

# Inducing Interpretable Causal Structures in Neural Networks

Christopher Potts

Joint work with Atticus Geiger, Zhengxuan Wu, Hanson Lu, Josh Rozner, Elisa Kreiss, Thomas Icard, and Noah Goodman

Stanford Linguistics and the Stanford NLP Group

UT Austin Linguistics, March 7, 2022



# Semantic insights in NLP models

# Semantic insights in NLP models: 1980s

Chat-80; Warren and Pereira 1982

## Semantic insights in NLP models: 1980s

```

/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt, Pronoun, []), Agmt, NPCase, def, _, Set, Nil) -->
  {is_pp(Set)},
  pers_pron(Pronoun, Agmt, Case),
  {empty(Nil), role(Case, decl, NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep, Arg), Case, Set, Mask) -->
  prep(Prep),
  {prep_case(NPCase)},
  np(Arg, _, NPCase, _, Case, Set, Mask) .

```

Chat-80; Warren and Pereira 1982

## Semantic insights in NLP models: 1980s

```

/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt, Pronoun, []), Agmt, NPCase, def, _, Set, Nil) -->
  {is_pp(Set)},
  pers_pron(Pronoun, Agmt, Case),
  {empty(Nil), role(Case, decl, NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep, Arg), Case, Set, Mask) -->
  prep(Prep),
  {prep_case(NPCase)},
  np(Arg, _, NPCase, _, Case, Set, Mask) .

```

- Which country bordering the Mediterranean borders a country that is bordered by a country whose population exceeds the population of India?

Chat-80; Warren and Pereira 1982

## Semantic insights in NLP models: 1980s

```
/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt, Pronoun, []), Agmt, NPCase, def, _, Set, Nil) -->
  {is_pp(Set)},
  pers_pron(Pronoun, Agmt, Case),
  {empty(Nil), role(Case, decl, NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep, Arg), Case, Set, Mask) -->
  prep(Prep),
  {prep_case(NPCase)},
  np(Arg, _, NPCase, _, Case, Set, Mask) .
```

- Which country bordering the Mediterranean borders a country that is bordered by a country whose population exceeds the population of India? [turkey](#).

## Semantic insights in NLP models: 1980s

```
/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt, Pronoun, []), Agmt, NPCase, def, _, Set, Nil) -->
  {is_pp(Set)},
  pers_pron(Pronoun, Agmt, Case),
  {empty(Nil), role(Case, decl, NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep, Arg), Case, Set, Mask) -->
  prep(Prep),
  {prep_case(NPCase)},
  np(Arg, _, NPCase, _, Case, Set, Mask) .
```

- Which country bordering the Mediterranean borders a country that is bordered by a country whose population exceeds the population of India? [turkey](#).
- How far is London from Paris?

Chat-80; Warren and Pereira 1982

## Semantic insights in NLP models: 1980s

```
/* Sentences */
sentence(S) --> declarative(S), terminator(.) .
sentence(S) --> wh_question(S), terminator(?) .
sentence(S) --> yn_question(S), terminator(?) .
sentence(S) --> imperative(S), terminator(!) .

/* Noun Phrase */
np(np(Agmt, Pronoun, []), Agmt, NPCase, def, _, Set, Nil) -->
  {is_pp(Set)},
  pers_pron(Pronoun, Agmt, Case),
  {empty(Nil), role(Case, decl, NPCase)}.

/* Prepositional Phrase */
pp(pp(Prep, Arg), Case, Set, Mask) -->
  prep(Prep),
  {prep_case(NPCase)},
  np(Arg, _, NPCase, _, Case, Set, Mask) .
```

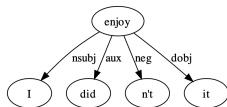
- Which country bordering the Mediterranean borders a country that is bordered by a country whose population exceeds the population of India? [turkey](#).
- How far is London from Paris? [I don't understand!](#)

Chat-80; Warren and Pereira 1982

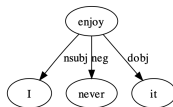
# Semantic insights in NLP models: 1990s

# Semantic insights in NLP models: 1990s

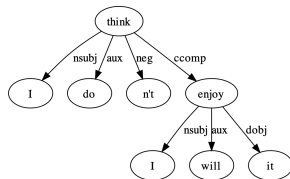
I didn't enjoy it.



I never enjoy it.

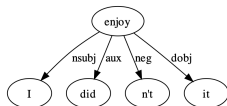


I don't think I will enjoy it.

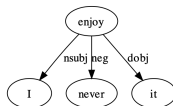


# Semantic insights in NLP models: 1990s

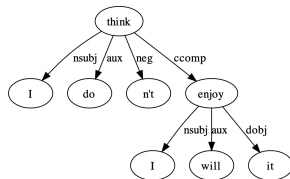
I didn't enjoy it.



I never enjoy it.



I don't think I will enjoy it.



$$\text{neg}(x, *) \Rightarrow x\_neg$$

$$\text{neg}(x, y) \wedge \text{ccomp}(x, z) \Rightarrow x\_neg, z\_neg$$

# Semantic insights in NLP models: 2000s

Zettlemoyer and Collins 2005

# Semantic insights in NLP models: 2000s

$$\begin{array}{c}
 \text{a) } \frac{\text{Utah} \quad \text{borders} \quad \text{Idaho}}{\frac{\frac{NP \quad (S \setminus NP) / NP \quad NP}{\lambda x. \lambda y. \text{borders}(y, x)} \quad \text{idaho}}{(S \setminus NP)} \quad \lambda y. \text{borders}(y, \text{idaho})} > \\
 \frac{}{S} \\
 \text{borders}(\text{utah}, \text{idaho}) <
 \end{array}$$

$$\begin{array}{c}
 \text{b) } \frac{\text{What} \quad \text{states} \quad \text{border} \quad \text{Texas}}{\frac{\frac{\frac{(S / (S \setminus NP)) / N \quad N \quad (S \setminus NP) / NP \quad NP}{\lambda f. \lambda g. \lambda x. f(x) \wedge g(x)} \quad \lambda x. \text{state}(x)} \quad \lambda x. \lambda y. \text{borders}(y, x)} \quad \text{texas}}{(S \setminus NP)} \quad \lambda y. \text{borders}(y, \text{texas})} > \\
 \frac{\frac{S / (S \setminus NP)}{\lambda g. \lambda x. \text{state}(x) \wedge g(x)} \quad \lambda y. \text{borders}(y, \text{texas})}{S} \\
 \lambda x. \text{state}(x) \wedge \text{borders}(x, \text{texas}) >
 \end{array}$$

Zettlemoyer and Collins 2005

# Semantic insights in NLP models: 2000s

a) 
$$\frac{\frac{\text{Utah}}{NP} \quad \frac{\text{borders}}{(S \setminus NP)/NP} \quad \frac{\text{Idaho}}{NP}}{\lambda x. \lambda y. \text{borders}(y, x)} \frac{NP}{idaho} >$$

$$\frac{(S \setminus NP)}{\lambda y. \text{borders}(y, idaho)} >$$

$$\frac{S}{\text{borders}(utah, idaho)} <$$

b) 
$$\frac{\frac{\text{What}}{(S/(S \setminus NP))/N} \quad \frac{\text{states}}{N} \quad \frac{\text{border}}{(S \setminus NP)/NP} \quad \frac{\text{Texas}}{NP}}{\lambda f. \lambda g. \lambda x. f(x) \wedge g(x)} \frac{N}{\lambda x. \text{state}(x)} >$$

$$\frac{S/(S \setminus NP)}{\lambda g. \lambda x. \text{state}(x) \wedge g(x)} >$$

$$\frac{(S \setminus NP)}{\lambda y. \text{borders}(y, texas)} >$$

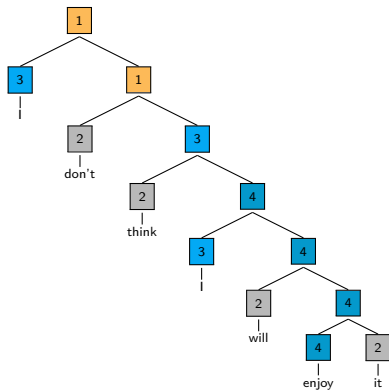
$$\frac{S}{\lambda x. \text{state}(x) \wedge \text{borders}(x, texas)} >$$

states :=  $N : \lambda x. \text{state}(x)$   
 major :=  $N/N : \lambda f. \lambda x. \text{major}(x) \wedge f(x)$   
 population :=  $N : \lambda x. \text{population}(x)$   
 cities :=  $N : \lambda x. \text{city}(x)$   
 rivers :=  $N : \lambda x. \text{river}(x)$   
 run through :=  $(S \setminus NP)/NP : \lambda x. \lambda y. \text{traverse}(y, x)$   
 the largest :=  $NP/N : \lambda f. \arg \max(f, \lambda x. \text{size}(x))$   
 river :=  $N : \lambda x. \text{river}(x)$   
 the highest :=  $NP/N : \lambda f. \arg \max(f, \lambda x. \text{elev}(x))$   
 the longest :=  $NP/N : \lambda f. \arg \max(f, \lambda x. \text{len}(x))$

Figure 6: Ten learned lexical items that had highest associated parameter values from a randomly chosen development run in the Geo880 domain.

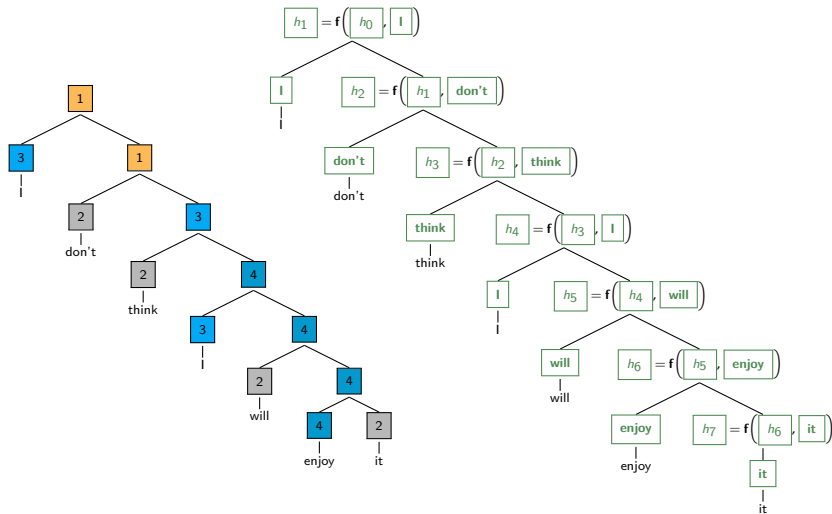
# Semantic insights in NLP models: 2010s

# Semantic insights in NLP models: 2010s



Socher et al. 2013

# Semantic insights in NLP models: 2010s

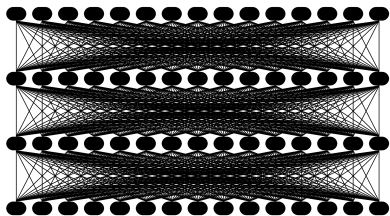


Socher et al. 2013

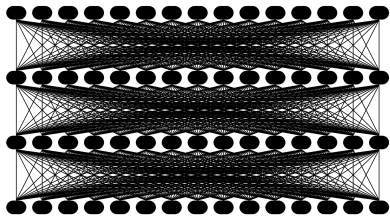


# Semantic insights in NLP models: 2020s

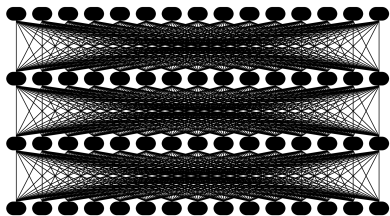
# Semantic insights in NLP models: 2020s



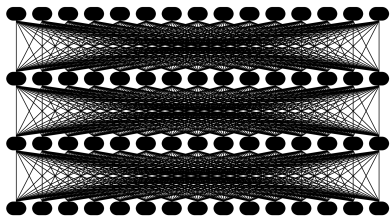
# Semantic insights in NLP models: 2020s



# Semantic insights in NLP models: 2020s



# Semantic insights in NLP models: 2020s



# Semantics in the era of deep learning

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP?

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? **No!**

## Semantics in the era of deep learning

A low point for connections between linguistics and NLP? **No!**

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? [No!](#)

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? [No!](#)

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations
- High-dimensional representations

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? [No!](#)

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations
- High-dimensional representations
- Context-dependent representations

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? [No!](#)

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations
- High-dimensional representations
- Context-dependent representations
- Holistic representations

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? **No!**

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations
- High-dimensional representations
- Context-dependent representations
- Holistic representations
- Ambitions to interpret even the most complex language

# Semantics in the era of deep learning

A low point for connections between linguistics and NLP? [No!](#)

Modern NLP systems based in deep learning (a.k.a. neural networks, connectionism):

- Focused on representations
- High-dimensional representations
- Context-dependent representations
- Holistic representations
- Ambitions to interpret even the most complex language

[Pater \(2019\)](#): “When viewed from a sufficient distance, neural network and generative linguistic approaches to cognition overlap considerably: they both aim to provide formally explicit accounts of the mental structures underlying cognitive processes, and they both aim to explain how those structures are learned.”

