}$ on all NQ and QED data

	Mention Identification			Mention Alignment			Answer Accuracy
	P	R	F1	P	R	F1	
QED-only	74.1	63.8	68.6	63.6	54.9	58.9	-
QA-only	-	-	-	-	-	-	73.4
QA+QED	77.5	64.6	70.5	68.6	57.3	62.4	74.5

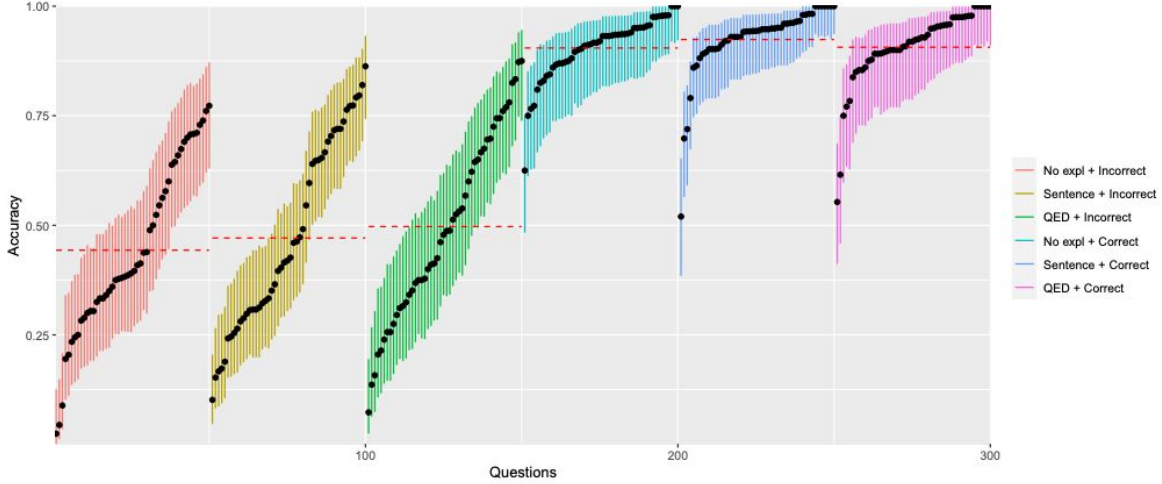
Rater Study

- Hypothesize that QED explanations should improve a user's ability to discover QA model errors
- Task
  - Given a question, passage and candidate answer, raters asses whether the answer is correct
  - 3 rater groups
    - **None** - question, passage, highlighted answer span
    - **Sentence** - additional highlighting of answer sentence
    - **QED** - additional referential highlighting

Rater Study Results

- Incorr. far more likely to be marked wrong
- Highest improvement comes from highlighting the sentence

	Accuracy			F1
	All	Corr	Incorr/Pred/Ref	Incorr
None	67.5	90.4	44.3/43.9/44.7	57.6
Sentence	69.7	**92.4**	47.1/46.1/48.0	60.9
QED	**70.2**	90.6	**49.7/48.2/51.0**	**62.5**



Future QED Work

- Baseline models for task three and four
- Ambiguous questions - resolving referential ambiguities
  - Provide multiple answers and explanations for each
- Complex referential Equalities
  - Be able to handle complex chaining of referential equalities throughout text
- Different data structure
  - Support table data

Recent Related Work

- Teaching Machine Comprehension with Compositional Explanations (Qinyuan Ye, X. Huang, X. Ren)
  - Uses referential equality framework to train a QA model on a smaller corpus of annotated data
  - Use semi-structured explanations to better train a reading comprehension model, rather than predicting an explanation itself
  - More challenging task

Discussion

- What constitutes a good explanation?
- Are the benefits of QED apparent enough to warrant further research?
- How can an explanation be faithful to a given model?
- How can future QED learn from research in the more general field of human-computer interaction and explainable AI?

November 2020 - CS 395T

TEXAS
The University of Texas at Austin

Obtaining Faithful Interpretations from Compositional Neural Networks

Sanjay Subramanian, Ben Bogin, Nitish Gupta, Tomer Wolfson, Sameer Singh, Jonathan Berant, Matt Gardner

Xi Ye
The University of Texas at Austin

Visual Question Answering

Are all dogs black?

(Shur et al., 2019)

Prediction in One-Shot

Are all dogs black?

