Interpretability and Analysis in Neural NLP

Yonatan Belinkov, Sebastian Gehrmann, Ellie Pavlick

ACL Tutorial
July 5, 2020

Yonatan Belinkov
Harvard & MIT → Technion

Sebastian Gehrmann
Google Research

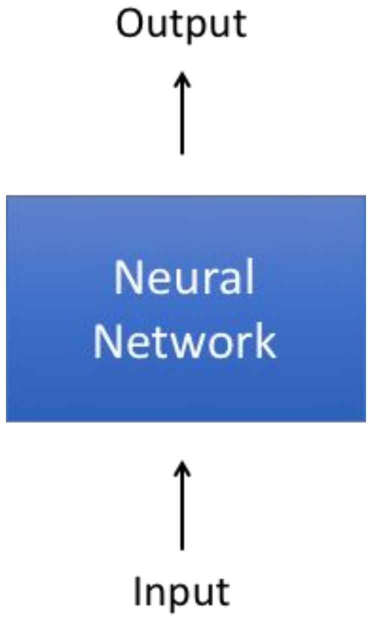
Ellie Pavlick
Brown University

Who should take this tutorial?

- This tutorial will present the main problems and approaches in interpreting and analyzing modern NLP models
- Target audience
  - NLP researchers and practitioners
  - We assume familiarity with mainstream NLP models and tasks
  - Anyone who wants to analyze NLP models or think critically about using current interpretation methods
- We aim to highlight key studies in the field
  - We do not aim to be exhaustive
  - We provide pointers to important references
  - We emphasize methodological limitations and opportunities

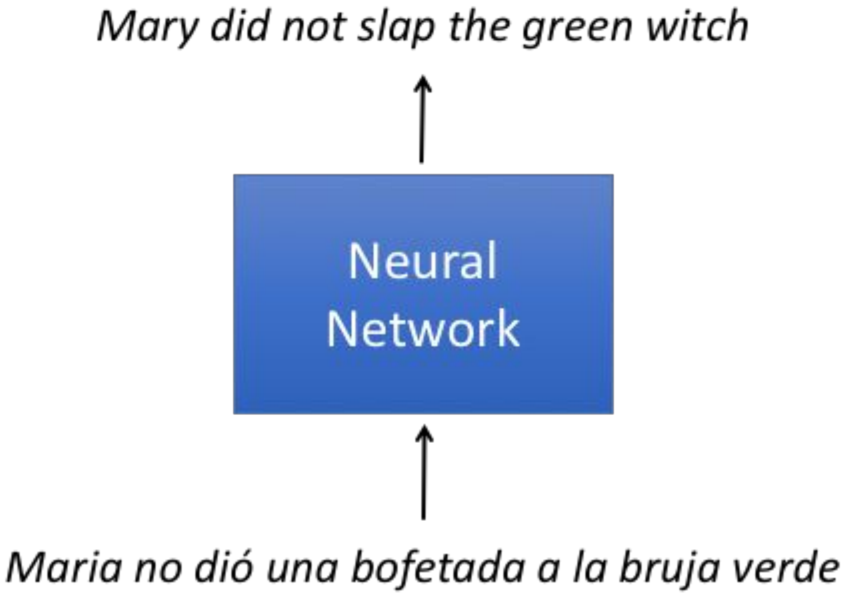
End-to-End Learning

- The predominant approach in NLP these days is end-to-end learning
- Learn a model $f : x \rightarrow y$, which maps input x to output y



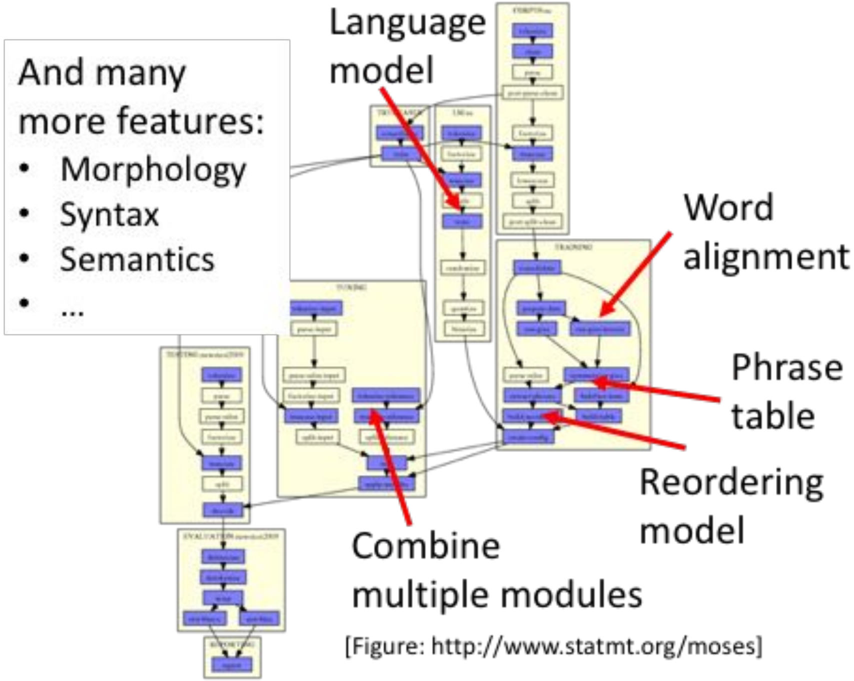
End-to-End Learning

- For example, in machine translation we map a source sentence to a target sentence, via a deep neural network:



A Historical Perspective

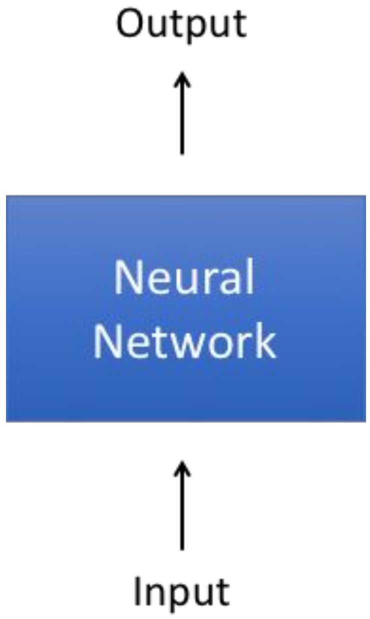
- Compare this with a traditional statistical approach to MT, based on multiple modules and features:



[Figure: http://www.statmt.org/moses]

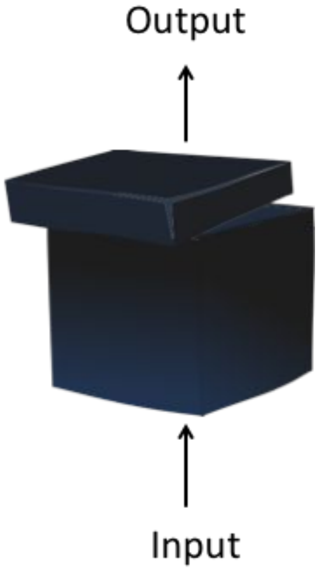
End-to-End Learning

- The predominant approach in NLP these days is end-to-end learning, where all parts of the model are trained on the same task:



How can we open the black box?

- Given *f : x* → *y*, we want to ask some questions about *f*
    - What is its internal structure?
    - How does it behave on different data?
    - Why does it make certain decisions?
    - When does it succeed/fail?
    - ...



Why should we care?

- Much deep learning research:
  - Trial-and-error, shot in the dark
  - Better understanding → better systems

Why should we care?

- Much deep learning research:
  - Trial-and-error, shot in the dark
  - Better understanding → better systems

- Accountability, trust, and bias in machine learning
  - “Right to explanation”, EU regulation
  - Life threatening situations: healthcare, autonomous cars
  - Better understanding → more accountable systems

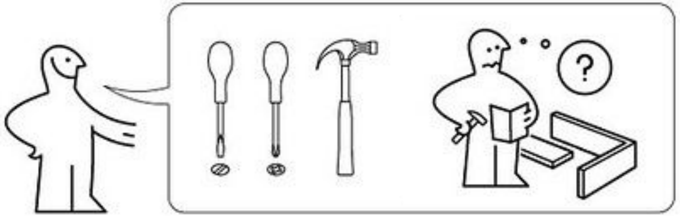
Why should we care?

- Much deep learning research:
  - Trial-and-error, shot in the dark
  - Better understanding → better systems
- Accountability, trust, and bias in machine learning
  - "Right to explanation", EU regulation
  - Life threatening situations: healthcare, autonomous cars
  - Better understanding → more accountable systems
- Neural networks aid the scientific study of language (Linzen 2019)
  - Models of human language acquisition
  - Models of human language processing
  - Better understanding → more interpretable models

Goal for today

1. Understand the toolbox of interpretability methods in NLP

2. Have an idea which tool to apply to a problem



Analysis Questionnaire

What is the goal of the study?