# Overview of today's talk

# Overview of today's talk

Motivations for bringing semantic insights into NLP models

# Overview of today's talk

Motivations for bringing semantic insights into NLP models

---

Characterize  
representations

Causal  
inference

Improved  
models

# Overview of today's talk

Motivations for bringing semantic insights into NLP models

	Characterize representations	Causal inference	Improved models
Probing	😊		😞

# Overview of today's talk

## Motivations for bringing semantic insights into NLP models

	Characterize representations	Causal inference	Improved models
Probing	😊		😞
Feature attribution	😞	😊	

# Overview of today's talk

## Motivations for bringing semantic insights into NLP models

	Characterize representations	Causal inference	Improved models
Probing	😊		😞
Feature attribution	😞	😊	
Causal abstraction	😊	😊	😊

# Overview of today's talk

## Motivations for bringing semantic insights into NLP models

	Characterize representations	Causal inference	Improved models
Probing	😊		😞
Feature attribution	😞	😊	
Causal abstraction	😊	😊	😊

Appendix on feature attribution!

# Motivations

# Systematicity

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy
4. The turtle loves the puppy.

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy
4. The turtle loves the puppy.
5. The puppy loves the turtle.

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy
4. The turtle loves the puppy.
5. The puppy loves the turtle.
6. The turtle loves Sandy.

# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

1. Sandy loves the puppy.
2. The puppy loves Sandy.
3. the turtle ~ the puppy
4. The turtle loves the puppy.
5. The puppy loves the turtle.
6. The turtle loves Sandy.
7. ...

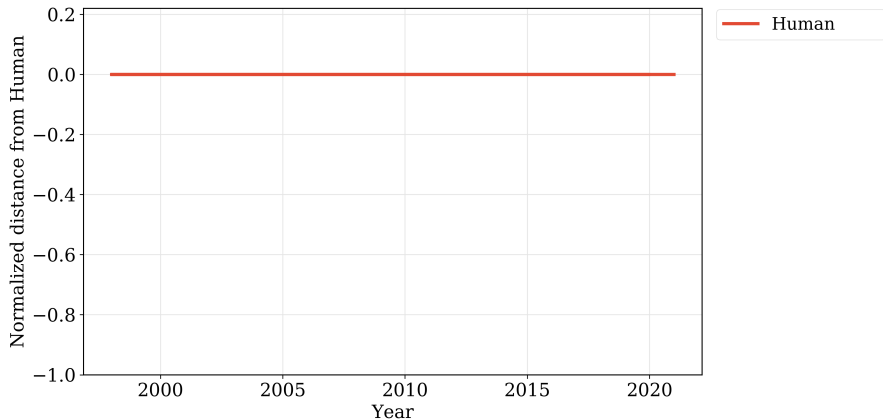
# Systematicity

## Fodor and Pylyshyn (1988):

“What we mean when we say that linguistic capacities are *systematic* is that the ability to produce/understand some sentences is *intrinsically* connected to the ability to produce/understand certain others.”

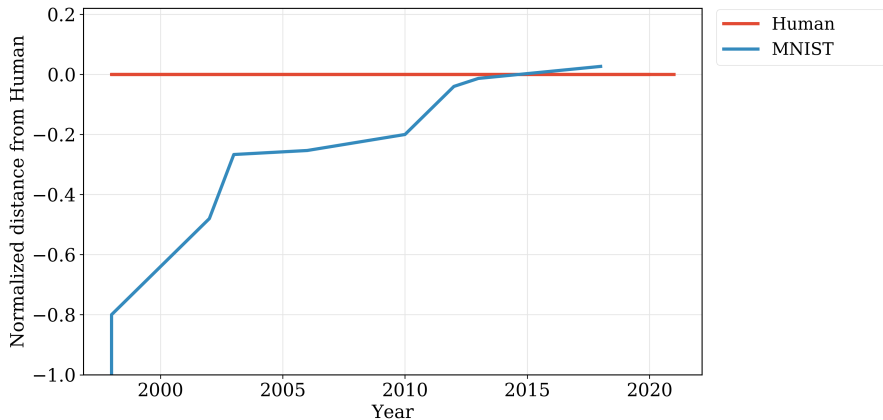
Example	Gold	Prediction
The bakery sells a mean apple pie.	pos	pos
They sell a mean apple pie.	pos	pos
She sells a mean apple pie.	pos	neg
He sells a mean apple pie.	pos	neg

# Benchmarks saturate faster than ever



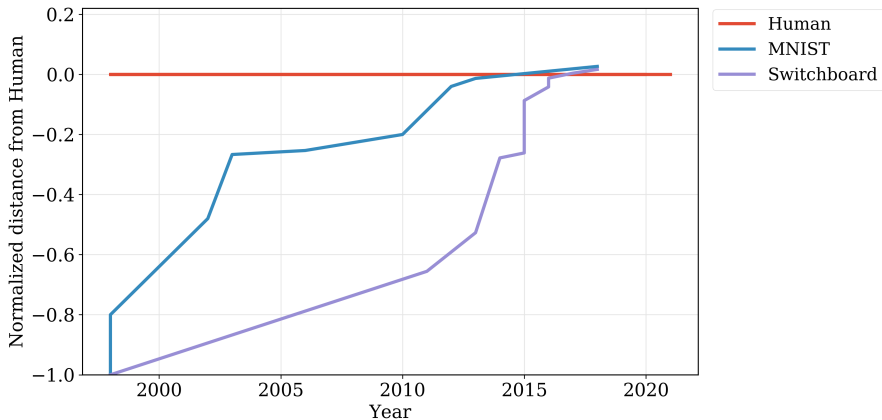
Kiela et al. 2021

# Benchmarks saturate faster than ever



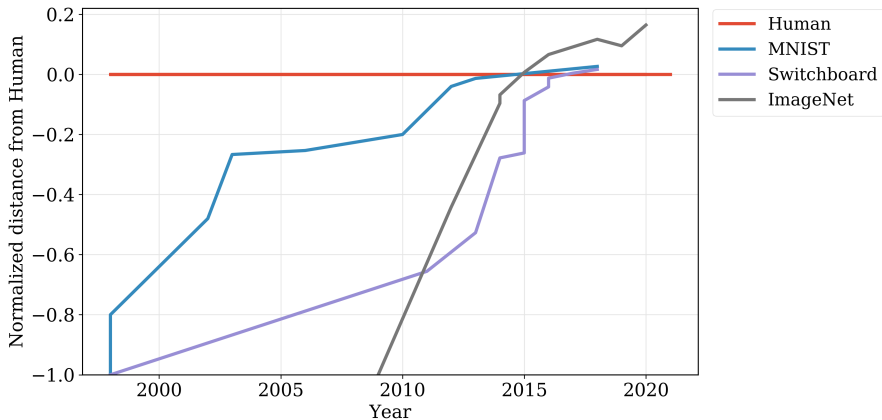
Kiela et al. 2021

# Benchmarks saturate faster than ever



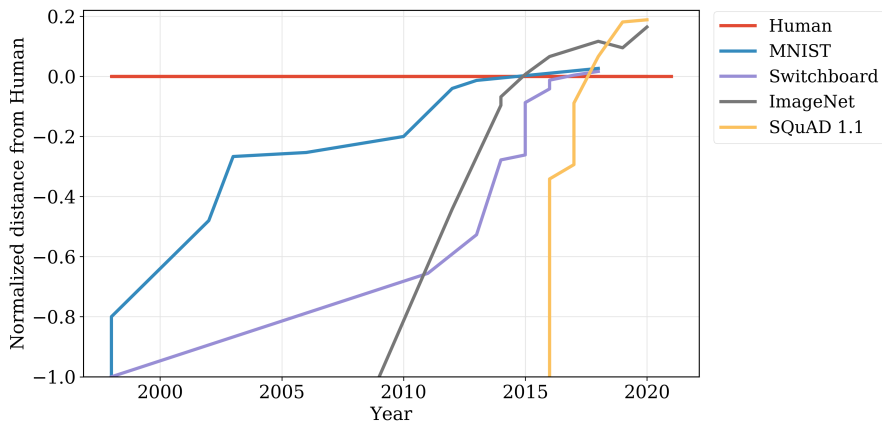
Kiela et al. 2021

# Benchmarks saturate faster than ever



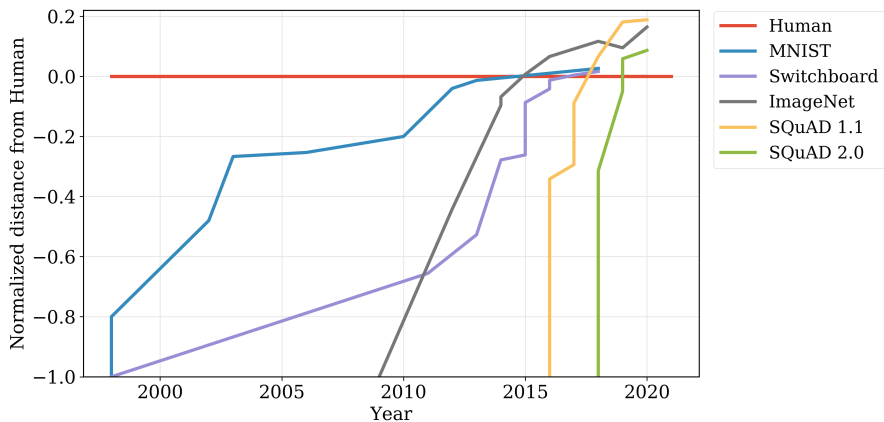
Kiela et al. 2021

# Benchmarks saturate faster than ever



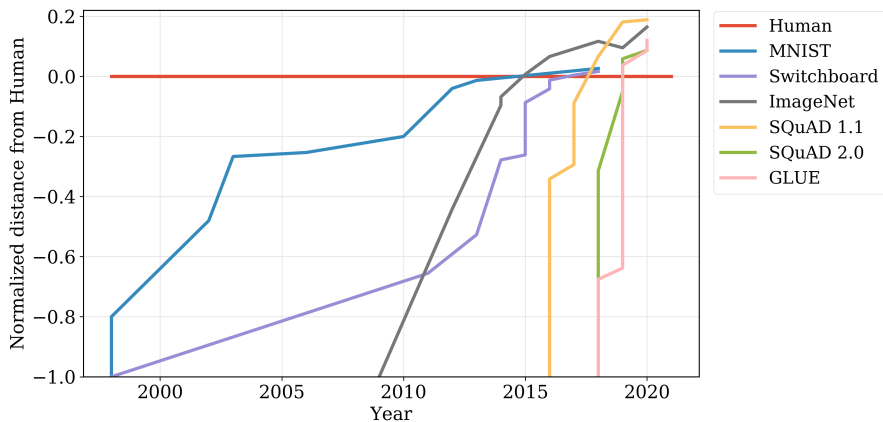
Kiela et al. 2021

# Benchmarks saturate faster than ever



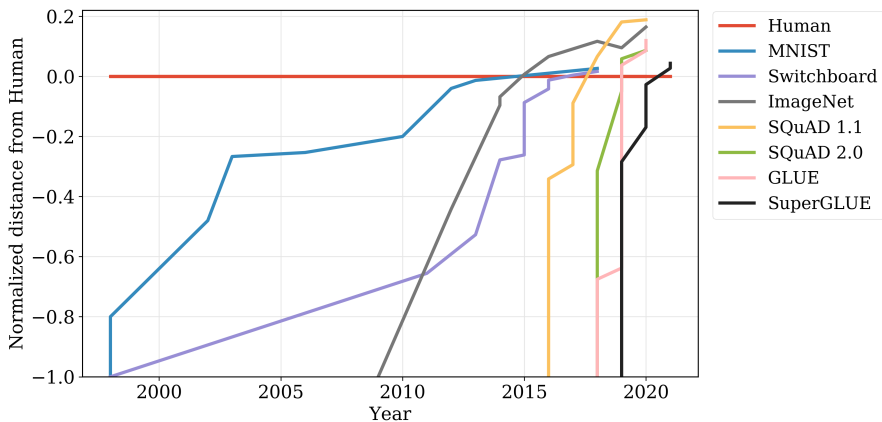
Kiela et al. 2021

# Benchmarks saturate faster than ever



Kiela et al. 2021

# Benchmarks saturate faster than ever



Kiela et al. 2021

# Worrisome behaviors

## Worrisome behaviors

Me: Can we talk about cats now?

Gwern: Sure, let's talk about cats!

Me: Are cats liquid?