LXMERT

False

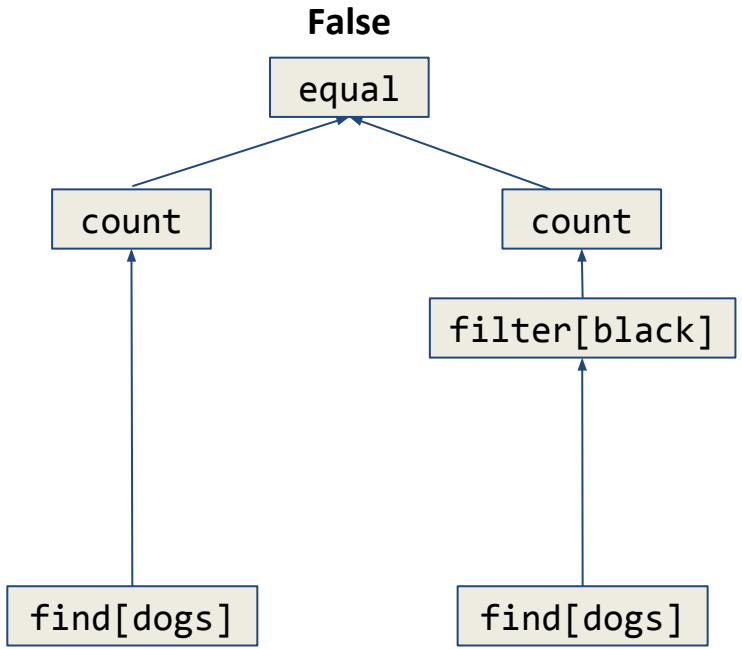
? black box

(Tan and Bansal, 2019)

Compositional Reasoning (NMN)

Are all dogs black?

Parse

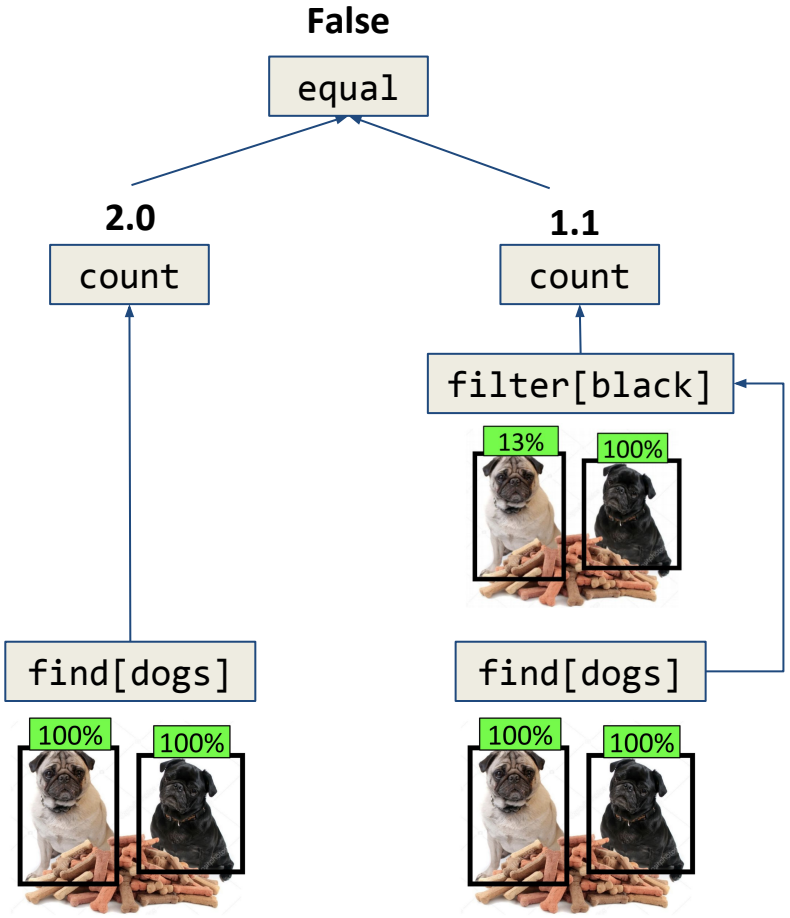


(Andreas et al., 2016)

? Ideal Execution Trace

Are all dogs black?