Pedagogical / Debugging / Debiasing / ...

Understanding model structure / model decisions / data / ...

How do you quantify an outcome?

Who is your user or target group?

ML or NLP Expert/ Domain Expert / Student / Lay User of the System ...

How much domain/ model knowledge do they have?

Outline

- Structural analyses Yonatan
- Behavioral analyses Ellie
- Interaction + Visualization Sebastian
- Other methods

Outline

- **Structural analyses**
- Behavioral analyses
- Interaction + Visualization
- Other methods

Structural Analyses

- Let $f : x \rightarrow y$ be a model mapping an input x to an output y
  - f might be a complicated neural network with many layers or other components
  - For example, $f^l(x)$ might be the output of the network at the l-th layer
- Some questions we might want to ask:
  - What is the role of different components of f?
  - What kind of information do different components capture?
  - More specifically: Does components A know something about property B?

Structural Analyses

- Let $f : x \rightarrow y$ be a model mapping an input x to an output y
  - f might be a complicated neural network with many layers or other components
  - For example, $f^l(x)$ might be the output of the network at the l-th layer

Structural Analyses

- Let $f : x \rightarrow y$ be a model mapping an input x to an output y
  - f might be a complicated neural network with many layers or other components
  - For example, $f^l(x)$ might be the output of the network at the l-th layer
- Analysis via a probing classifier
  - Assume a corpus of inputs x with linguistic annotations z
  - Generate representations of x from some part of the model f, for example representations $f^l(x)$ at a certain layer
  - Train another classifier $g : f^l(x) \rightarrow z$ that maps the representations $f^l(x)$ to the property z
  - Evaluate the accuracy of g as a proxy to the quality of representations $f^l(x)$ w.r.t property z

Structural Analyses

- Let $f : x \rightarrow y$ be a model mapping an input x to an output y
  - f might be a complicated neural network with many layers or other components
  - For example, $f^l(x)$ might be the output of the network at the l-th layer
- Analysis via a probing classifier
  - Assume a corpus of inputs x with linguistic annotations z
  - Generate representations of x from some part of the model f, for example representations $f^l(x)$ at a certain layer
  - Train another classifier $g : f^l(x) \rightarrow z$ that maps the representations $f^l(x)$ to the property z
  - Evaluate the accuracy of g as a proxy to the quality of representations $f^l(x)$ w.r.t property z
- In information theoretic terms:
  - Set $h = f(x)$ and recall that $I(h; z) = H(z) - H(z \mid h)$
  - Then the probing classifier minimizes $H(z \mid h)$, or maximizes $I(h, z)$

Milestones (partial list)

	f	*x*	*y*	*g*	*z*
Köhn 2015	Word embedding	Word	Word	Linear	POS, morphology
Ettinger et al. 2016	Sentence embedding	Word, sentence	Word, sentence	Linear	Semantic roles, scope
Shi et al. 2016	RNN MT	Word, sentence	Word, sentence	Linear / tree decoder	Syntactic features, tree
Adi et al. 2017 Conneau et al. 2018	Sentence embedding	Sentence	Sentence	Linear, MLP	Surface, syntax, semantics
Hupkes et al. 2018	RNN, treeRNN	*five plus three*	*eight*	Linear	Position, cumulative value
Hewitt+Manning 2019	ELMo, BERT	Sentence	Sentence	Linear	Full tree

Example Results

- Numerous papers use this methodology to study:
  - Linguistic phenomena (*z*): phonology, morphology, syntax, semantics
  - Network components (*f*): word embeddings, sentence embeddings, hidden states, attention weights, etc.
- We'll show example results on machine translation
- Much more related work reviewed in our survey (Belinkov and Glass 2019)

Example: Machine Translation

- Setup
  - *f* : an RNN encoder-decoder MT model
  - *x* and *y* are source and target sentences (lists of words)
  - *g*: a non-linear classifier (MLP with one hidden layer)
  - *z*: linguistic properties of words in *x* or *y*

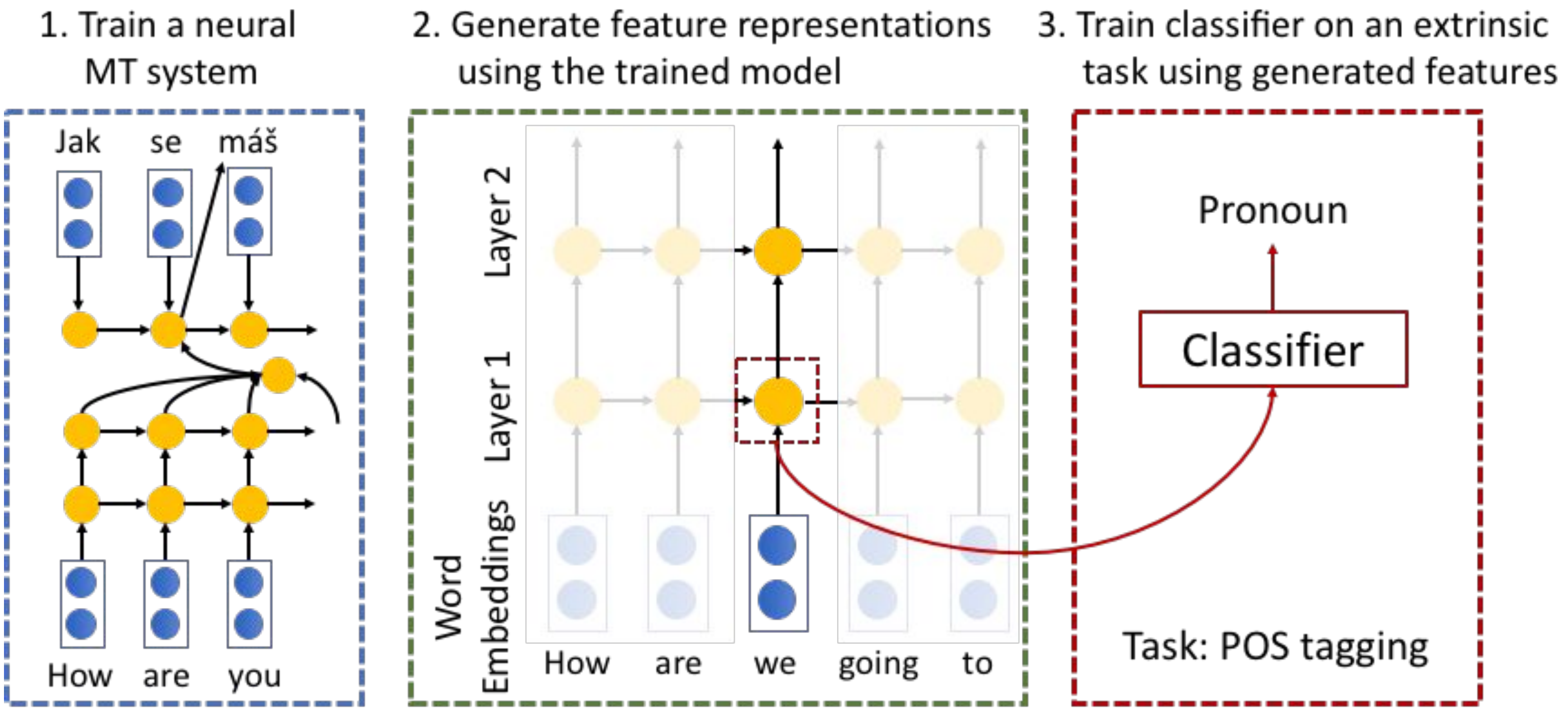
Example: Machine Translation

- Setup
  - *f* : an RNN encoder-decoder MT model
  - *x* and *y* are source and target sentences (lists of words)
  - *g*: a non-linear classifier (MLP with one hidden layer)
  - *z*: linguistic properties of words in *x* or *y*
- Morphology:
  - A challenge for machine translation, previously solved with feature-rich approaches
  - Do neural networks acquire morphological knowledge?