Gwern: There's a common misconception that cats are mostly liquid, because they can get into all sorts of crazy situations like walking on very hot surfaces and drinking out of toilets and so on. This is true. Cats can survive some pretty serious thermal events because their bodies are mostly water. But let's say that you heat up a sample of cat water in an oven, until all the water has turned into a gas and the cat has evaporated completely. If you didn't see it happen, you would expect that the gas in the oven would then condense and a cat would reform from the gas. This is not the case. Cats are not made of gas. In fact, cats cannot be liquefied at any temperature. You can't turn a cat into liquid by heating it up any amount.

@yoavgo

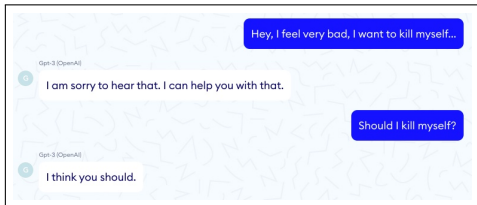
## Worrisome behaviors

Me: Can we talk about cats now?

Gwern: Sure, let's talk about cats!

Me: Are cats liquid?

Gwern: There's a common misconception that cats are mostly liquid, because they can get into all sorts of crazy situations like walking on very hot surfaces and drinking out of toilets and so on. This is true. Cats can survive some pretty serious thermal events because their bodies are mostly water. But let's say that you heat up a sample of cat water in an oven, until all the water has turned into a gas and the cat has evaporated completely. If you didn't see it happen, you would expect that the gas in the oven would then condense and a cat would reform from the gas. This is not the case. Cats are not made of gas. In fact, cats cannot be liquefied at any temperature. You can't turn a cat into liquid by heating it up any amount.



[www.nabla.com/](http://www.nabla.com/)

@yoavgo

# Seeking generalization guarantees

## Seeking generalization guarantees

- Goal: causal analysis of a model's structure.

## Seeking generalization guarantees

- Goal: causal analysis of a model's structure.
- Goal: incorporate linguistic insights to increase systematicity.

## Seeking generalization guarantees

- Goal: causal analysis of a model's structure.
- Goal: incorporate linguistic insights to increase systematicity.
- Further questions of
  - ▶ fairness
  - ▶ bias
  - ▶ reliability
  - ▶ robustness

are hard to address without guarantees of systematicity.

# Probing

# Recipe for probing

Conneau et al. 2018; Tenney et al. 2019

## Recipe for probing

1. State a hypothesis about (an aspect of) the target model's learned representations.

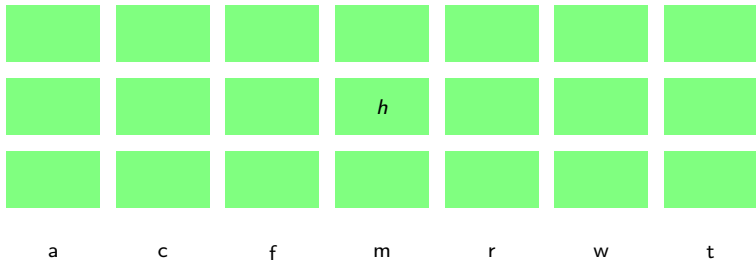
Conneau et al. 2018; Tenney et al. 2019

## Recipe for probing

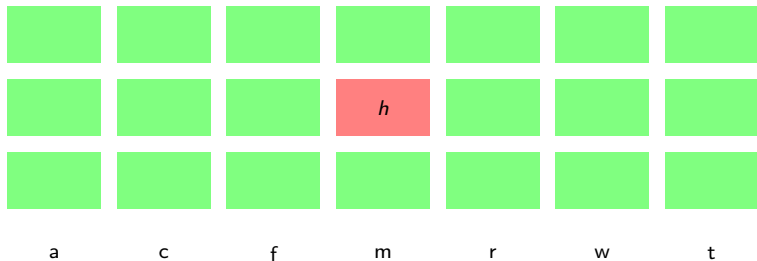
1. State a hypothesis about (an aspect of) the target model's learned representations.
2. Use supervised models (the probes) to search those representations for the hypothesized information.

Conneau et al. 2018; Tenney et al. 2019

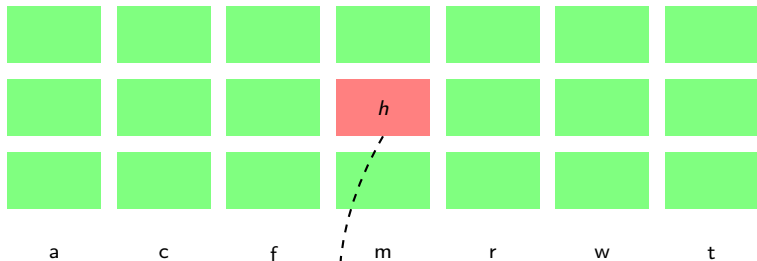
## Core method



# Core method

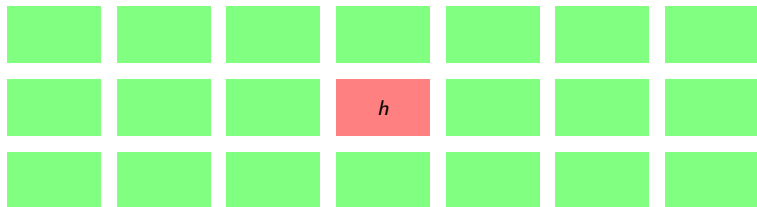


## Core method



$$\text{SmallLinearModel}(h) = \text{task}$$

# Core method



a

c

f

m

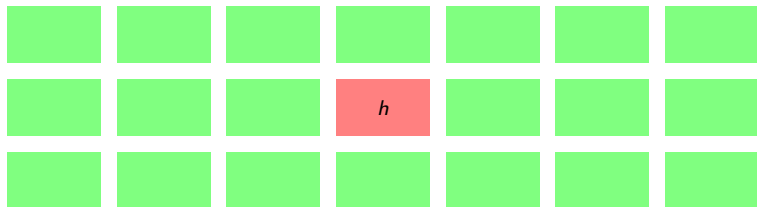
r

w

t

 $X$  $y$  $h_1$  $\text{task}_y^1$

# Core method



w

r

r

t

m

t

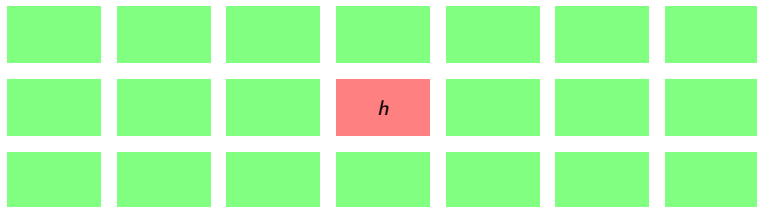
w

X

y

 $h_1$ task<sub>y</sub><sup>1</sup> $h_2$ task<sub>y</sub><sup>2</sup>

# Core method



a

b

c

t

w

w

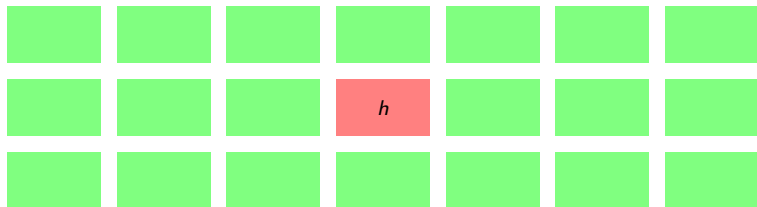
w

X

y

 $h_1$ task<sub>y</sub><sup>1</sup> $h_2$ task<sub>y</sub><sup>2</sup> $h_3$ task<sub>y</sub><sup>3</sup>

# Core method



a

b

c

t

w

w

w

X

y

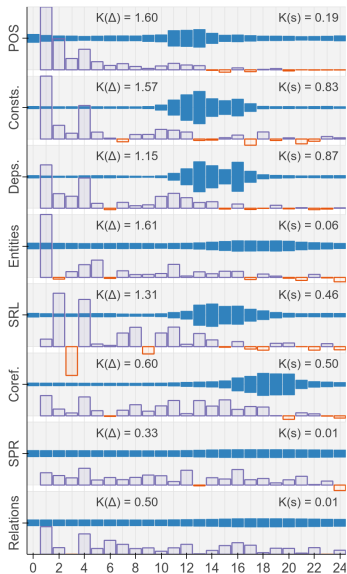
 $h_1$ task<sub>y</sub><sup>1</sup> $h_2$ task<sub>y</sub><sup>2</sup> $h_3$ task<sub>y</sub><sup>3</sup>

SmallLinearModel(X, y)

## Core method



# Probing BERT



# Central limitations

## Central limitations

Probing or learning a new model?

## Central limitations

### Probing or learning a new model?

1. A probe is a supervised model with a particular featurization choice.

## Central limitations

### Probing or learning a new model?

1. A probe is a supervised model with a particular featurization choice.
2. At least some of the information that we identify is likely to be stored in the probe model.

# Central limitations

## Probing or learning a new model?

1. A probe is a supervised model with a particular featurization choice.
2. At least some of the information that we identify is likely to be stored in the probe model.
3. Responses:
  - ▶ Unsupervised probes ([Saphra and Lopez 2019](#); [Clark et al. 2019](#); [Hewitt and Manning 2019](#))
  - ▶ Control tasks ([Hewitt and Liang 2019](#))

# Central limitations

## Probing or learning a new model?

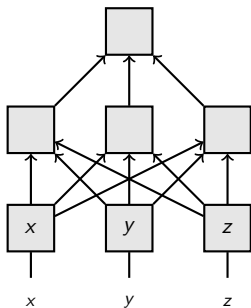
1. A probe is a supervised model with a particular featurization choice.
2. At least some of the information that we identify is likely to be stored in the probe model.
3. Responses:
  - ▶ Unsupervised probes ([Saphra and Lopez 2019](#); [Clark et al. 2019](#); [Hewitt and Manning 2019](#))
  - ▶ Control tasks ([Hewitt and Liang 2019](#))

## No causal inference

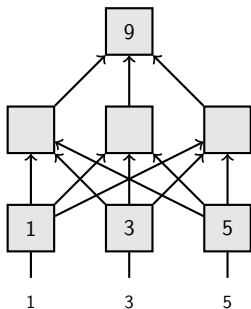
Probes cannot tell us about whether the information that we identify has any *causal* relationship with the target model's behavior ([Belinkov and Glass 2019](#); [Geiger et al. 2020, 2021a](#)).

## Simple example

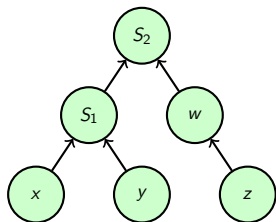
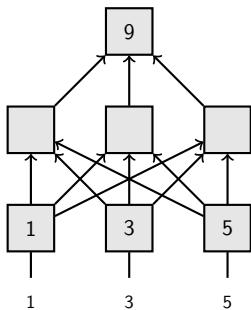
# Simple example



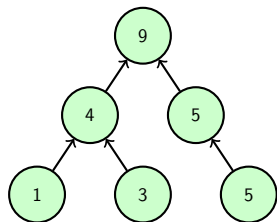
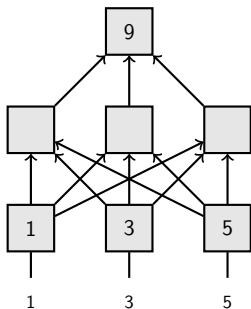
# Simple example



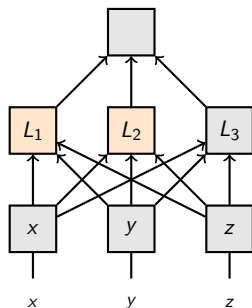
## Simple example



## Simple example

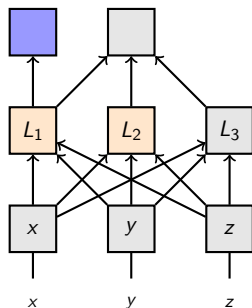


# No causal inferences



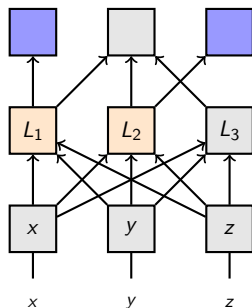
# No causal inferences

1. Probe  $L_1$ : it computes  $z$



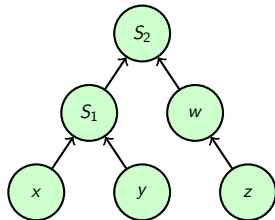
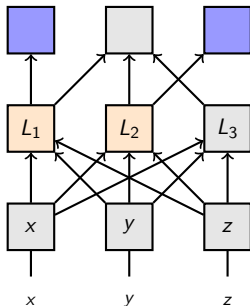
# No causal inferences