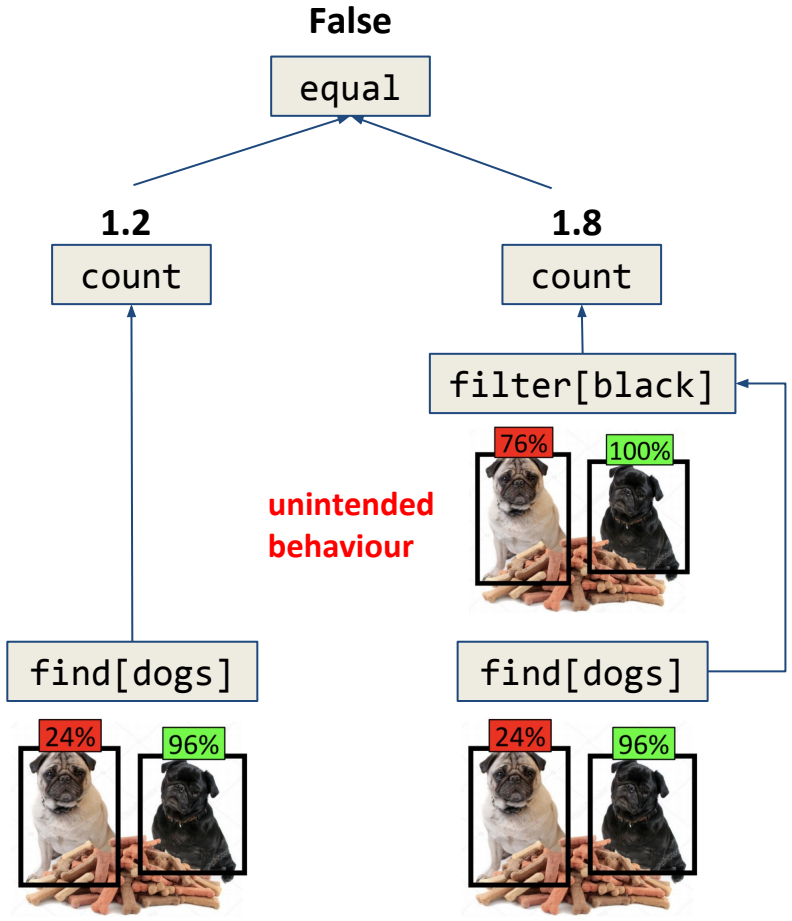
Parse



Unfaithful Execution Trace

Are all dogs black?

Parse

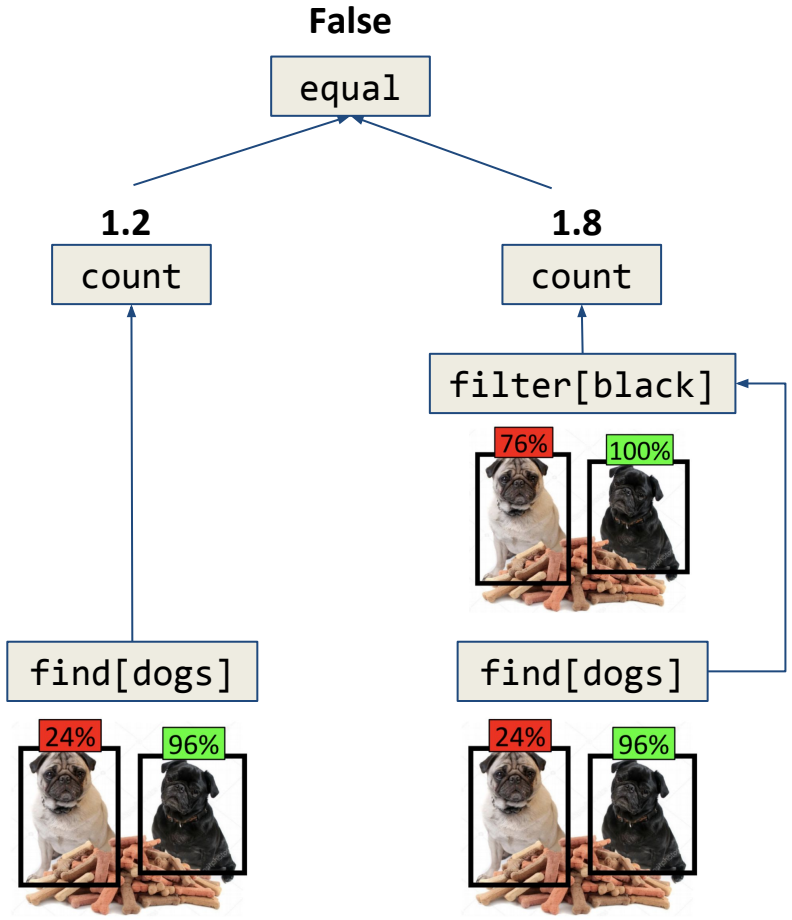


Unfaithful Execution Trace

- Modules not performing the intended behaviours
- Programs cannot serve as faithful explanations

Contributions

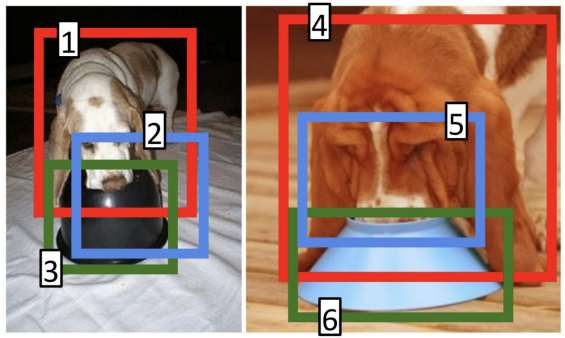
- Systematic evaluation of faithfulness of intermediate module execution
- Methods for improving module-wise faithfulness