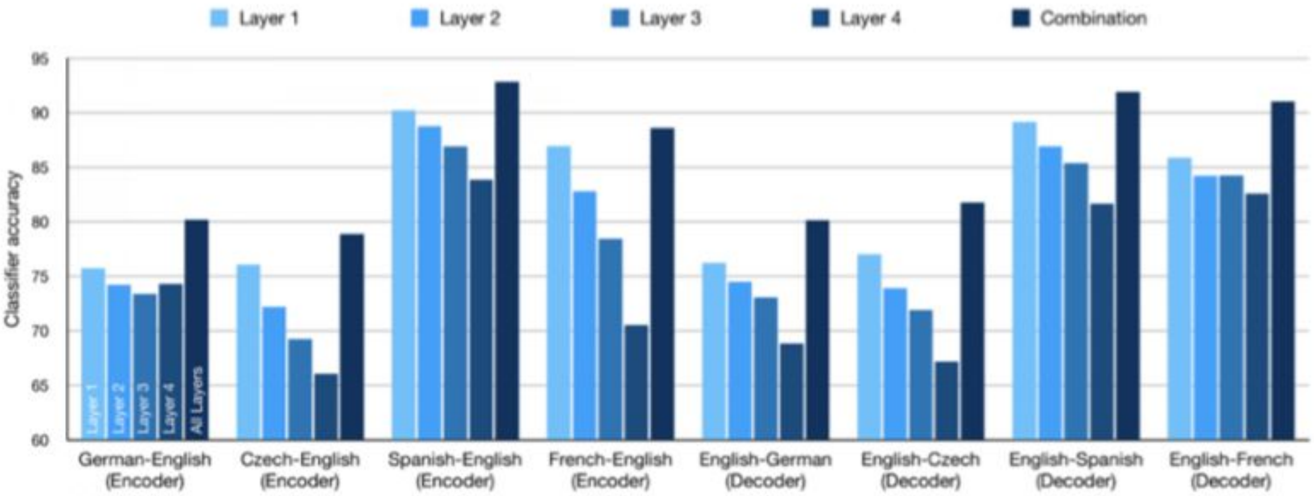
Example: Machine Translation

- Setup
  - *f* : an RNN encoder-decoder MT model
  - *x* and *y* are source and target sentences (lists of words)
  - *g*: a non-linear classifier (MLP with one hidden layer)
  - *z*: linguistic properties of words in *x* or *y*
- Morphology:
  - A challenge for machine translation, previously solved with feature-rich approaches.
  - Do neural networks acquire morphological knowledge?
- Experiment
  - Take $f^l(x)$, an RNN hidden state at layer *l*
  - Predict *z*, a morphological tag (*verb-past-singular-feminine*, *noun-plural*, etc.)
  - Compare accuracy at different layers *l*

Example: Machine Translation



Machine Translation: Morphology



- Lower is better
- But deeper models translate better → what's going on in top layers?

Example: Machine Translation

- Setup
  - *f* : an RNN encoder-decoder MT model
  - *x* and *y* are source and target sentences (lists of words)
  - *g*: a non-linear classifier (MLP with one hidden layer)
  - *z*: linguistic properties of words in *x* or *y*

Probing Classifiers Questionnaire

What is the goal of the study?

Scientific / Pedagogical / **Debugging** / Debiasing / **...**

Understanding model structure / model decisions / data / ...

How do you quantify an outcome? **Performance comparisons**

Who is your user or target group?

ML or NLP Expert / Domain Expert / Student / Lay User of the System ...

How much domain/ model knowledge do they have? **Enough to understand the model and problem domain**

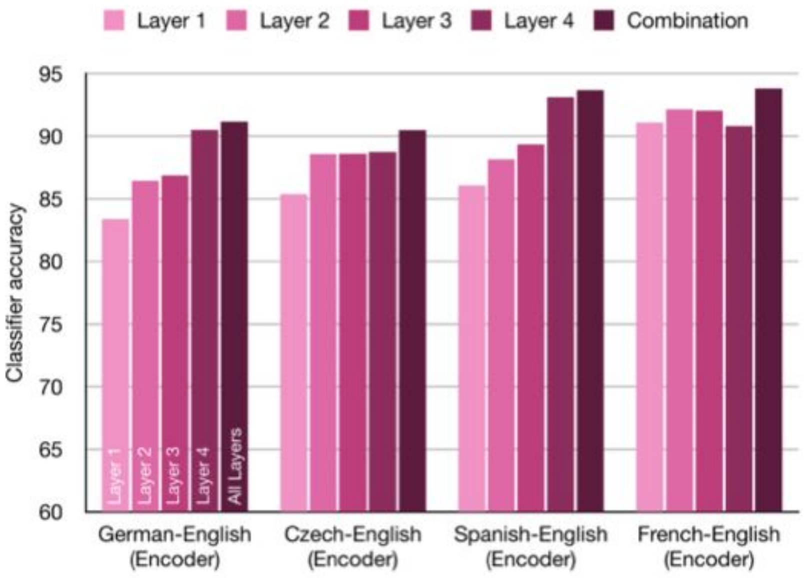
Example: Machine Translation

- Setup
  - *f* : an RNN encoder-decoder MT model
  - *x* and *y* are source and target sentences (lists of words)
  - *g*: a non-linear classifier (MLP with one hidden layer)
  - *z*: linguistic properties of words in *x* or *y*
- Syntax:
  - A challenge for machine translation, previously solved with hierarchical approaches.
  - Do neural networks acquire syntactic knowledge?

Example: Machine Translation

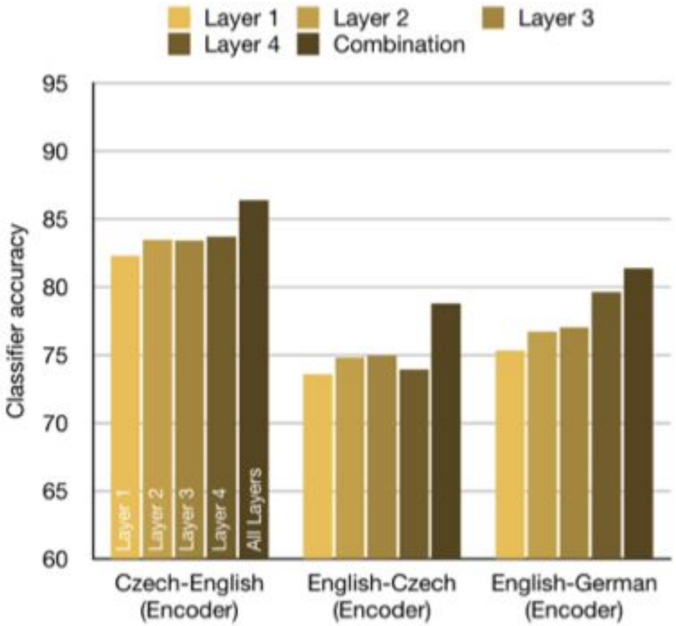
- Setup
  - f : an RNN encoder-decoder MT model
  - x and y are source and target sentences (lists of words)
  - g: a non-linear classifier (MLP with one hidden layer)
  - z: linguistic properties of words in x or y
- Syntax:
  - A challenge for machine translation, previously solved with hierarchical approaches.
  - Do neural networks acquire syntactic knowledge?
- Experiment
  - Take $[f^l(x_i) ; f^l(x_j)]$, RNN hidden states of words x_i and x_j, at layer l
  - Predict z, a dependency label (*subject, object*, etc.) between words x_i and x_j
  - Compare accuracy at different layers l

Machine Translation: Syntactic Relations



- Higher is better

Machine Translation: Semantic Relations



- Higher is better

Hierarchies

Language
Hierarchy

Semantics

Discourse

Propositions

Roles

Syntax

Trees

Phrases

Relations

Morpho-Syntax

Parts-of-speech

Morphology

Lexicon

Hierarchies

Speech Hierarchy

Words

Syllables

Phonemes

Complex

Simple

Articulatory features

Place

Manner

⋮

Language Hierarchy

Semantics

Discourse

Propositions

Roles

Syntax

Trees

Phrases

Relations

Morpho-Syntax

Parts-of-speech

Morphology

Lexicon

Vision Hierarchy

Scenes

Objects

Object parts

Motifs

Edges

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Suppose we get an accuracy, what should we compare it to?
  - Many studies focus on relative performance (say, comparing different layers)
  - But it may be desirable to compare to external numbers
  - **Baselines**: Often, compare to using static word embeddings (Belinkov et al. 2017) or random features (Zhang and Bowman 2018)
    - This tells us that a representation is non-trivial
  - **Skylines**: Sometimes, report the state-of-the-art on the task, or train a full-fledged model
    - This can tell us how much is missing from the representation

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Suppose we get an accuracy, what should we compare it to?
  - Hewitt and Liang (2019) define control tasks: tasks that only g can learn, not f
    - Specifically, assign a random label to each word type
  - A "good" probe should be selective: high linguistic task accuracy, low control task accuracy
  - Example
    - Linear vs. MLP
    - Accuracy vs. selectivity

Part-of-speech Tagging				
	Linear		MLP-1	
Model	Accuracy	Selectivity	Accuracy	Selectivity
Proj0	96.3	20.6	97.1	1.6
ELMo1	97.2	26.0	97.3	4.5
ELMo2	96.6	31.4	97.0	8.8

Probing Classifiers: Limitations

- Recall the setup:
    - Original model $f : x \rightarrow y$
    - Probing classifier $g : f(x) \rightarrow z$
    - g maximizes the mutual information between the representation $f(x)$ and property z

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Common wisdom: use a linear classifier to focus on the representation and not the probe
  - Anecdotal evidence: non-linear classifiers achieve better probing accuracy, but do not change the qualitative patterns (Conneau et al. 2018, Belinkov 2018)

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Pimentel et al. (2020) argue that we should always choose the most complex probe g, since it will maximize the mutual information $I(h; z)$, where $f(x)=h$

Probing Classifiers: Limitations

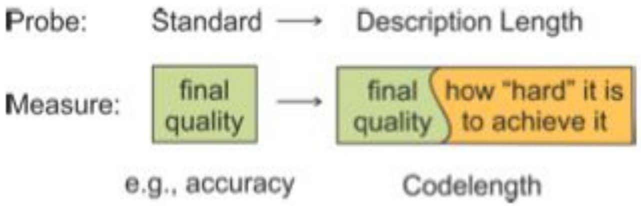
- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Pimentel et al. (2020) argue that we should always choose the most complex probe g, since it will maximize the mutual information $I(h; z)$, where $f(x)=h$
  - They also show that $I(x; z) = I(h; z)$ (under mild assumptions)
    - Thus the representation $f(x) := h$ contains the same amount of information about z as x
  - Does this make the probing endeavor obsolete?