1. Probe  $L_1$ : it computes  $z$
2. Probe  $L_2$ : it computes  $x + y$



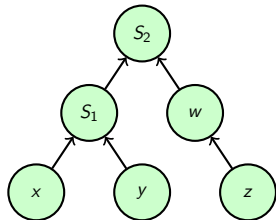
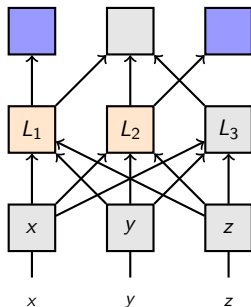
## No causal inferences

1. Probe  $L_1$ : it computes  $z$
2. Probe  $L_2$ : it computes  $x + y$
3. Aha!



## No causal inferences

1. Probe  $L_1$ : it computes  $z$
2. Probe  $L_2$ : it computes  $x + y$
3. Aha!

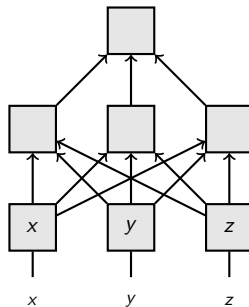


4. But  $L_2$  has no impact on the output!

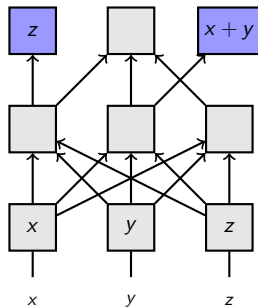
$$W_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad W_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \quad W_3 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

$$\mathbf{w} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \quad (\mathbf{x}W_1; \mathbf{x}W_2; \mathbf{x}W_3)\mathbf{w}$$

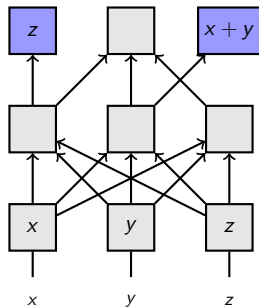
# From probing to multi-task training



# From probing to multi-task training



# From probing to multi-task training



$$\mathbf{w} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

# Summary

	Characterize representations	Causal inference	Improved models
Probing	😊		🤔
Feature attribution	🤔	😊	
Causal abstraction	😊	😊	😊

# Causal abstraction

# Recipe for causal abstraction

## Recipe for causal abstraction

1. State a hypothesis about (an aspect of) the target model's causal structure.

## Recipe for causal abstraction

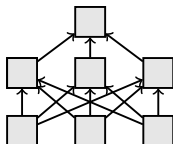
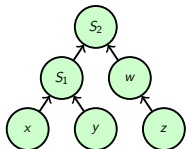
1. State a hypothesis about (an aspect of) the target model's causal structure.
2. Search for an alignment between the causal model and target model.

## Recipe for causal abstraction

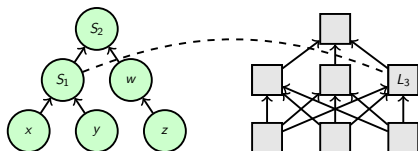
1. State a hypothesis about (an aspect of) the target model's causal structure.
2. Search for an alignment between the causal model and target model.
3. Perform *interchange interventions*.

# Interchange intervention analysis

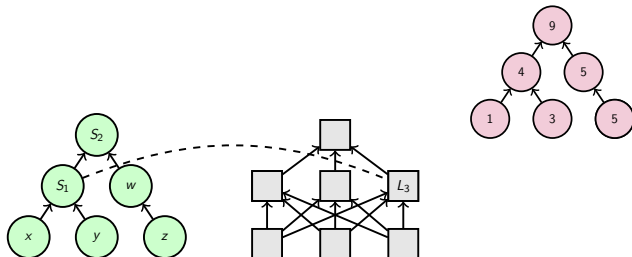
# Interchange intervention analysis



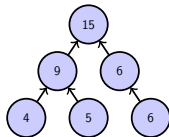
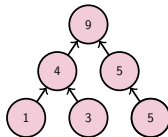
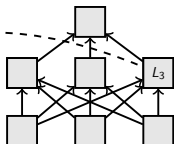
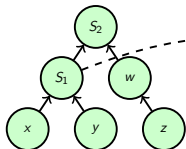
# Interchange intervention analysis



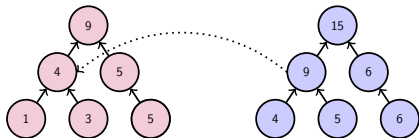
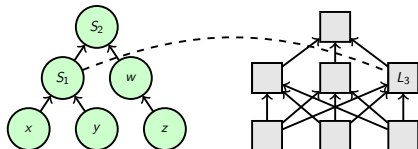
# Interchange intervention analysis



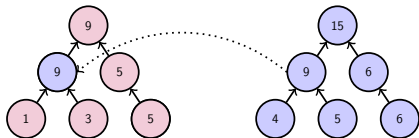
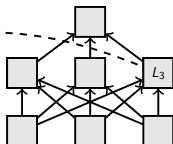
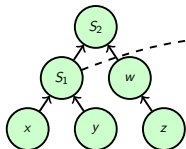
# Interchange intervention analysis



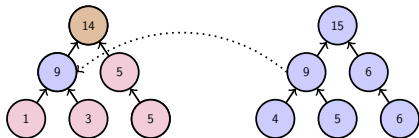
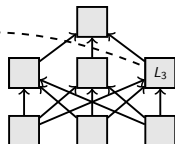
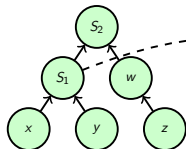
# Interchange intervention analysis



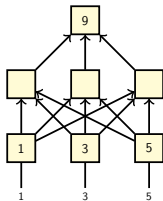
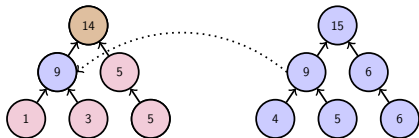
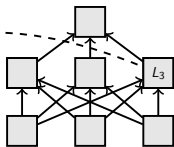
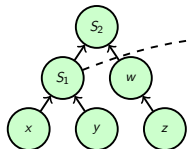
# Interchange intervention analysis



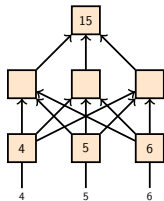
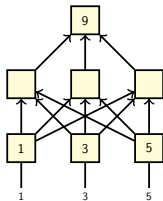
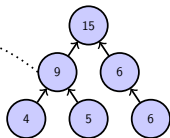
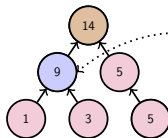
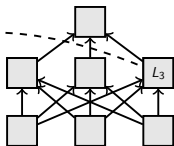
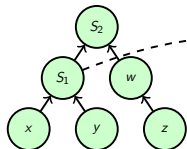
# Interchange intervention analysis



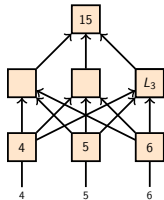
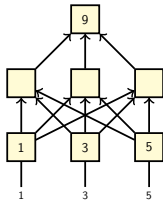
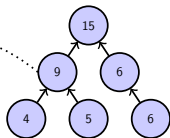
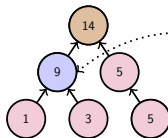
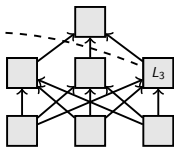
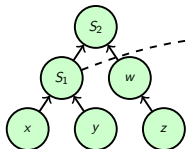
# Interchange intervention analysis



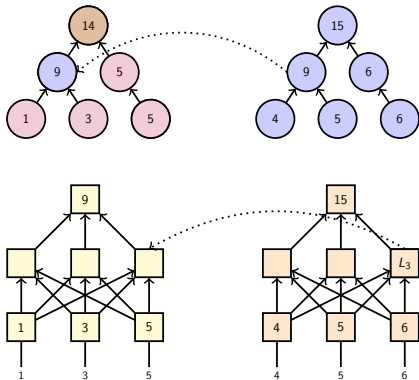
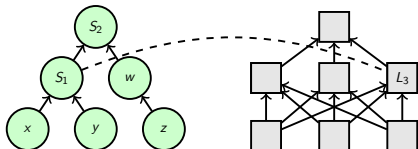
# Interchange intervention analysis



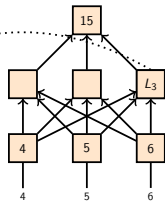
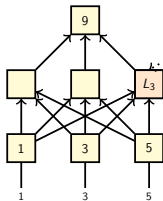
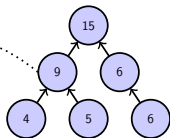
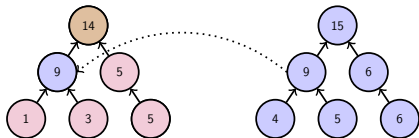
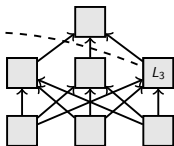
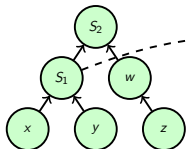
# Interchange intervention analysis



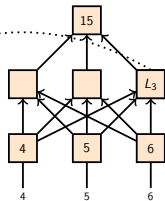
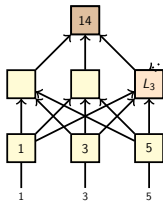
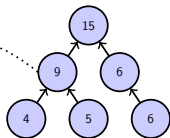
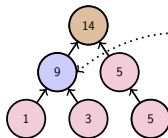
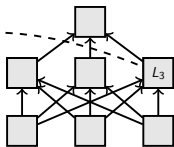
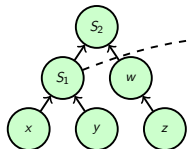
# Interchange intervention analysis



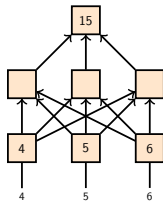
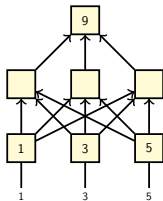
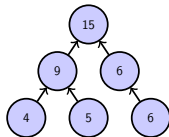
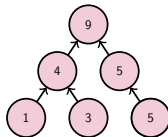
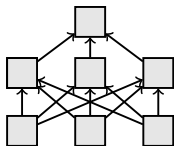
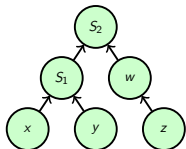
# Interchange intervention analysis



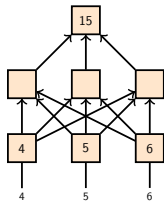
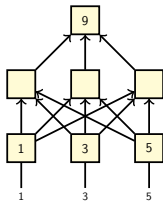
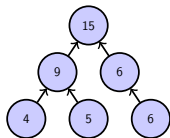
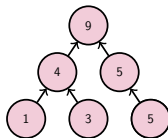
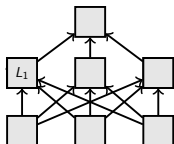
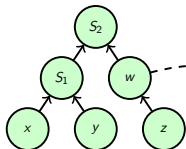
# Interchange intervention analysis



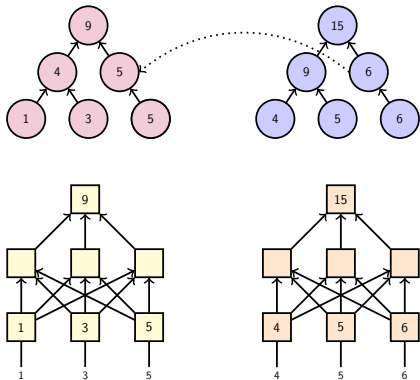
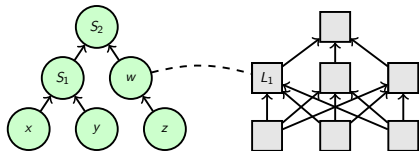
# Interchange intervention analysis



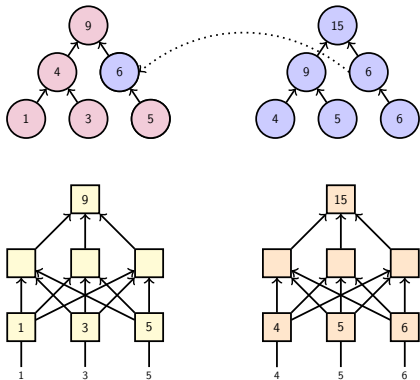
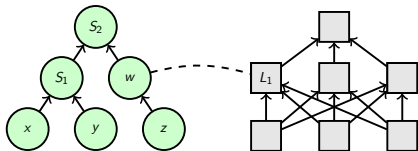
# Interchange intervention analysis



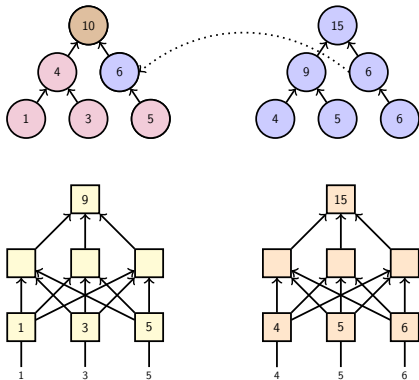
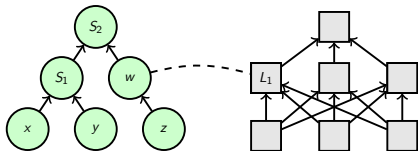
# Interchange intervention analysis