Tasks

NLVR2

two dogs are touching a food dish with there face



DROP

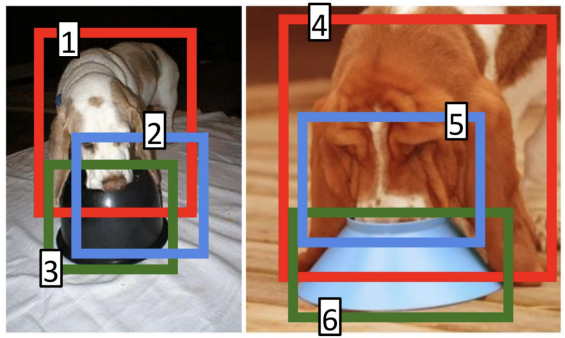
who threw the longest touchdown pass in the second half?

In the first quarter, the Texans trailed early after QB Kerry Collins threw a 19-yard TD pass [1] to WR Nate Washington. Second quarter started with kicker Neil Rackers made a 37-yard field goal, and the quarter closed with kicker Rob Bironas hitting a 30-yard field goal. The Texans tried to cut the lead with QB Matt Schaub getting a 8-yard TD pass [2] to WR Andre Johnson, but the Titans would pull away with RB Javon Ringer throwing a 7-yard TD pass [3]. The Texans tried to come back into the game in the fourth quarter, but only came away with Schaub [4] throwing a 12-yard TD pass [5] to WR Kevin Walter.

NLVR2 vs VQA

NLVR2

two dogs are touching a food dish with there face



- reasoning over image pairs
- compositional reasoning & relation

VQA

what color are her eys?

- identification of object properties and few spatial relations

Gold (from Break dataset) programs used (Wolfson et al., 2020)

Evaluation of Faithfulness

Module-wise Faithfulness

- Prior work
  - Assume modules perform intended functions
  - Qualitatively analyze the intermediate outputs of several examples

Module-wise Faithfulness

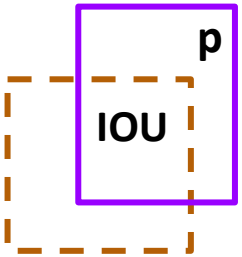
- Prior work
  - Assume modules perform intended functions
  - Qualitatively analyze the intermediate outputs of several examples

- This Work
  - Collect gold program and gold outputs
  - Systematically evaluate the correctness of each module

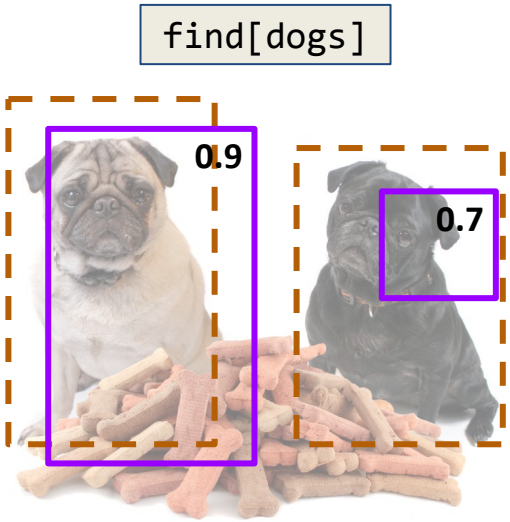
Faithfulness of Visual-NMN

Module Output bounding boxes with probabilities

Gold human annotated boulding boxes



Match Criterion
p > 0.5 and IOU > 0.5



- **Data Collection:** Expert-annotated intermediate outputs for 536 programs
- **Metrics:** precision, recall, F1

Faithfulness of Text-NMN

Module Output
probabilities over tokens

Gold
spans

$$[s_1, \ldots, s_N]$$

$$s_i = (t_{\text{s}}^i, t_{\text{e}}^i)$$

who threw the longest touchdown pass in the second half?

In the first quarter, the Texans trailed early after QB Kerry Collins threw a 19-yard TD pass to WR Nate Washington. Second quarter started with kicker Neil Rackers made a 37-yard field goal, and the quarter closed with kicker Rob Bironas hitting a 30-yard field goal. The Texans tried to cut the lead with QB Matt Schaub getting a 8-yard TD pass to WR Andre Johnson, but the Titans would pull away with RB Javon Ringer throwing a 7-yard TD pass. The Texans tried to come back into the game in the fourth quarter, but only came away with Schaub throwing a 12-yard TD pass to WR Kevin Walter.