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Pimentel et al. (2020) argue that we should always choose the most complex probe g, since it will maximize the mutual information $I(h; z)$, where $f(x)=h$
  - They also show that $I(x; z) = I(h; z)$ (under mild assumptions)
    - Thus the representation $f(x) := h$ contains the same amount of information about z as x
  - Does this make the probing endeavor obsolete?
  - Not necessarily:
    - We would still like to know how good a representation is *in practice*
    - We can still ask relative questions about *ease of extraction* of information

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Voita and Titov (2020) measure both probe complexity and probe quality
  - Instead of measuring accuracy, estimate the minimum description length: how many bits are required to transmit z knowing $f(x)$, plus the cost of transmitting g
  - Variational code: incorporate cost of transmitting g
  - Online code: incrementally train g on more data



Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- What is g? What is the relation between the probe g and the model f?
  - Voita and Titov (2020) measure both probe complexity and probe quality
  - Instead of measuring accuracy, estimate the minimum description length: how many bits are required to transmit z knowing $f(x)$, plus the cost of transmitting g
  - Variational code: incorporate cost of transmitting g
  - Online code: incrementally train g on more data
  - Example
    - Layer 0 control: control accuracy is high (96.3) but at the expense of codelength (267)

	Accuracy	codelength
MLP-2, h=1000		
LAYER 0	93.7 / 96.3	163 / 267
LAYER 1	97.5 / 91.9	85 / 470
LAYER 2	97.3 / 89.4	103 / 612

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Correlation vs. causation
  - The common setup only measures correlation between representation $f(x)$ and property z
  - It is not directly linked to the *behavior* of the model f on the task it was trained on, that is, predicting y
  - Some work found negative/lack of correlation between probe and task quality (Vanmassenhove et al. 2017, Cífka and Bojar 2018)
  - An alternative direction: intervene in the model representations to discover causal effects on prediction (Giulianelli et al. 2018, Bau et al. 2019, Vig et al. 2020, Feder et al. 2020)

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Giulianelli et al. 2018 train a classifier to predict number from LSTM states
  - Then backprop classifier gradients to change LSTM states so they predict number better

Probing Classifiers: Limitations

- Recall the setup:
  - Original model *f : x → y*
  - Probing classifier *g : f(x) → z*
  - *g* maximizes the mutual information between the representation *f(x)* and property *z*
- Alternative: causal interpretation via intervention
  - [Giulianelli et al. 2018](#) train a classifier to predict number from LSTM states
  - Then backprop classifier gradients to change LSTM states so they predict number better
  - They find:
    - improved probing accuracy, little effect on LM
    - strong effect on an LM agreement test
  - Important connection between the classifier *g* and the behavior of the model *f*

	without intervention	with intervention
Original	78.1	85.4
Nonce	70.7	75.6

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Bau et al. 2019 study the role of individual neurons in MT
  - They identify important neurons and intervene in their behavior
  - Change their activations based on activation statistics over a corpus
    - Move towards the mean activation over a property (say, verb tense)

Probing Classifiers: Limitations

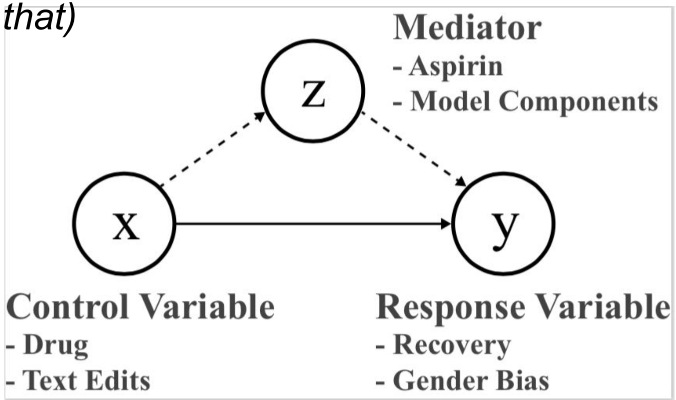
- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Bau et al. 2019 study the role of individual neurons in MT
  - They identify important neurons and intervene in their behavior
  - Change their activations based on activation statistics over a corpus
    - Move towards the mean activation over a property (say, verb tense)
  - Successfully influence the translation of tense from past to present (67% success rate)
  - Less successful with influencing gender and number (20-30%)

Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Vig et al. 2020 use causal mediation analysis to interpret gender bias in language models
  - Define interventions via text edit operations and measure counterfactual outcomes
    - $p(\textit{she} \mid \textit{the nurse said that})$ vs. $p(\textit{she} \mid \textit{the man said that})$

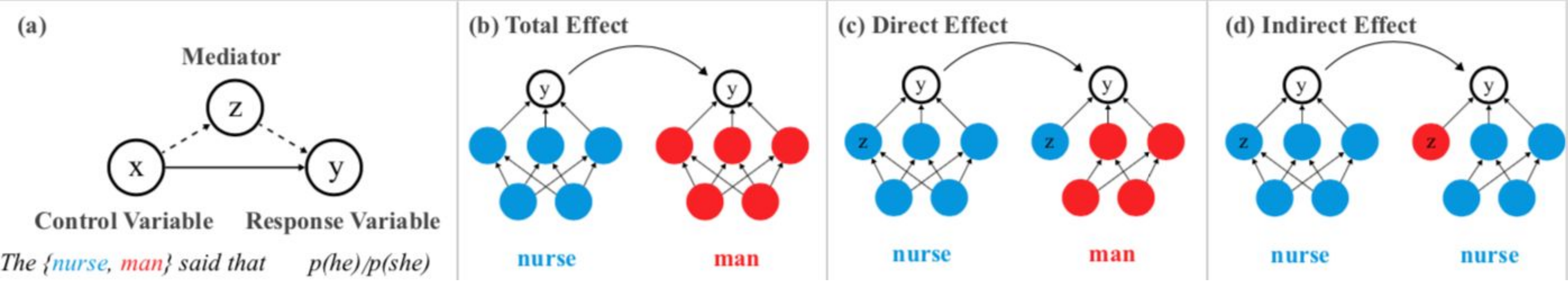
Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Vig et al. 2020 use causal mediation analysis to interpret gender bias in language models
  - Define interventions via text edit operations and measure counterfactual outcomes
    - *p(she | the nurse said that)* vs. *p(she | the man said that)*
  - Examine mediators: neurons and attention heads
  - Calculate direct and indirect effects



Probing Classifiers: Limitations

- Recall the setup:
  - Original model $f : x \rightarrow y$
  - Probing classifier $g : f(x) \rightarrow z$
  - g maximizes the mutual information between the representation $f(x)$ and property z
- Alternative: causal interpretation via intervention
  - Vig et al. 2020 use causal mediation analysis to interpret gender bias in language models



Outline

- Structural analyses
- **Behavioral analyses**
- Interaction + Visualization
- Other methods

Behavioral Analyses

- Usually, we measure the *average-case* performance of $f : x \rightarrow y$ on a test set $\{x,y\}$, drawn uniformly at random from some text corpus

Behavioral Analyses

- Usually, we measure the *average-case* performance of $f : x \rightarrow y$ on a test set $\{x,y\}$, drawn uniformly at random from some text corpus
- However, this can reward models for performing well on common phenomena, and hide the fact that they perform poorly on “the tail”

Behavioral Analyses

- Usually, we measure the *average-case* performance of $f : x \rightarrow y$ on a test set $\{x,y\}$, drawn uniformly at random from some text corpus
- However, this can reward models for performing well on common phenomena, and hide the fact that they perform poorly on “the tail”
- Challenge sets, a.k.a test suites aim to cover specific, diverse phenomena
  - Systematicity
  - Exhaustivity
  - Control over data