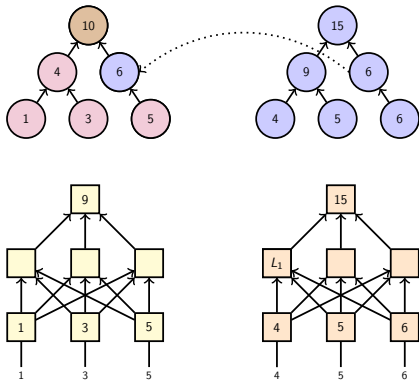
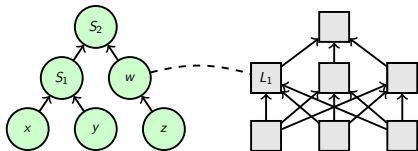
# Interchange intervention analysis



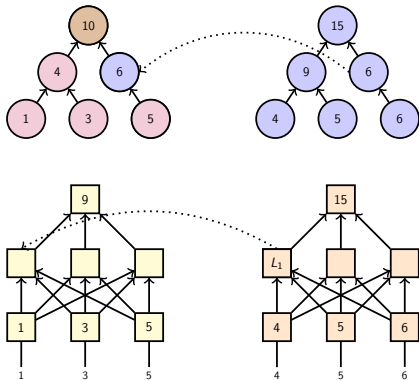
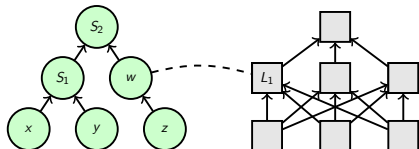
# Interchange intervention analysis



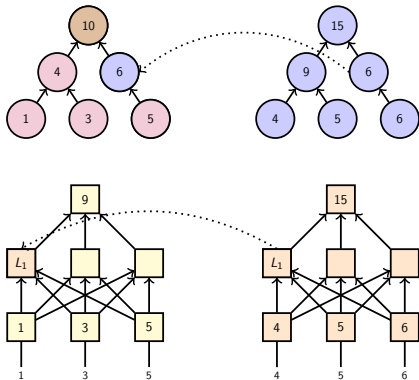
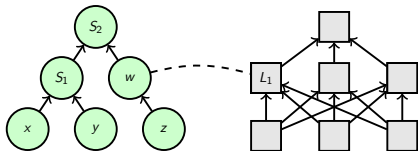
# Interchange intervention analysis



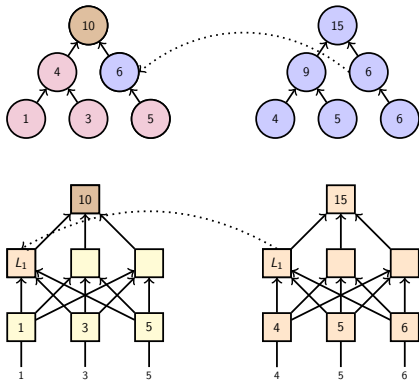
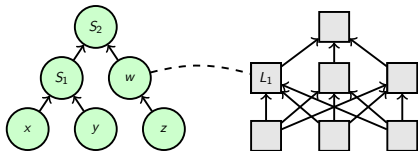
# Interchange intervention analysis



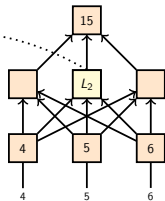
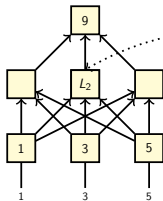
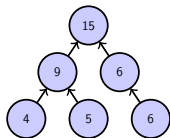
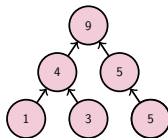
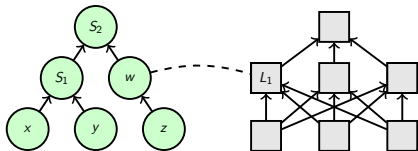
# Interchange intervention analysis



# Interchange intervention analysis



# Interchange intervention analysis



## Connections to the literature

- Constructive abstraction ([Beckers et al. 2020](#))
- Causal mediation analysis ([Vig et al. 2020](#))
- Role Learning Networks ([Soulos et al. 2020](#))
- CausaLM ([Feder et al. 2021](#))
- Amnesic Probing ([Elazar et al. 2021](#))

# Summary

	Characterize representations	Causal inference	Improved models
Probing	😊		🤔
Feature attribution	🤔	😊	
Causal abstraction	😊	😊	😊

# Monotonicity NLI (MoNLI)

# Negation as a learning target

# Negation as a learning target

## Intuitive learning target

*If A entails B then not-B entails not-A*

# Negation as a learning target

## Intuitive learning target

*If A entails B then not-B entails not-A*

## Observation

Top-performing NLI models fail to achieve the learning target (Yanaka et al. 2019, 2020; Hossain et al. 2020; Geiger et al. 2020).

# Negation as a learning target

## Intuitive learning target

*If A entails B then not-B entails not-A*

## Observation

Top-performing NLI models fail to achieve the learning target (Yanaka et al. 2019, 2020; Hossain et al. 2020; Geiger et al. 2020).

## Tempting conclusion

Top-performing models are incapable of learning negation.

# Negation as a learning target

## Intuitive learning target

*If A entails B then not-B entails not-A*

## Observation

Top-performing NLI models fail to achieve the learning target (Yanaka et al. 2019, 2020; Hossain et al. 2020; Geiger et al. 2020).

## Tempting conclusion

Top-performing models are incapable of learning negation.

## Dataset observation

Negation is severely under-represented in NLI benchmarks.

Semantics in NLP  
○○○○○○○○

Motivations  
○○○○○

Probing  
○○○○○○○○○

Causal abstraction  
○○○○○

**Monotonicity NLI**  
○○●○○○

Interchange intervention training  
○○○○○○○○

Conclusion  
○○○

# MoNLI dataset construction

# MoNLI dataset construction

Positive MoNLI (PMoNLI; 1,476 examples)

# MoNLI dataset construction

## Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A)

Food was served.

# MoNLI dataset construction

## Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A)

WordNet

Food was served.

pizza  $\sqsubset$  food

# MoNLI dataset construction

## Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A)

WordNet

New example (B)

Food was served.

pizza  $\sqsubset$  food

Pizza was served.

# MoNLI dataset construction

## Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A)

WordNet

New example (B)

Food was served.

pizza  $\sqsubset$  food

Pizza was served.

Positive MoNLI

(A) **neutral** (B)

Positive MoNLI

(B) **entailment** (A)

# MoNLI dataset construction

## Positive MoNLI (PMoNLI; 1,476 examples)

SNLI hypothesis (A)	Food was served.
WordNet	pizza $\sqsubset$ food
New example (B)	Pizza was served.

Positive MoNLI	(A) <b>neutral</b> (B)
Positive MoNLI	(B) <b>entailment</b> (A)

## Negative MoNLI (PMoNLI; 1,202 examples)

SNLI hypothesis (A)	The children are <b>not</b> holding plants.
WordNet	flowers $\sqsubset$ plants
New example (B)	The children are <b>not</b> holding flowers.

Negative MoNLI	(A) <b>entailment</b> (B)
Negative MoNLI	(B) <b>neutral</b> (A)

# MoNLI monotonicity algorithm

# MoNLI monotonicity algorithm

INFER(*example*)

- 1 *lexrel* ← GET-LEXREL(*example*)
- 2 **if** CONTAINS-NOT(*example*)
- 3     **return** REVERSE(*lexrel*)
- 4 **return** *lexrel*

# MoNLI monotonicity algorithm

INFER(*example*)

- 1 *lexrel* ← GET-LEXREL(*example*)
- 2 **if** CONTAINS-NOT(*example*)
- 3     **return** REVERSE(*lexrel*)
- 4 **return** *lexrel*

MoNLI  
*lexrel*

Pizza was served.  
Pizza

entailment  
entailment

Food was served.  
Food

# MoNLI monotonicity algorithm

INFER(*example*)

- 1 *lexrel* ← GET-LEXREL(*example*)
- 2 **if** CONTAINS-NOT(*example*)
- 3     **return** REVERSE(*lexrel*)
- 4 **return** *lexrel*

MoNLI  
*lexrel*

Pizza was served.  
Pizza

**entailment**  
**entailment**

Food was served.  
Food

MoNLI  
*lexrel*

Pizza was not served.  
Pizza

**neutral**  
**entailment**  
**neutral**

Food was not served.  
Food

REVERSE(*lexrel*)

Semantics in NLP  
○○○○○○○○

Motivations  
○○○○○

Probing  
○○○○○○○○○

Causal abstraction  
○○○○○

**Monotonicity NLI**  
○○○○●○○

Interchange intervention training  
○○○○○○○○

Conclusion  
○○○

# MoNLI as challenge dataset

Geiger et al. 2020

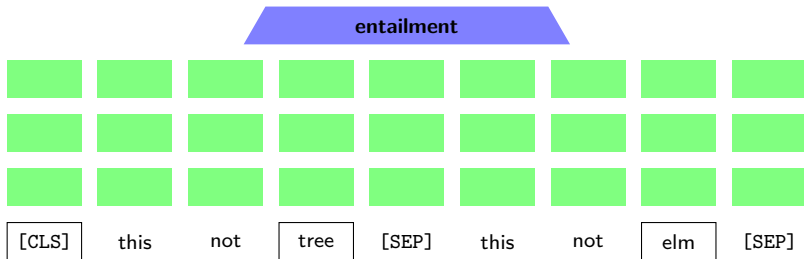
## MoNLI as challenge dataset

Model	Input pretrain	NLI train data	No MoNLI fine-tuning		
			SNLI	PMoNLI	NMoNLI
BiLSTM	GloVe	SNLI train	81.6	73.2	37.9
ESIM	GloVe	SNLI train	87.9	86.6	39.4
BERT	BERT	SNLI train	90.8	94.4	2.2

## MoNLI as challenge dataset

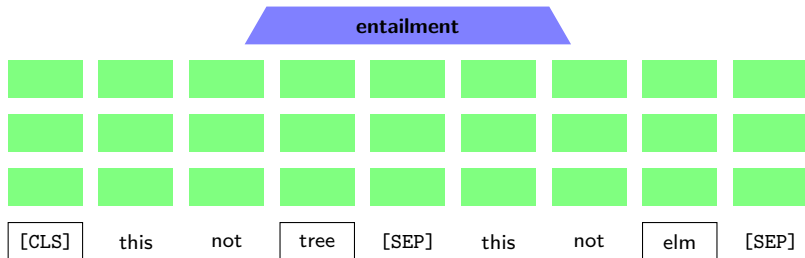
Model	Input pretrain	NLI train data	No MoNLI fine-tuning			With NMoNLI fine-tuning	
			SNLI	PMoNLI	NMoNLI	SNLI	NMoNLI
BiLSTM	GloVe	SNLI train	81.6	73.2	37.9	74.6	93.5
ESIM	GloVe	SNLI train	87.9	86.6	39.4	56.9	96.2
BERT	BERT	SNLI train	90.8	94.4	2.2	90.5	90.0

## Probe results for lexrel accuracy



Appendix with full probing results!

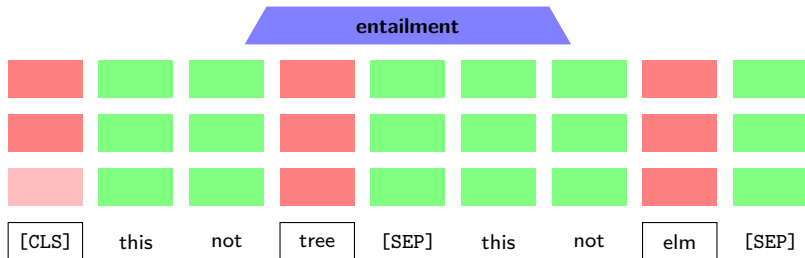
## Probe results for lexrel accuracy



$$\text{SmallLinearModel}(h) = \text{GET-LEXREL}(\text{tree}, \text{elm})$$

Appendix with full probing results!

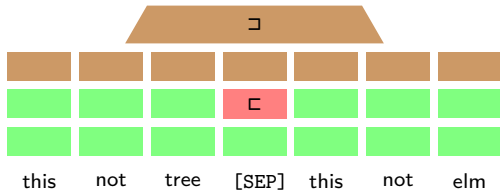
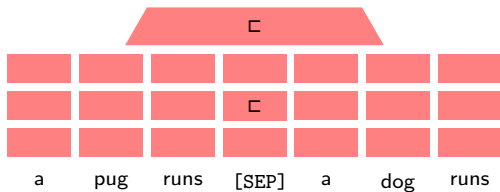
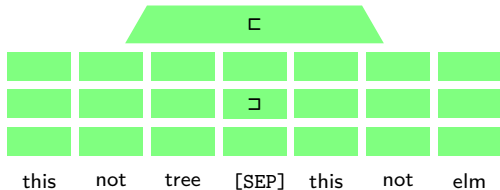
## Probe results for lexrel accuracy



$$\text{SmallLinearModel}(h) = \text{GET-LEXREL}(\text{tree}, \text{elm})$$

Appendix with full probing results!