- **Data Collection:** Intermediate outputs for 215 programs
- **Metrics:** Cross entropy between gold and predicted span probabilities

$$I = -\sum_{i=1}^{N} \left(\log \sum_{j=t_{\text{s}}^i}^{t_{\text{e}}^i} p_{\text{att}}^j \right)$$

Improving Module-wise Faithfulness

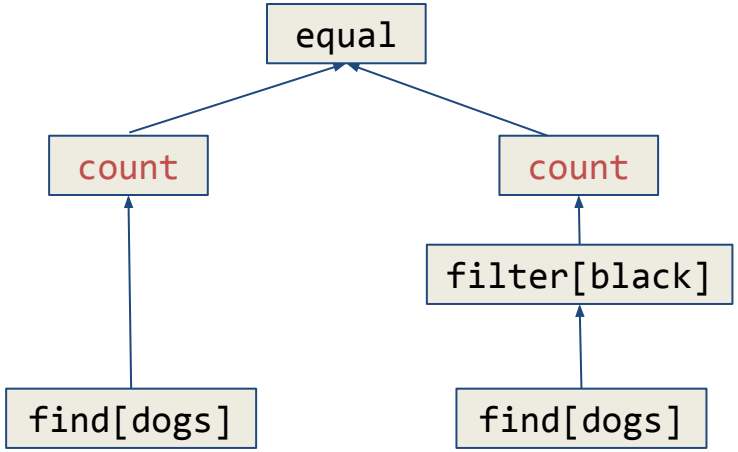
Factors affecting Module-wise Faithfulness

- Choice & implementation of modules
- Supervising module outputs
- Decontextualized representations

Count in Visual-NMN

- Count modules are often placed before output modules, mediating the backpropagation of other modules
- A very expressive count module might learn to perform tasks designated for descendant modules (e.g., find and filter), hurting faithfulness

Are all dog black?



Trade-off

Performance ⟶

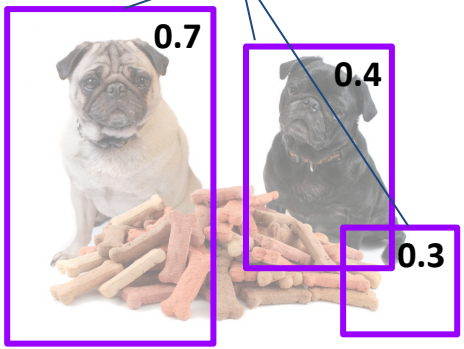
⟵ Faithfulness

too limited

Sum Count
(no parameters)

1.4

math +



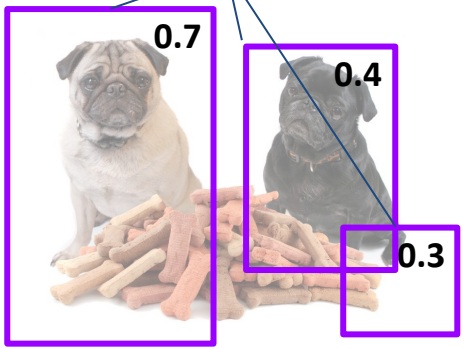
middle ground

Graph-Count
(<300 parameters)

1.7

NN For Counting

graph representing the boxes



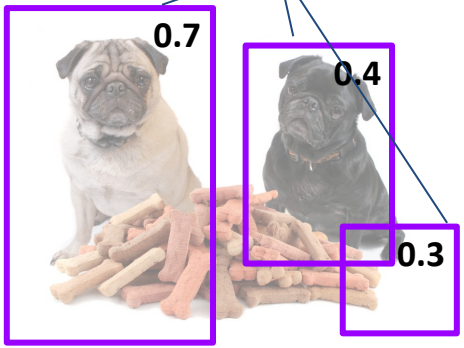
too expressive hurting faithfulness

Layer-Count

1.8

FFNN

weighted sum of representations

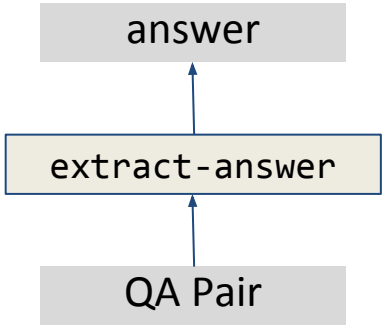


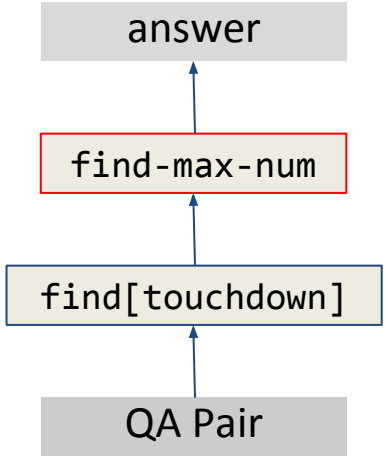
Module Choice in Text-NMN

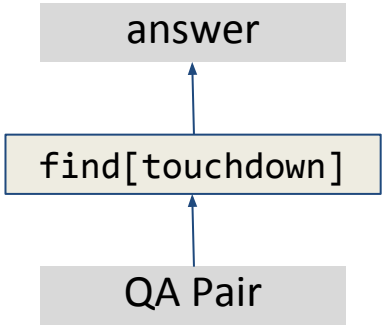
- investigating the impacts of allowing non-atomic reasoning