Behavioral Analyses

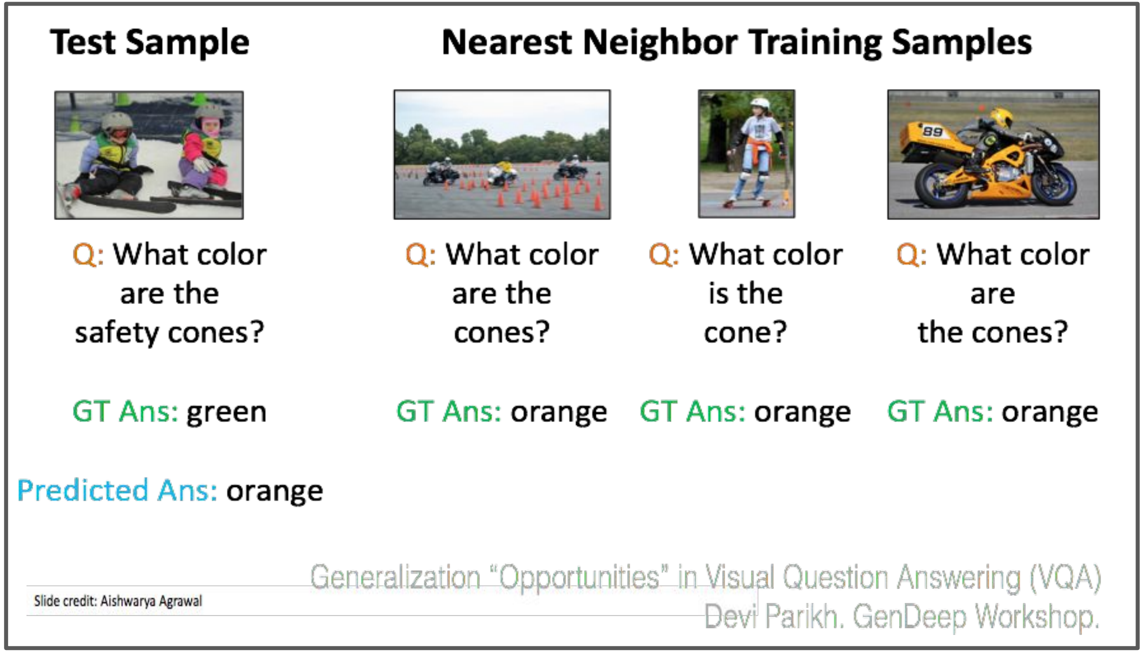
- Usually, we measure the *average-case* performance of $f : x \rightarrow y$ on a test set $\{x,y\}$, drawn uniformly at random from some text corpus
- However, this can reward models for performing well on common phenomena, and hide the fact that they perform poorly on “the tail”
- Challenge sets, a.k.a test suites aim to cover specific, diverse phenomena
  - Systematicity
  - Exhaustivity
  - Control over data
- Thus they facilitate *fine-grained* analysis of model performance
- And they have a long history in NLP evaluation (Lehmann et al. 1996, Cooper et al. 1996, …)

Behavioral Analyses

- Key idea: Design experiments that allow us to make inferences about the model's representation based on the model's behavior.

Behavioral Analyses

- Key idea: Design experiments that allow us to make inferences about the model's representation based on the model's behavior.



Behavioral Analyses

- Key idea: Design experiments that allow us to make inferences about the model’s representation based on the model’s behavior.

Test Sample

Q: What color are the safety cones?

GT Ans: green

Predicted Ans: orange

Nearest Neighbor Training Samples

Q: What color are the cones?

GT Ans: orange

Q: What color is the cone?

GT Ans: orange

Q: What color are the cones?

GT Ans: orange

Slide credit: Aishwarya Agrawal

Generalization “Opportunities” in Visual Question Answering (VQA)
Devi Parikh. GenDeep Workshop.

Brett knew what many waiters find ✔

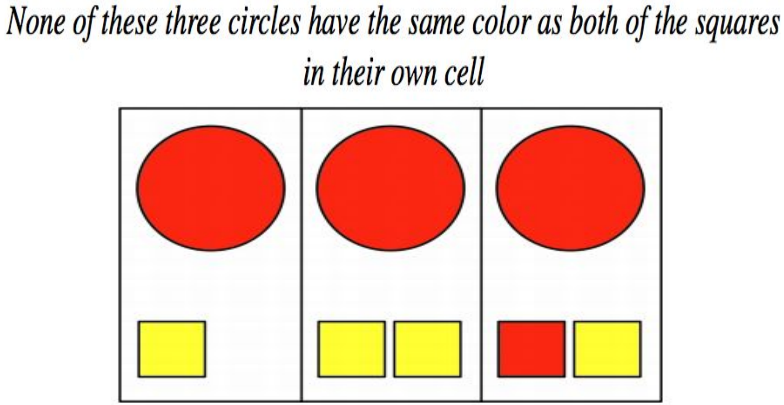
Brett knew that many waiters find. ✘

Warstadt et al. (2020)

Behavioral Analyses

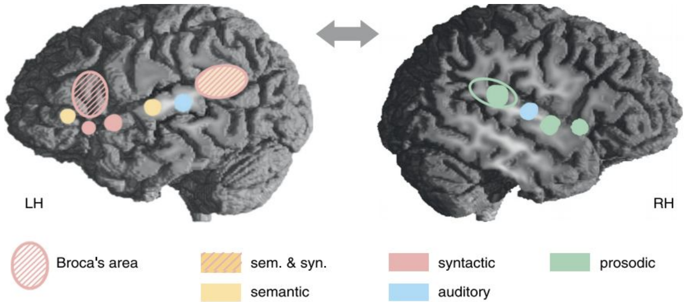
- Key idea: Design experiments that allow us to make inferences about the model's representation based on the model's behavior.
- As with theories about human language representations: Claims about how a model works must be consistent with *both* physiological and behavioral data

Behavioral



On the semantics of phi features on pronouns. Sudo (2012).

Structural



The Brain Basis of Language Processing: From Structure to Function. Friederici (2011)

Behavioral Analyses

- Benefits:

- Limitations

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data

- Limitations

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
  - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing

- Limitations

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
  - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
  - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.

- Limitations

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
  - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
  - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.
  - Practical--not whether the model represents a feature, but whether it uses it in the right way
- Limitations

Behavioral Analyses

- Benefits:
    - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
    - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
    - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.
    - Practical--not whether the model represents a feature, but whether it uses it in the right way
- Limitations
    - What’s to blame, the model or the data? How do we know what generalizations are “fair”?

Behavioral Analyses

- Benefits:
    - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
    - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
    - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.
    - Practical--not whether the model represents a feature, but whether it uses it in the right way
- Limitations
    - What’s to blame, the model or the data? How do we know what generalizations are “fair”?
    - Only tells us *that* a model did/didn’t solve a task; few insights into *how* the model solved the task, or *why* it failed to

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
  - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
  - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.
  - Practical--not whether the model represents a feature, but whether it uses it in the right way
- Limitations
  - What’s to blame, the model or the data? How do we know what generalizations are “fair”?
  - Only tells us *that* a model did/didn’t solve a task; few insights into *how* the model solved the task, or *why* it failed to
  - Hard to design tightly controlled stimuli, probing sets themselves can have artifacts

Behavioral Analyses

- Benefits:
  - Theory agnostic, avoids prescriptivism. No constraints on how you represent it (symbolic, neural, feature-engineered) as long as it explains the data
  - Avoid “squinting at the data”. Objective criteria for what counts as “representing” a thing
  - Interfaces well with linguistics and other fields. “We are all responsible for the same data”.
  - Practical--not whether the model represents a feature, but whether it uses it in the right way
- Limitations
  - What’s to blame, the model or the data? How do we know what generalizations are “fair”?
  - Only tells us *that* a model did/didn’t solve a task; few insights into *how* the model solved the task, or *why* it failed to
  - Hard to design tightly controlled stimuli, probing sets themselves can have artifacts
  - Risk of overfitting to the challenge sets

Challenge Sets Questionnaire

What is the goal of the tool?

Scientific / Pedagogical / **Debugging** / Debiasing / **...**

Understanding model structure / **model decisions** / data / ...

How do you quantify an outcome? **(Relative) accuracy across different challenge sets**

Who is your user?

ML or NLP Expert/ Domain Expert / Student / ...