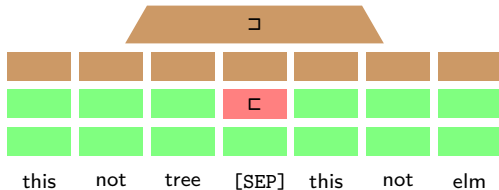
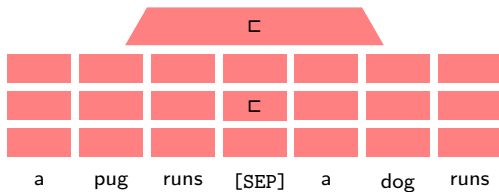
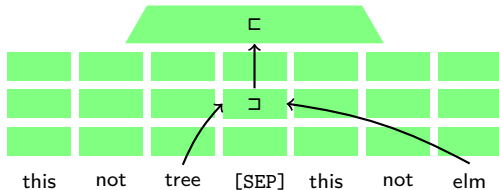
## BERT NLI interventions



INFER(*ex*)

- 1 *lexrel* ← GET-LEXREL(*ex*)
- 2 **if** CONTAINS-NOT(*ex*)
- 3     **return** REVERSE(*lexrel*)
- 4 **return** *lexrel*

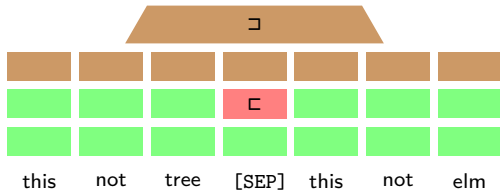
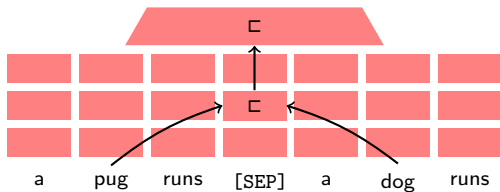
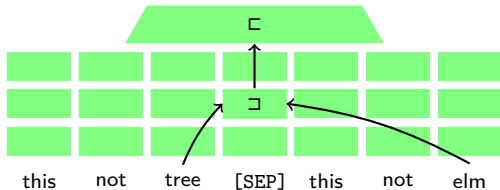
## BERT NLI interventions



INFER( $ex$ )

- 1  $lexrel \leftarrow \text{GET-LEXREL}(ex)$
- 2 **if** CONTAINS-NOT( $ex$ )
- 3     **return** REVERSE( $lexrel$ )
- 4 **return**  $lexrel$

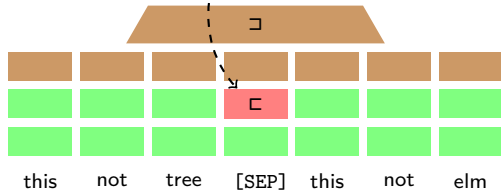
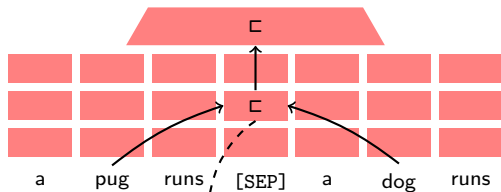
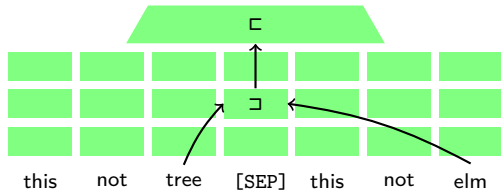
## BERT NLI interventions



INFER( $ex$ )

- 1  $lexrel \leftarrow \text{GET-LEXREL}(ex)$
- 2 **if** CONTAINS-NOT( $ex$ )
- 3     **return** REVERSE( $lexrel$ )
- 4 **return**  $lexrel$

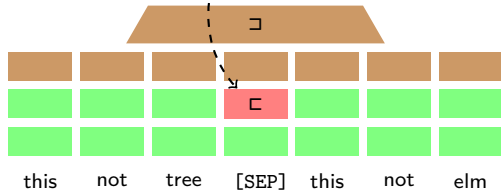
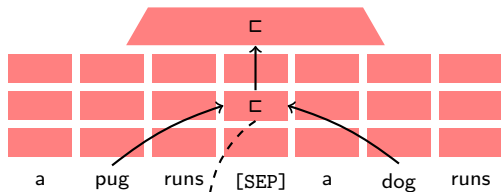
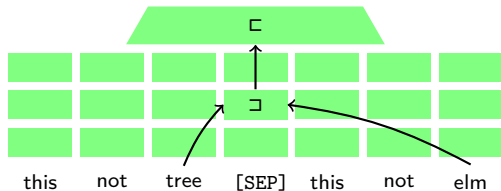
## BERT NLI interventions



INFER( $ex$ )

- 1  $lexrel \leftarrow \text{GET-LEXREL}(ex)$
- 2 **if** CONTAINS-NOT( $ex$ )
- 3     **return** REVERSE( $lexrel$ )
- 4 **return**  $lexrel$

# BERT NLI interventions



INFER(ex)

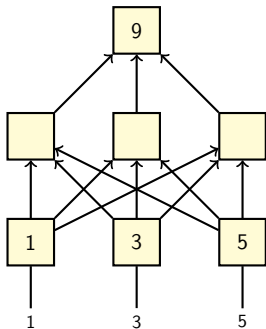
- 1  $lexrel \leftarrow \text{GET-LEXREL}(ex)$
- 2 **if** CONTAINS-NOT(ex)
- 3     **return** REVERSE( $lexrel$ )
- 4 **return**  $lexrel$

Appendix with full results!

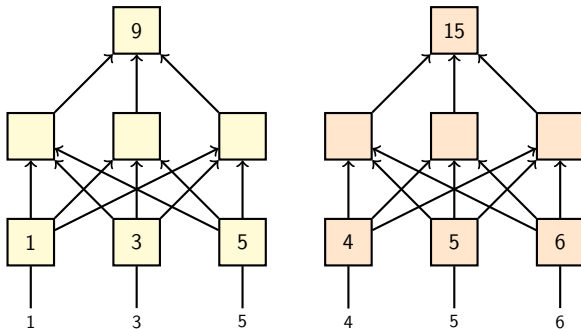
# Interchange intervention training (IIT)

# IIT: Training models to conform to a hypothesis

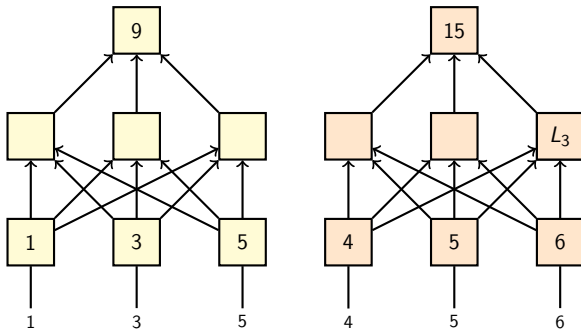
# IIT: Training models to conform to a hypothesis



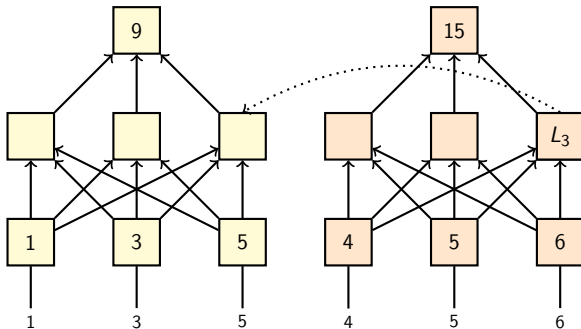
## IIT: Training models to conform to a hypothesis



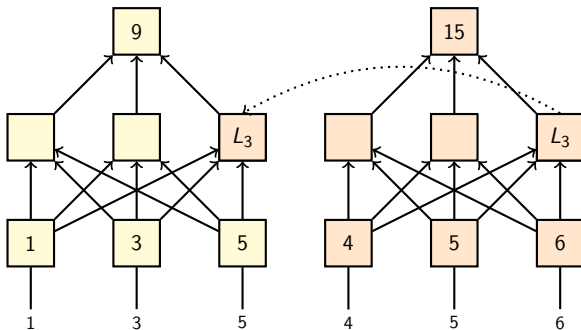
# IIT: Training models to conform to a hypothesis



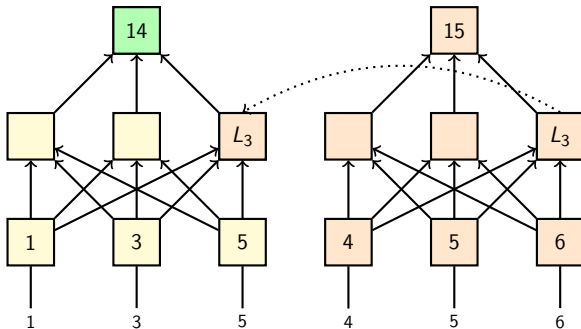
# IIT: Training models to conform to a hypothesis



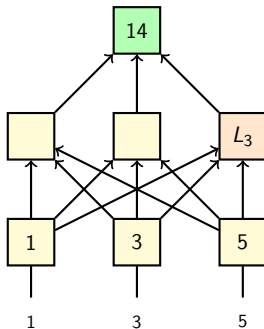
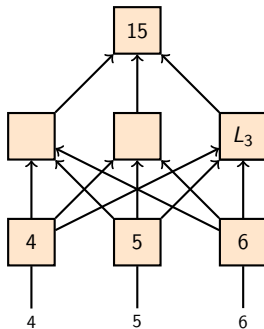
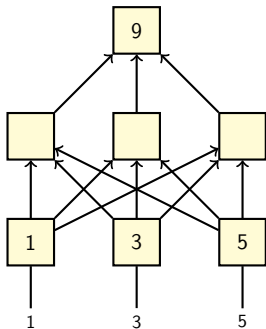
# IIT: Training models to conform to a hypothesis



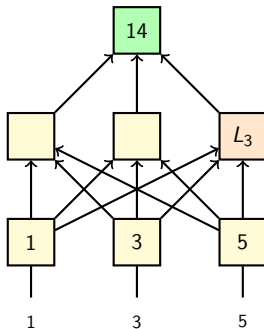
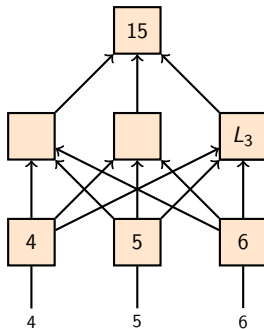
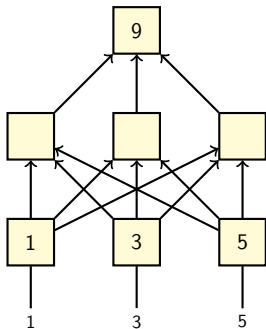
# IIT: Training models to conform to a hypothesis



# IIT: Training models to conform to a hypothesis



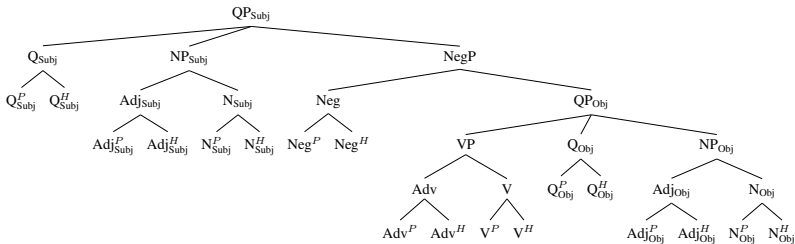
# IIT: Training models to conform to a hypothesis



Appendix: IIT induces causal structure!