introducing modules (extract-answer)
predicting the answer in one-shot

removing sorting and
comparison modules







what is the longest touchdown pass?

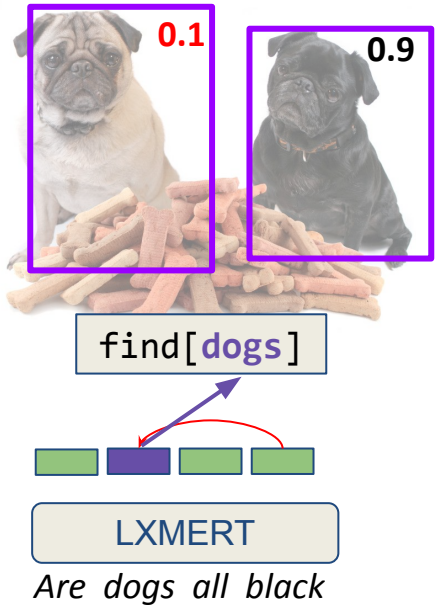
Supervised Module Outputs

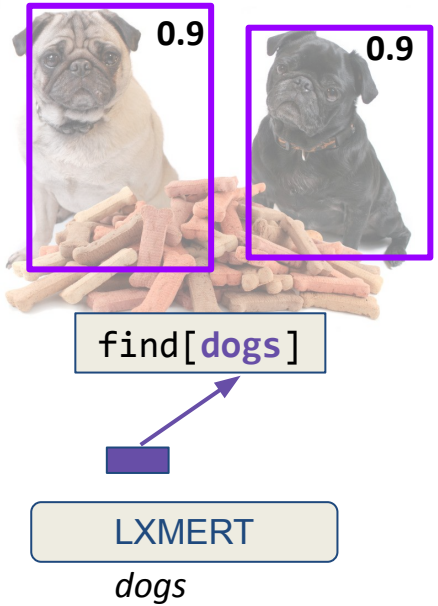
- **Visual:** pretrain find and filter modules with gold intermediate supervision on GQA dataset
- **Text:** use heuristically generated supervision as auxiliary supervision

(Hudson and Manning, 2019)

Decontextualized Representations

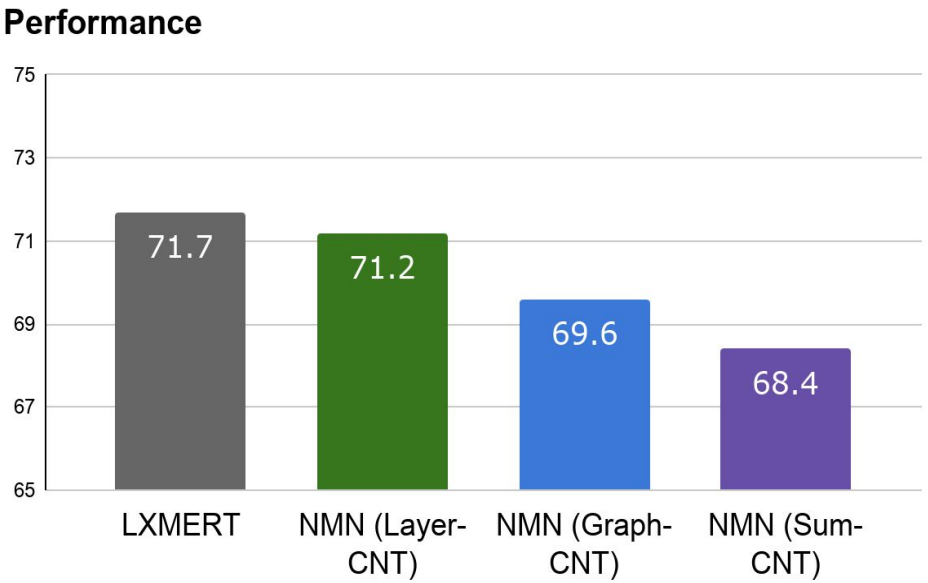
- Contextualized token representations know the **global** information
- Decontextualizing by encoding each token independently