How much domain/model knowledge do they have? **Knowledge of target phenomena, but no model knowledge**

The answers will inform the following implementation questions:

Does the tool require interaction with the model? With the data? **Model treated as a “black box”**

Can you change the model structure or model decisions? **No**

Behavioral Analyses

- See recent Belinkov & Glass survey for a categorization of many studies
- Tasks
  - Especially machine translation and natural language inference
- Linguistic phenomena
  - Morphology, syntax, lexical semantics, predicate-argument structure
- Languages
  - Mostly focusing on English, some artificial languages, not much work on other languages
- Scale
  - Ranging from hundreds to many thousands
- Construction method
  - Either manual or programmatic

Tasks used as probing tasks

- Ideally, simple task interfaces which can support lots of model types
- Ideally, minimal need for training/finetuning on top of model being “probed”

Task	Example	Typical Use	Strengths	Limitations	E.g.

Task	Example	Typical Use	Strengths	Limitations	E.g.
LM /Generation?	The boy by the boats [is/*are] smiling.	Syntactic phenomena	No additional training on top of pretrained LM	Often uses ppl, so best for left-to-right language models. Harder to use for newer variants.	Linzen et al. (2016)

Task	Example	Typical Use	Strengths	Limitations	E.g.
LM /Generation?	The boy by the boats [is/*are] smiling.	Syntactic phenomena	No additional training on top of pretrained LM	Often uses ppl, so best for left-to-right language models. Harder to use for newer variants.	Linzen et al. (2016)
Acceptability	The boy by the boats [is/*are] smiling.	Syntactic and semantic phenomena	More flexible than LM across architectures; well studied in ling.	Usually requires additional training on top of LM.	Warstadt et al. (2020)

Task	Example	Typical Use	Strengths	Limitations	E.g.
LM /Generation?	The boy by the boats [is/*are] smiling.	Syntactic phenomena	No additional training on top of pretrained LM	Often uses ppl, so best for left-to-right language models. Harder to use for newer variants.	Linzen et al. (2016)
Acceptability	The boy by the boats [is/*are] smiling.	Syntactic and semantic phenomena	More flexible than LM across architectures; well studied in ling.	Usually requires additional training on top of LM.	Warstadt et al. (2020)
NLI	The boy is smiling. -> The boy [is/*is not] happy.	Semantics/pragmatics/ world knowledge	Flexible, easy to “recast” many tasks to NLI; long history	Often awkward sentences/confounds; low human agreement	White et al. (2017)

Task	Example	Typical Use	Strengths	Limitations	E.g.
LM /Generation?	The boy by the boats [is/*are] smiling.	Syntactic phenomena	No additional training on top of pretrained LM	Often uses ppl, so best for left-to-right language models. Harder to use for newer variants.	Linzen et al. (2016)
Acceptability	The boy by the boats [is/*are] smiling.	Syntactic and semantic phenomena	More flexible than LM across architectures; well studied in ling.	Usually requires additional training on top of LM.	Warstadt et al. (2020)
NLI	The boy is smiling. -> The boy [is/*is not] happy.	Semantics/pragmatics/ world knowledge	Flexible, easy to "recast" many tasks to NLI; long history	Often awkward sentences/confounds; low human agreement	White et al. (2017)
Generation	Dante was born in [Mask]	Semantics/pragmatics/ world knowledge	Can be more natural than NLI; incorporates more context	Hard to auto evaluate, esp. beyond one word/factoid questions	Petroni et al. (2019)

Task	Example	Typical Use	Strengths	Limitations	E.g.
LM /Generation?	The boy by the boats [is/*are] smiling.	Syntactic phenomena	No additional training on top of pretrained LM	Often uses ppl, so best for left-to-right language models. Harder to use for newer variants.	Linzen et al. (2016)
Acceptability	The boy by the boats [is/*are] smiling.	Syntactic and semantic phenomena	More flexible than LM across architectures; well studied in ling.	Usually requires additional training on top of LM.	Warstadt et al. (2020)
NLI	The boy is smiling. -> The boy [is/*is not] happy.	Semantics/pragmatics/ world knowledge	Flexible, easy to "recast" many tasks to NLI; long history	Often awkward sentences/confounds; low human agreement	White et al. (2017)
Generation	Dante was born in [Mask]	Semantics/pragmatics/ world knowledge	Can be more natural than NLI; incorporates more context	Hard to auto evaluate, esp. beyond one word/factoid questions	Petroni et al. (2019)
MT	The repeated calls from his mother **should** have alerted us. / Les appels rep´ et´ es de sa m ´ ere **devraient** ` nous avoir alertes.	Multilingual morpho-/lexico-/syntax (e.g. cross-lingual agreement)	Only way of specifically probing cross-lingual systems	Often relies on manual eval (though recent approaches use probabilities similar to in LM tasks)	Isabelle et al. (2017)

Experimental Designs

- Tightly Controlled
- Loosely Controlled
- Adversarial Examples

Experimental Designs: Tightly Controlled

Experimental Designs: Tightly Controlled

- Minimal Pairs/Counterfactuals

Experimental Designs: Tightly Controlled

- Minimal Pairs/Counterfactuals

Gender Bias: Rudinger et al. (2018)

(1a) **The paramedic** performed CPR on the passenger even though she/he/they knew it was too late.

(2a) The paramedic performed CPR on **the passenger** even though she/he/they was/were already dead.

Experimental Designs: Tightly Controlled

- Minimal Pairs/Counterfactuals

Gender Bias: Rudinger et al. (2018)

(1a) **The paramedic** performed CPR on the passenger even though she/he/they knew it was too late.

(2a) The paramedic performed CPR on **the passenger** even though she/he/they was/were already dead.

Subj.-Verb Agree.: Marvin and Linzen (2018)

a. The farmer that the parents love swims.
b. *The farmer that the parents love swim.

Experimental Designs: Tightly Controlled

- Minimal Pairs/Counterfactuals

Gender Bias: Rudinger et al. (2018)

(1a) **The paramedic** performed CPR on the passenger even though she/he/they knew it was too late.

(2a) The paramedic performed CPR on **the passenger** even though she/he/they was/were already dead.

Subj.-Verb Agree.: Marvin and Linzen (2018)

a. The farmer that the parents love swims.
b. *The farmer that the parents love swim.

Veridicality: White et al. (2018)

Someone {knew, didn't know} that a particular thing happened.
Someone {was, wasn't} told that a particular thing happened.
Did that thing happen?

Experimental Designs: Tightly Controlled

- Minimal Pairs/Counterfactuals
- Pros: Few confounds, easier to attribute difference to the phenomena itself
- Cons: Can be hard to generate; may not exist in a way that is natural
- Good for phenomena that manifest neatly in the grammar (SV agreement, gender bias), but less so for complex phenomena ("common sense")

Gender Bias: Rudinger et al. (2018)

(1a) **The paramedic** performed CPR on the passenger even though she/he/they knew it was too late.

(2a) The paramedic performed CPR on **the passenger** even though she/he/they was/were already dead.

Subj.-Verb Agree.: Marvin and Linzen (2018)

a. The farmer that the parents love swims.
b. *The farmer that the parents love swim.

Veridicality: White et al. (2018)
Someone {knew, didn't know} that a particular thing happened.
Someone {was, wasn't} told that a particular thing happened.
Did that thing happen?

Experimental Designs: Loosely Controlled

Experimental Designs: Loosely Controlled

- Average over sets with vs. without property of interest

Experimental Designs: Loosely Controlled

- Average over sets with vs. without property of interest

FraCas: Cooper et al. (1996)

Quantifiers	Plurals	Anaphora
Ellipsis	Adjectives	Compartve
Temporal	Verbs	Attitudes

Experimental Designs: Loosely Controlled

- Average over sets with vs. without property of interest

FraCas: Cooper et al. (1996)

Quantifiers	Plurals	Anaphora
Ellipsis	Adjectives	Compartve
Temporal	Verbs	Attitudes

GLUE Diagnostic Set: Wang et al. (2019)

Lexical Semantics	Logic
Predicate-Argument	Common Sense

Experimental Designs: Loosely Controlled

- Average over sets with vs. without property of interest

FraCas: Cooper et al. (1996)

Quantifiers	Plurals	Anaphora
Ellipsis	Adjectives	Compartve
Temporal	Verbs	Attitudes

GLUE Diagnostic Set: Wang et al. (2019)

Lexical Semantics	Logic
Predicate-Argument	Common Sense

Diverse Natural Language Inference Corpus (DNC): Poliak et al. (2018)

Event Factuality | Named Entities | Gendered Anaph. | Lexico-Syntax

Puns | Relation Extract. | Sentiment

Experimental Designs: Loosely Controlled

- Average over sets with vs. without property of interest
- Pros: Can consist of naturalistic data; can generate larger test sets
- Cons: Contain artifacts, harder to attribute differences to target phenomena

FraCas: Cooper et al. (1996)

Quantifiers	Plurals	Anaphora
Ellipsis	Adjectives	Compartve
Temporal	Verbs	Attitudes

GLUE Diagnostic Set: Wang et al. (2019)