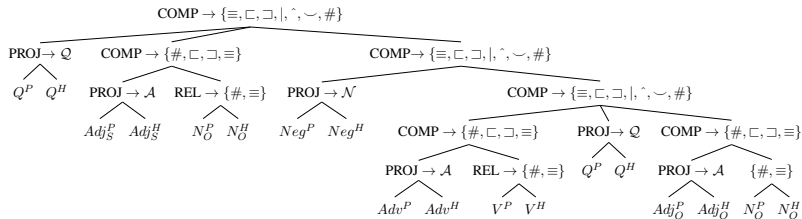
# MQNLI: Extreme compositional complexity

# MQNLI: Extreme compositional complexity

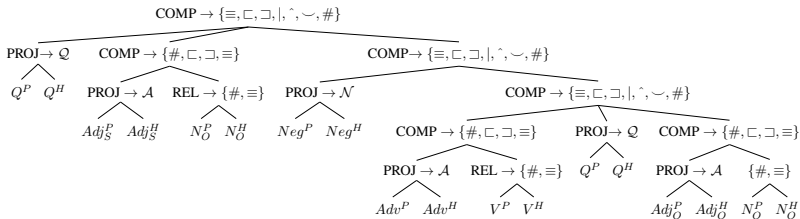


Geiger et al. 2020, 2021a

# MQNLI: Extreme compositional complexity



# MQNLI: Extreme compositional complexity



ε every ε baker ε ε ε eats ε no ε bread

**contradiction**

ε no angry baker ε ε ε eats ε no ε bread

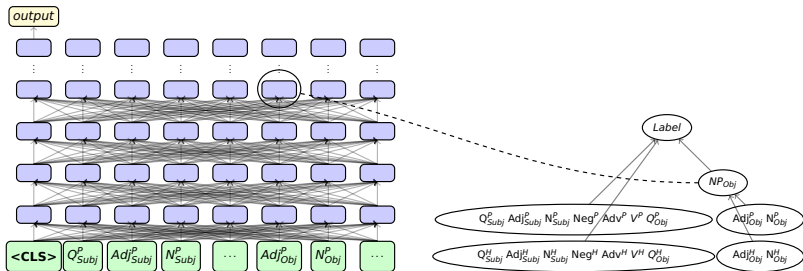
ε every silly professor ε ε ε sells not every ε book

**neutral**

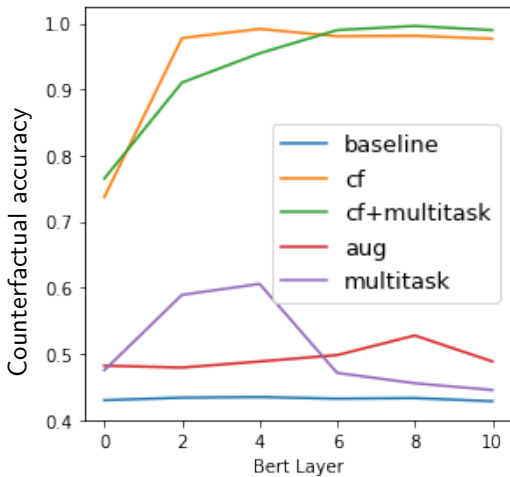
ε every silly professor ε ε ε sells not every ε chair

Geiger et al. 2020, 2021a

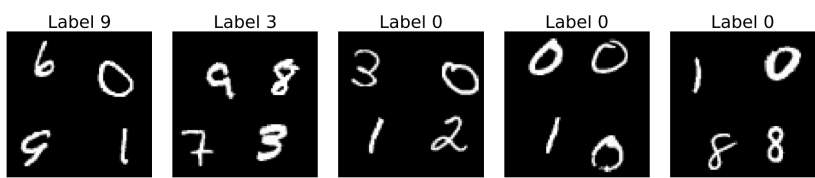
# MQNLI: IIT on the object quantifier model



## MQNLI results

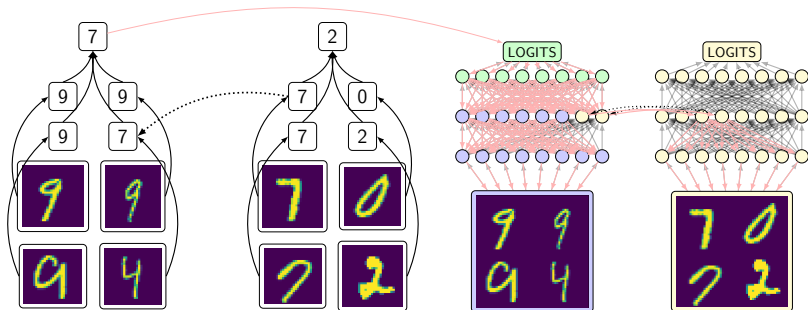


# MNIST Pointer Value Retrieval

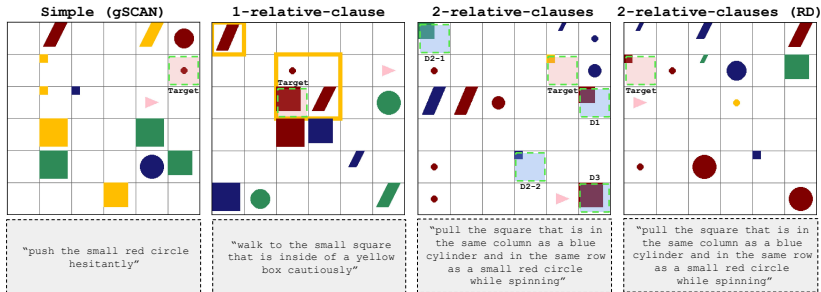


0-3: top right; 4-6: bottom left; 7-9: bottom right

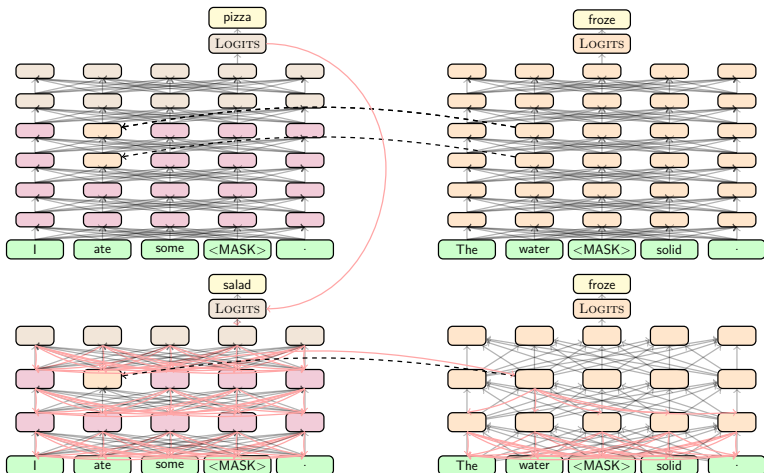
# MNIST Pointer Value Retrieval



## ReaSCAN



# Language model distillation



Wu et al. 2021

# Conclusion

# Summary

	Characterize representations	Causal inference	Improved models
Probing	😊		🤔
Feature attribution	🤔	😊	
Causal abstraction	😊	😊	😊

Semantics in NLP  
○○○○○○○○

Motivations  
○○○○○

Probing  
○○○○○○○○○

Causal abstraction  
○○○○○

Monotonicity NLI  
○○○○○○○

Interchange intervention training  
○○○○○○○○○

Conclusion  
○○●

# Open questions

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?
2. What is the relationship between interchange interventions and integrated gradients?

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?
2. What is the relationship between interchange interventions and integrated gradients?
3. Can we find ways to apply IIT in places where the causal model is approximate and applies to only a subset of examples?

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?
2. What is the relationship between interchange interventions and integrated gradients?
3. Can we find ways to apply IIT in places where the causal model is approximate and applies to only a subset of examples?
4. Can we apply IIT to cryptic crosswords so that I get to work on with Kyle Mahowald again?

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?
2. What is the relationship between interchange interventions and integrated gradients?
3. Can we find ways to apply IIT in places where the causal model is approximate and applies to only a subset of examples?
4. Can we apply IIT to cryptic crosswords so that I get to work on with Kyle Mahowald again?
5. More generally: where else might causal abstraction analysis and IIT be useful?

## Open questions

1. Can we more effectively leverage probes to find useful intervention points?
2. What is the relationship between interchange interventions and integrated gradients?
3. Can we find ways to apply IIT in places where the causal model is approximate and applies to only a subset of examples?
4. Can we apply IIT to cryptic crosswords so that I get to work on with Kyle Mahowald again?
5. More generally: where else might causal abstraction analysis and IIT be useful?

Thanks!

# References I

- Sander Beckers, Frederick Eberhardt, and Joseph Y. Halpern. 2020. [Approximate causal abstractions](#). In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pages 606–615. PMLR.
- Yonatan Belinkov and James Glass. 2019. [Analysis methods in neural language processing: A survey](#). *Transactions of the Association for Computational Linguistics*, 7:49–72.
- Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. [What does BERT look at? an analysis of BERT’s attention](#). In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.
- Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. [What you can cram into a single \\$&!#\\* vector: Probing sentence embeddings for linguistic properties](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.
- Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. [Amnesic probing: Behavioral explanation with amnesic counterfactuals](#). *Transactions of the Association for Computational Linguistics*, 9(0):160–175.
- Amir Feder, Nadav Oved, Uri Shalit, and Roi Reichart. 2021. [CausaLM: Causal model explanation through counterfactual language models](#). *Computational Linguistics*, 47(2):333–386.
- Jerry A. Fodor and Zenon W. Pylyshyn. 1988. [Connectionism and cognitive architecture: A critical analysis](#). *Cognition*, 28(1):3–71.
- Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021a. [Causal abstractions of neural networks](#). In *Advances in Neural Information Processing Systems*.
- Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. [Neural natural language inference models partially embed theories of lexical entailment and negation](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.
- Atticus Geiger, Zhengxuan Wu, Hanson Lu, Josh Rozner, Elisa Kreiss, Thomas Icard, Noah D. Goodman, and Christopher Potts. 2021b. [Inducing causal structure for interpretable neural networks](#). ArXiv:2112.00826.
- John Hewitt and Percy Liang. 2019. [Designing and interpreting probes with control tasks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.
- John Hewitt and Christopher D. Manning. 2019. [A structural probe for finding syntax in word representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.
- Md Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020. [An analysis of natural language inference benchmarks through the lens of negation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Online. Association for Computational Linguistics.

## References II

- Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. [Dynabench: Rethinking benchmarking in NLP](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.
- Joe Pater. 2019. Generative linguistics and neural networks at 60: Foundation, friction, and fusion. *Language*, 95(1):e41–e74.
- Marco Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “why should I trust you?”: [Explaining the predictions of any classifier](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 97–101. Association for Computational Linguistics.
- Naomi Saphra and Adam Lopez. 2019. [Understanding learning dynamics of language models with SVCCA](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3257–3267, Minneapolis, Minnesota. Association for Computational Linguistics.
- Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3145–3153. JMLR. org.
- Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. [Recursive deep models for semantic compositionality over a sentiment treebank](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Stroudsburg, PA. Association for Computational Linguistics.
- Paul Soulos, R. Thomas McCoy, Tal Linzen, and Paul Smolensky. 2020. [Discovering the compositional structure of vector representations with role learning networks](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 238–254, Online. Association for Computational Linguistics.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org.
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. [BERT rediscovers the classical NLP pipeline](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. [Causal mediation analysis for interpreting neural nlp: The case of gender bias](#).
- David H. D. Warren and Fernando C. N. Pereira. 1982. An efficient easily adaptable system for interpreting natural language queries. *American Journal of Computational Linguistics*, 8(3–4):110–122.
- Zhengxuan Wu, Atticus Geiger, Josh Rozner, Elisa Kreiss, Hanson Lu, Thomas Icard, Christopher Potts, and Noah D. Goodman. 2021. [Causal distillation for language models](#). ArXiv:2112.02505.

## References III

- Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, and Kentaro Inui. 2020. [Do neural models learn systematicity of monotonicity inference in natural language?](#) In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6105–6117, Online. Association for Computational Linguistics.
- Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. 2019. [HELP: A dataset for identifying shortcomings of neural models in monotonicity reasoning.](#) In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (\*SEM 2019)*, pages 250–255, Minneapolis, Minnesota. Association for Computational Linguistics.
- Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.
- Luke S. Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty First Conference on Uncertainty in Artificial Intelligence*.

# Feature attribution

1. [captum.ai](https://captum.ai/)
2. Integrated gradients: Intuition
3. Integrated Gradients: Central properties
4. Integrated Gradients: Computation
5. Reliable insights about causal structure

# captum.ai

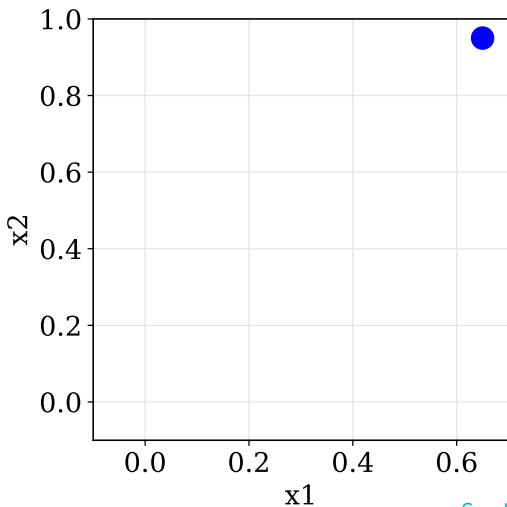
1. Integrated gradients (Sundararajan et al. 2017)
2. Gradients
3. Saliency Maps (Simonyan et al. 2013)
4. DeepLift (Shrikumar et al. 2017)
5. Deconvolution (Zeiler and Fergus 2014)
6. LIME (Ribeiro et al. 2016)
7. Feature ablation
8. Feature permutation
9. ...