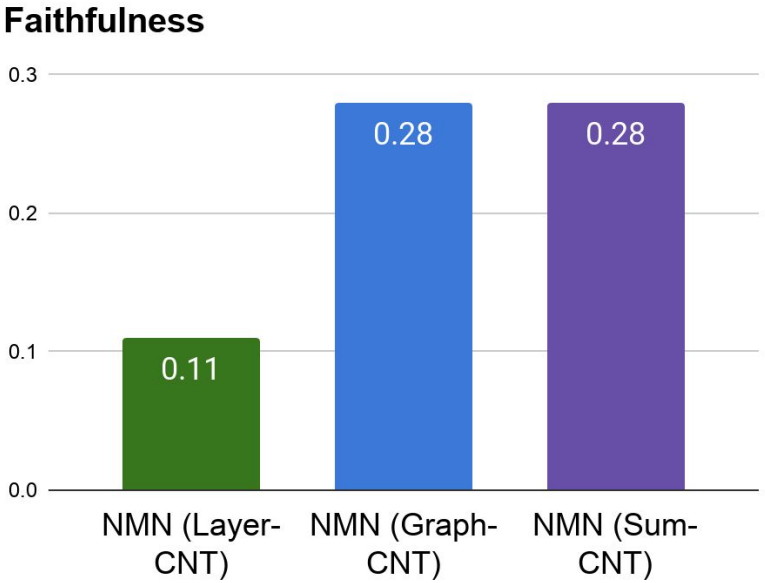




Experiments

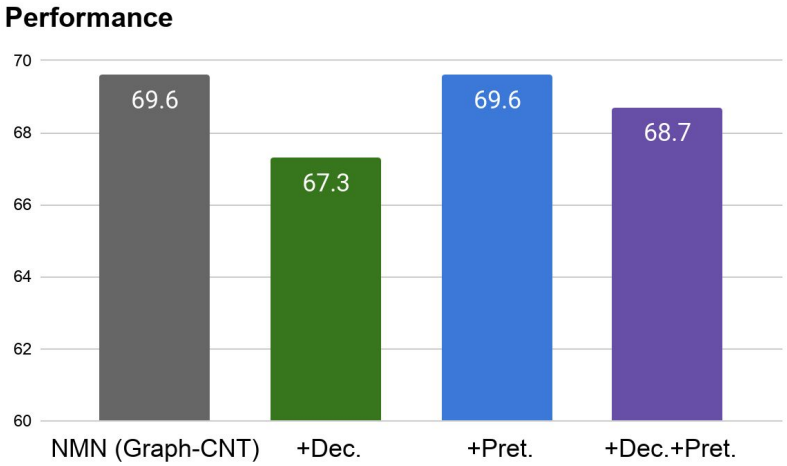
Visual: Impact of Module Choice

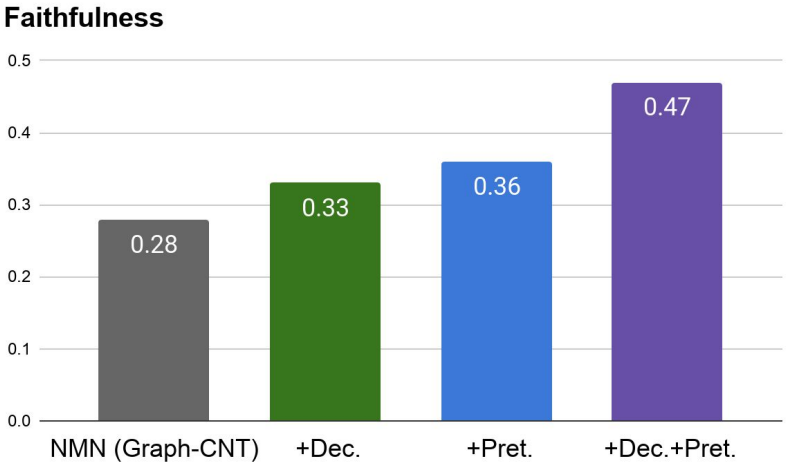




- More expressive models achieve higher performance but are less faithful
- NMN (Graph-CNT) maximally retains the performance while achieving on par faithfulness compared to NMN (Sum-CNT)