Lexical Semantics	Logic
Predicate-Argument	Common Sense

Diverse Natural Language Inference Corpus (DNC): Poliak et al. (2018)

Event Factuality | Named Entities | Gendered Anaph. | Lexico-Syntax

Puns | Relation Extract. | Sentiment

Experimental Designs: Adversarial Examples

Experimental Designs: Adversarial Examples

- Design data sets (usually using minimal pairs or “perturbations”) that specifically emphasize a model’s weaknesses

Experimental Designs: Adversarial Examples

- Design data sets (usually using minimal pairs or "perturbations") that specifically emphasize a model's weaknesses

Jia and Liang (2017)

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

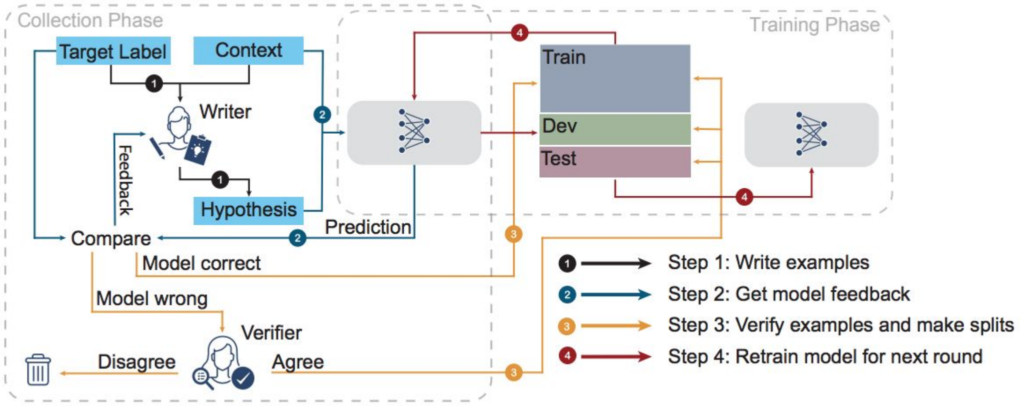
Experimental Designs: Adversarial Examples

- Design data sets (usually using minimal pairs or “perturbations”) that specifically emphasize a model’s weaknesses

Jia and Liang (2017)

Article: Super Bowl 50
Paragraph: “*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*”
Question: “*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*”
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Adversarial NLI: Nie et al. (2019)



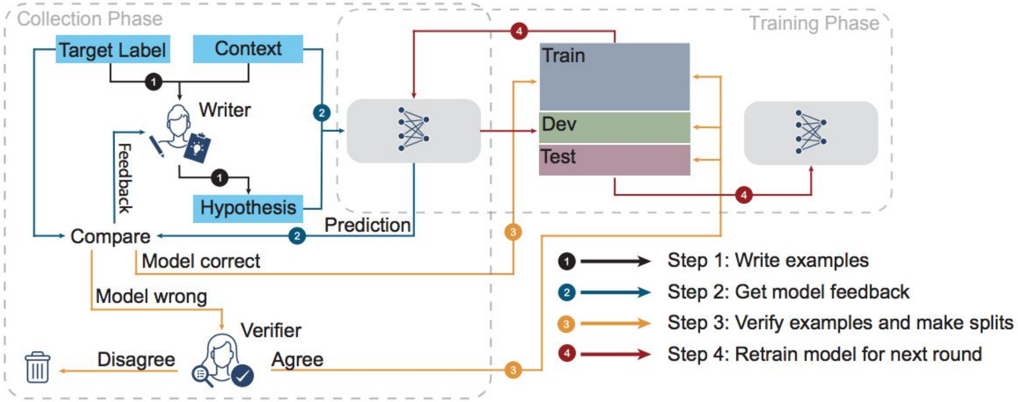
Experimental Designs: Adversarial Examples

- Design data sets (usually using minimal pairs or “perturbations”) that specifically emphasize a model’s weaknesses
- Pros: Practical analysis of failures; can be used as training to improve model
- Cons: Sets age quickly; are model/data specific; “whack-a-mole” approach

[Jia and Liang (2017)](#)

Article: Super Bowl 50
Paragraph: “*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*”
Question: “*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*”
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Adversarial NLI: [Nie et al. (2019)](#)



Construction Methods

- Sources of Data
- Example/Label Generation

Construction Methods

- Sources of Data
  - Sentences drawn from existing corpora
  - Sentences drawn from existing benchmark sets/test suites
  - Templates
  - Manual Generation
- Example/Label Generation

Construction Methods

- Sources of Data
  - Sentences drawn from existing corpora
  - Sentences drawn from existing benchmark sets/test suites
  - Templates
  - Manual Generation
- Example/Label Generation
  - Labels are given by-definition (e.g. if using templates or manual generation)
  - Automatically manipulate sentences and assume heuristic labels (+/- human filtering)
  - Purely automatic (e.g. adversarial)
  - Purely manual labeling (e.g. human generated examples)

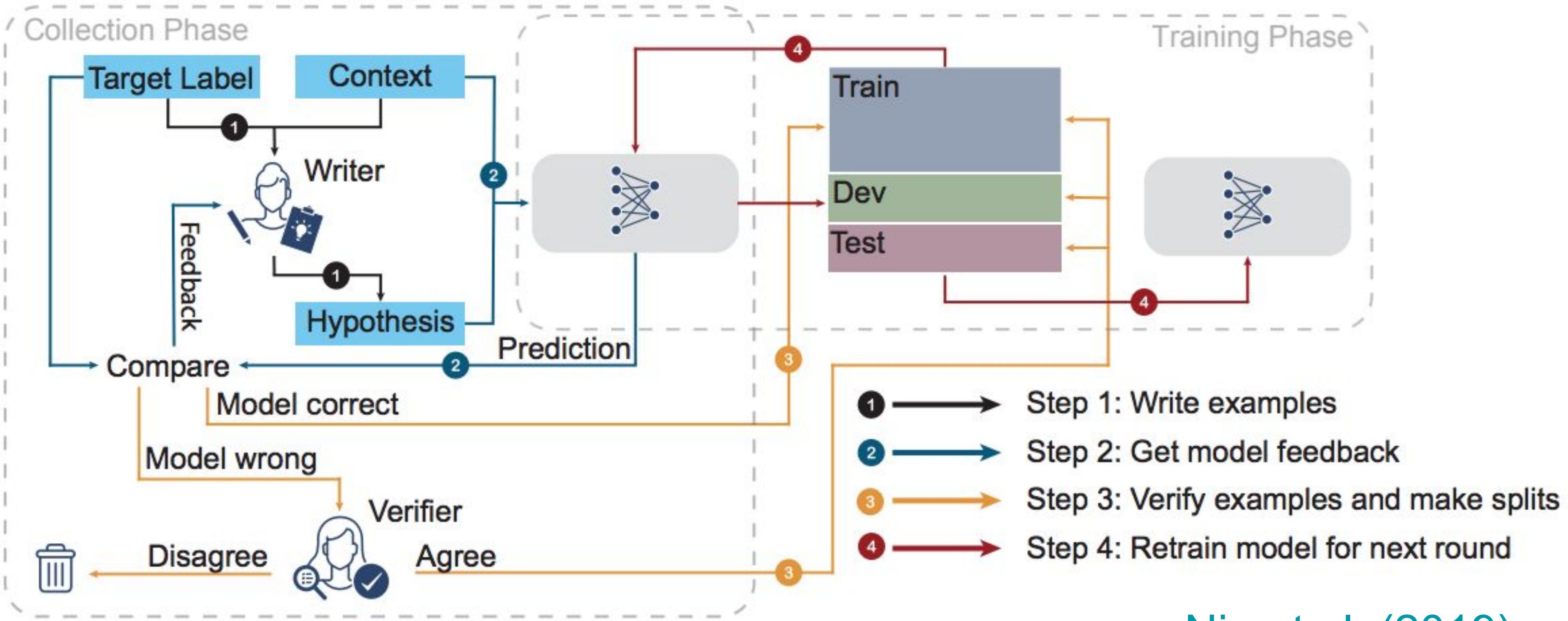
Construction Methods: Entirely Manual

Construction Methods: Entirely Manual

- Examples: Build-It-Break-It, Adversarial NLI

Construction Methods: Entirely Manual

- Examples: Build-It-Break-It, Adversarial NLI



Nie et al. (2019)

Construction Methods: Semi-Automatic

Construction Methods: Semi-Automatic

- Manipulate Existing Corpora, Filter with Crowdsourcing
  - Examples: Ross and Pavlick (2018), Kim et al. (2018), Poliak et al. (2018)

Construction Methods: Semi-Automatic

- Manipulate Existing Corpora, Filter with Crowdsourcing
  - Examples: Ross and Pavlick (2018), Kim et al. (2018), Poliak et al. (2018)

Find sentences in existing corpus containing target phenomenon

Everyone **knows that** the CPI is the most accurate.