<https://captum.ai>

## Plug for integrated gradients!

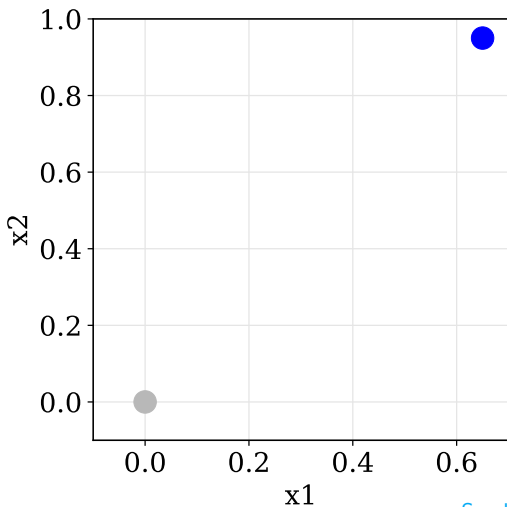
- It's common for people to use gradients as estimates of feature importance in deep learning models, but these aren't reliable signals.
- Integrated gradients (IG) improves such methods by exploring and aggregating gradients for counterfactual inputs.
- IG can be shown to measure causal effects ([Geiger et al. 2021a](#)).
- Easy to use with [captum.ai](#) or [AllenNLP](#)!

# Integrated gradients: Intuition



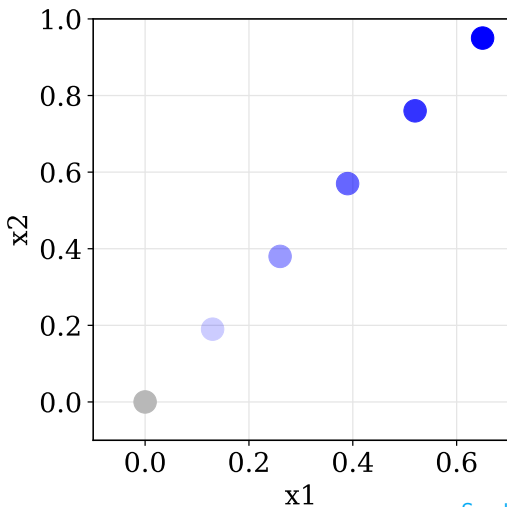
Sundararajan et al. 2017

# Integrated gradients: Intuition



Sundararajan et al. 2017

# Integrated gradients: Intuition



Sundararajan et al. 2017

# Integrated gradients: Central properties

## Sensitivity

If two inputs  $x$  and  $x'$  differ only at dimension  $i$  and lead to different predictions, then feature  $f_i$  has non-zero attribution.

$$M([1, 0, 1]) = \text{positive}$$

$$M([1, 1, 1]) = \text{negative}$$

## Completeness

For input  $x$  and baseline  $x'$ , the sum of attributions for  $x$  is equal to  $M(x) - M(x')$ .

## Implementation invariance

If two models  $M$  and  $M'$  have identical input/output behavior, then the attributions for  $M$  and  $M'$  are identical.

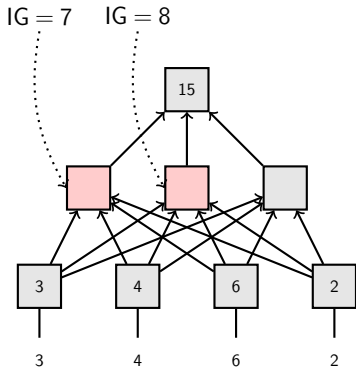
# Integrated Gradients: Computation

$$\text{IG}_i(M, x, x') = \underbrace{(x_i - x'_i)}_5 \cdot \underbrace{\sum_{k=1}^4}_4 \frac{\underbrace{\partial M(x' + \frac{k}{m} \cdot (x - x'))}_2}_{\partial x_i} \cdot \underbrace{\frac{1}{m}}_4$$

1. Generate  $\alpha = [1, \dots, m]$
2. Interpolate inputs between baseline  $x'$  and actual input  $x$
3. Compute gradients for each interpolated input
4. Integral approximation through averaging
5. Scaling to remain in the space region as the original

Adapted from the [TensorFlow integrated gradients tutorial](#)

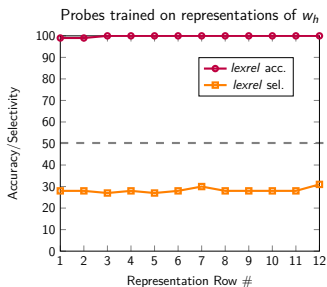
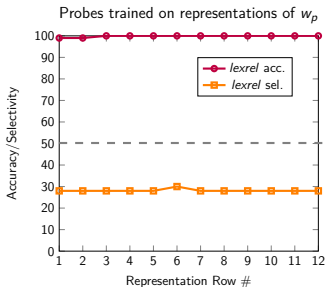
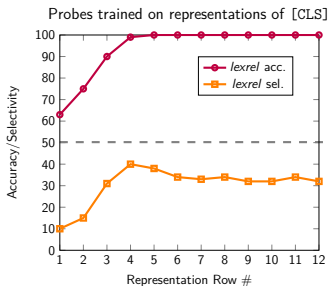
# Reliable insights about causal structure



$$W_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad W_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \quad W_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\mathbf{w} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \quad (\mathbf{x}W_1; \mathbf{x}W_2; \mathbf{x}W_3)\mathbf{w}$$

# Probe results for lexrel accuracy



## MoNLI causal abstraction analysis details

1. A systematic generalization task
2. Methods and findings
3. Largest exchangeable cluster
4. Which algorithm is BERT implementing then?

# A systematic generalization task

NMoNLI Train		NMoNLI Test	
person	198	dog	88
instrument	100	building	64
food	94	ball	28
machine	60	car	12
woman	58	mammal	4
music	52	animal	4
tree	52		
boat	46		
fruit	42		
produce	40		
fish	40		
plant	38		
jewelry	36		
anything	34		
hat	20		
man	20		
horse	16		
gun	12		
adult	10		
shirt	8		
shoe	6		
store	6		
cake	4		
individual	4		
clothe	2		
weapon	2		
creature	2		

Our models know these lexical relations (high Positive MoNLI accuracy) and will be compelled to combine this knowledge with what they learn about negation during Negative MoNLI fine-tuning.

## Methods and findings

1. Find a useful intervention point.
2. Interchange interventions for every pair of examples at that site.
3. Find clusters of examples in which BERT mimics the causal dynamics of INFER.
4. The largest subsets we found 98, 63, 47, and 37.
  - a. For a random graph, the expected number of subsets larger than 20 is effectively 0.
  - b. If the site perfectly captured INFER, we would get a single huge cluster.

# What it means for BERT to implement Infer

INFER(*example*)

- 1 *lexrel* ← GET-LEXREL(*example*)
- 2 **if** CONTAINS-NOT(*example*)
- 3     **return** REVERSE(*lexrel*)
- 4 **return** *lexrel*

$$\text{INFER}_{lexrel(i) \rightarrow lexrel(j)}(i) = \begin{cases} \text{INFER}(i) & lexrel(i) = lexrel(j) \\ \text{REVERSE}(\text{INFER}(i)) & lexrel(i) \neq lexrel(j) \end{cases}$$

$$\text{INFER}_{lexrel(i) \rightarrow lexrel(j)}(i) = \text{BERT}_{L(i) \rightarrow L(j)}(i)$$

# Largest exchangeable cluster

	(cemetery,location)		(dogs,huskies)		(hood,thing)	
	(house,location)	(den,location)	(dog,husky)	(dog,chiuahua)	(nut,thing)	(capsule,thing)
	(ghetto,location)	(backyard,location)	(dog,retriever)	(dog,maltese)	(pouch,thing)	(structure,thing)
	(jungle,location)	(meadow,location)	(dog,terrier)	(dog,pomeranian)	(root,thing)	(nugget,thing)
(laboratory,location)	(playground,location)	(studio,location)	(beetle,insect)		(tube,thing)	
(slum,location)	(station,location)	(farm,location)	(grasshopper,insect)	(bee,insect)		
(lab,location)	(campsite,location)		(wasp,insect)	(fly,insect)	(cricket,insect)	
	(town,location)	(lawn,location)	(butterfly,insect)	(bumblebee,insect)		
			(flea,insect)	(roach,insect)	(moth,insect)	
			(mosquito,insect)			
(saxophone,instrument)	(flute,instrument)		(person,vegetarian)	(person,lunatic)	(box,object)	
(bass,instrument)	(piano,instrument)		(person,repulican)	(person,trooper)	(object,sweater)	(hat,object)
(violin,instrument)	(tuba,instrument)		(person,business)	(person,navigator)	(object,jacket)	(toy,object)
	(harmonica,instrument)	(person,steward)	(person,consultant)	(person,goalkeeper)	(cane,object)	
		(person,farmer)			(water,rainwater)	
	(liquid,whiskey)	(person,sophomore)	(person,housekeeper)		(water,saltwater)	
(liquid,margarita)	(liquid,tequila)	(person,cleanser)	(person,physicist)	(person,cop)	(sculptor,artist)	
(liquid,alcohol)		(person,cambodian)	(person,detective)		(berry,blueberry)	
	(woman,granny)	(person,genius)	(person,sergeant)	(person,californian)	(tree,cypress)	
	(woman,widow)	(person,doctor)	(person,runner)		(tree,magnolia)	(trees,elms)
					(tree,maple)	

# Which algorithm is BERT implementing then?

INFER(*example*)

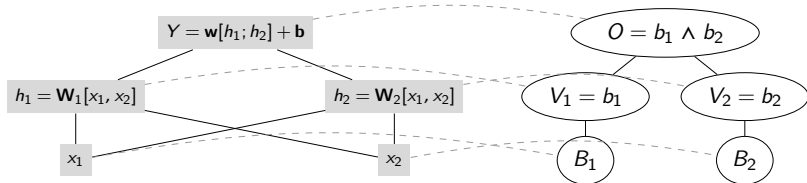
```
1 lexrel ← GET-LEXREL(example)
2 if CONTAINS-NOT(example)
3     return REVERSE(lexrel)
4 return lexrel
```

INFER(*example*)

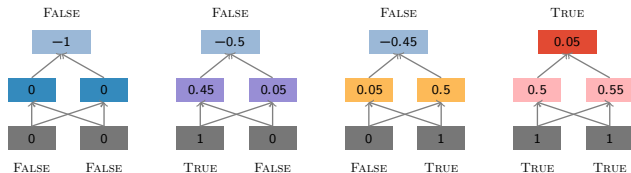
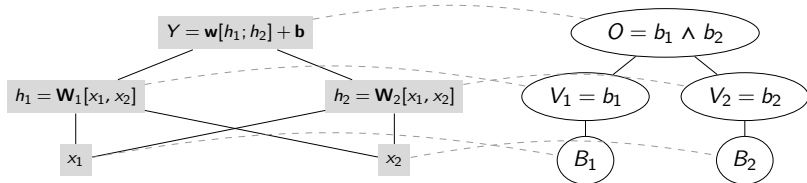
```
1 if INCLUSTER( $C_1$ , example)
2     lexrel1 ← GET-LEXREL(example)
3     if CONTAINS-NOT(example)
4         return REVERSE(lexrel1)
5     return lexrel1
6 if INCLUSTER( $C_2$ , example)
7     lexrel2 ← GET-LEXREL(example)
8     if CONTAINS-NOT(example)
9         return REVERSE(lexrel2)
10    return lexrel2
11 if INCLUSTER( $C_3$ , example)
12    lexrel3 ← GET-LEXREL(example)
13    if CONTAINS-NOT(example)
14        return REVERSE(lexrel3)
15    return lexrel3
16 ...
```

# IIT induces causal structure

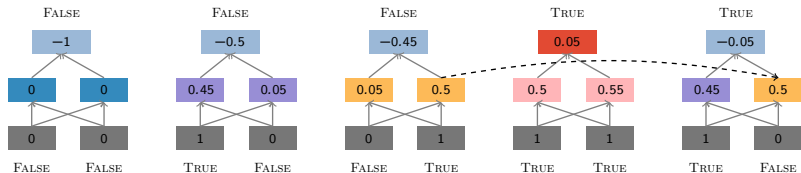
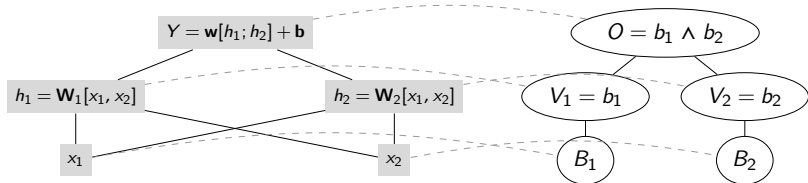
# IIT induces causal structure



# IIT induces causal structure



# IIT induces causal structure



# IIT induces causal structure