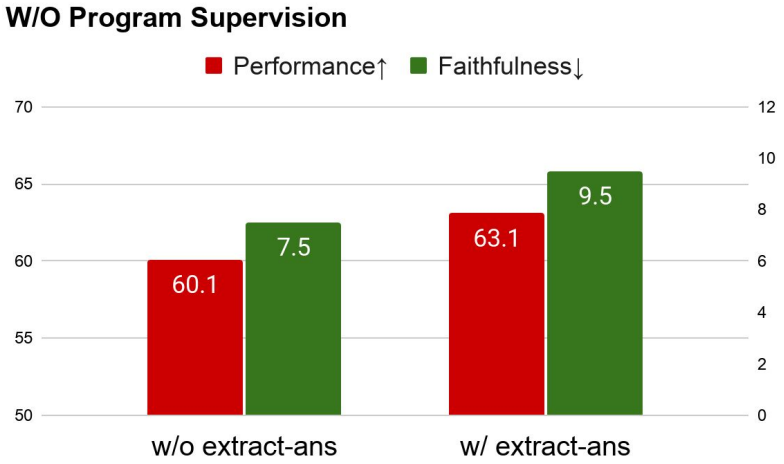
Visual: Pretraining and Decontextualizing

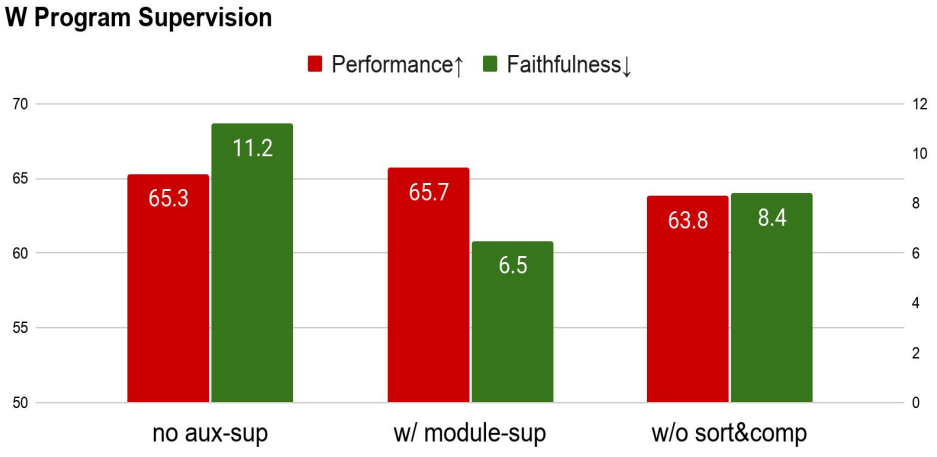




- Both decontextualizing and pretraining improves faithfulness
- Combining the two achieves the best F1 with at minimal expense of accuracy

Results on DROP





- Introducing extract-ans improves performance but hurts faithfulness
- Using module-supervision improves both performance and faithfulness
- Removing sort/comp improves performance but hurts faithfulness, similar to introducing extract-ans

Analysis: Systematic Generalization

- Test on out-of-domain data points excluded from training set
- **Visual**
  - Train on programs with length within 7, test on programs with length longer than 7
  - Faithful models do not empirically generalize better
- **Text**
  - Train on program involving max, test on program involving min
  - Compare models trained using gold program with/without module-output supervision
  - Intermediate supervision improves both faithfulness and generalization

Error Analysis

- Long-tail objects (e.g., safety pin)
- Hard-to-annotate objects (e.g., grass)
- Relocate module: less annotation and is often executed with small objects

Wrap-up

- Naively trained modules in NMN should not be assumed to be faithful
- Additional annotation can improve faithfulness while retaining good performance

Related Reading

Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness?（Jacovi and Goldberg, ACL'20)

Benefits of Intermediate Annotations in Reading Comprehension (Dua et al., ACL'20)

Latent Compositional Representations Improve Systematic Generalization in Grounded Question Answering (Bogin et al., TACL'20)

Discussion Questions

- Why do faithful models fail on the systematic generalization tasks on NLVR2?
- Is the trade-off between interpretability and performance inevitable?
- Curious case in module-wise faithfulness

Model	**Performance** (Accuracy)	**Overall Faithful.** (↑)			Module-wise Faithfulness F_1 (↑)			
		Prec.	Rec.	F_1	find	filter	with-relation	relocate
LXMERT	**71.7**							
Upper Bound		1	0.84	0.89	0.89	0.92	0.95	0.75
NMN w/ Layer-count	71.2	0.39	0.39	0.11	0.12	0.20	0.37	**0.27**
NMN w/ Sum-count	68.4	**0.49**	0.31	0.28	0.31	0.32	0.44	0.26
NMN w/ Graph-count	69.6	0.37	0.39	0.28	0.31	0.29	0.37	0.19
NMN w/ Graph-count + decont.	67.3	0.29	0.51	0.33	0.38	0.30	0.36	0.13
NMN w/ Graph-count + pretraining	69.6	0.44	0.49	0.36	0.39	0.34	0.42	0.21
NMN w/ Graph-count + decont. + pretraining	68.7	0.42	**0.66**	**0.47**	**0.52**	**0.41**	**0.47**	0.21