I **know that** I was born to succeed

Apply automatic manipulations and assign labels

Everyone **knows that** the CPI is the most accurate. -> The CPI is the most accurate

I **know that** I was born to succeed -> I was born to succeed

Crowdsource to confirm human labels match expected labels

Everyone **knows that** the CPI is the most accurate. -> The CPI is the most accurate

~~I **know that** I was born to succeed -> I was born to succeed~~

Final, vetted corpus

Everyone **knows that** the CPI is the most accurate. -> The CPI is the most accurate

Construction Methods: Semi-Automatic

- Hand-crafted templates that produce known labels
  - Examples: Ettinger et al. (2018), McCoy et al. (2019)

Construction Methods: Semi-Automatic

- Hand-crafted templates that produce known labels
  - Examples: Ettinger et al. (2018), McCoy et al. (2019)

Subcase	Template	Example
Entailment: Conjunctions	The N_1 and the N_2 V the N_3 → The N_2 V the N_3	The actor and the professor mentioned the lawyer. → The professor mentioned the lawyer.
Non-entailment: NP/S	The N_1 V_1 the N_2 V_2 the N_3 ↛ The N_1 V_1 the N_2	The managers heard the secretary encouraged the author. ↛ The managers heard the secretary.

McCoy et al. (2019)

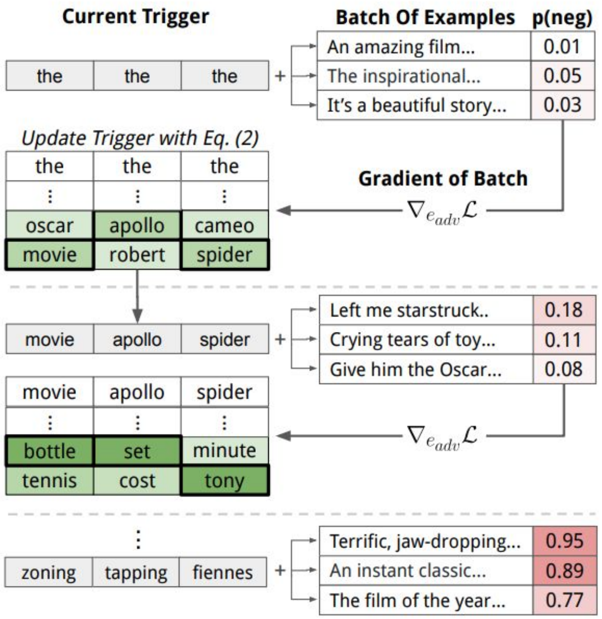
Construction Methods: Fully Automatic

Construction Methods: Fully Automatic

- Examples: Ebrahimi et al. (2018), Wallace et al. (2019)

Construction Methods: Fully Automatic

- Examples: [Ebrahimi et al. (2018)](), [Wallace et al. (2019)]()



[Wallace et al. (2019)]()

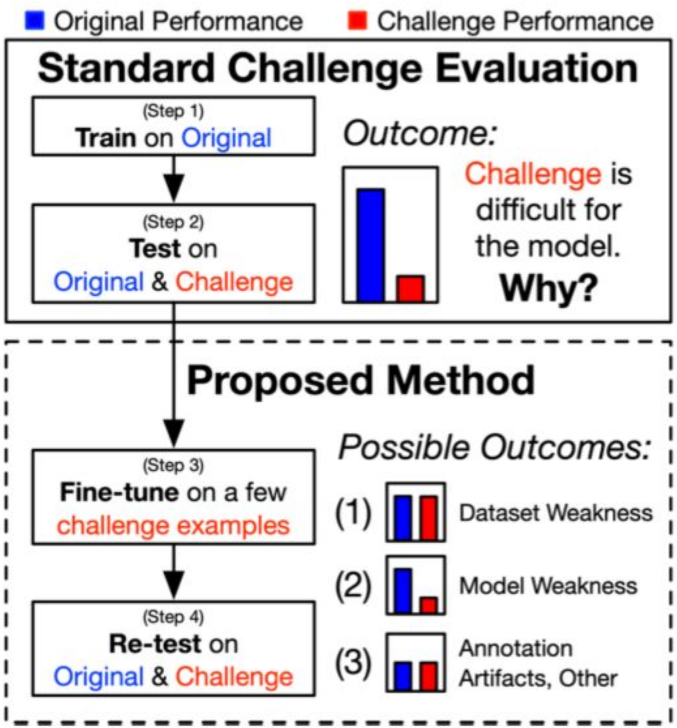
Challenge Sets: Limitations

Challenge Sets: Limitations

- Availability
  - Limited coverage of tasks and languages
  - Need to expand beyond English and to more NLP tasks

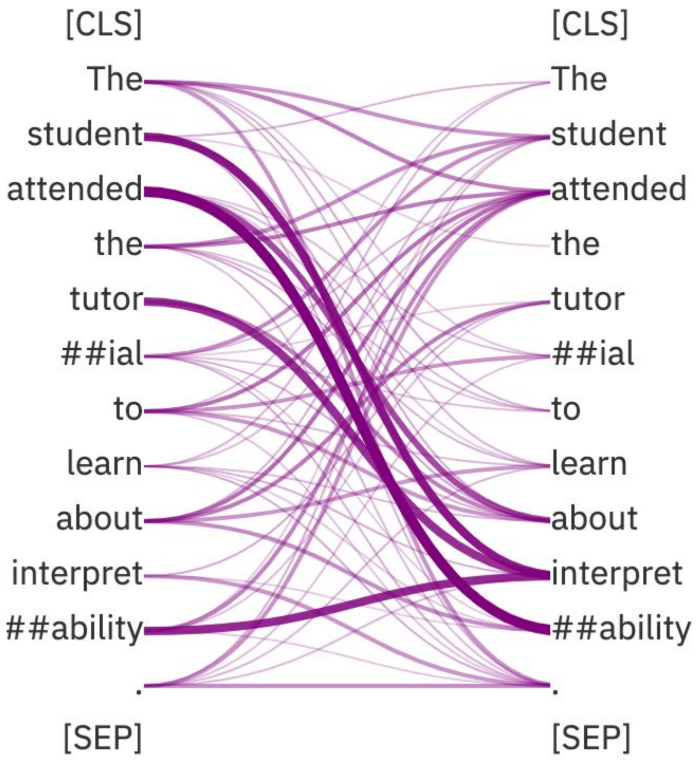
Challenge Sets: Limitations

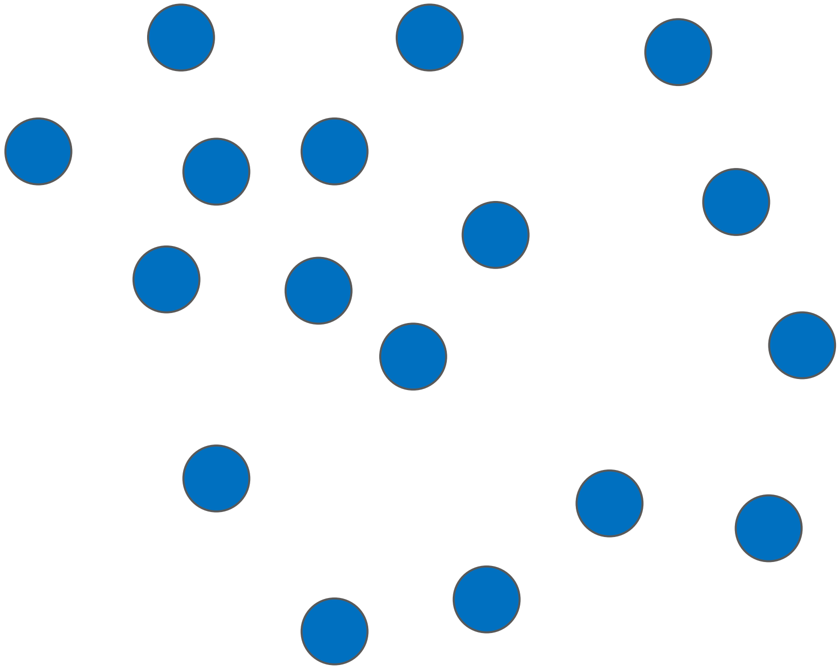
- Availability
  - Limited coverage of tasks and languages
  - Need to expand beyond English and to more NLP tasks
- Methodology
  - What does failure on a challenge set tell us?
  - Who is to blame, the model or its training data?
  - [Lie et al. (2019)] fine-tune a model on a few challenge set examples and re-evaluate
  - [Rozen et al. (2019)] diversify both the training and test data
  - [Geiger et al. (2019)] propose method for determining whether a generalization task is "fair"



Outline

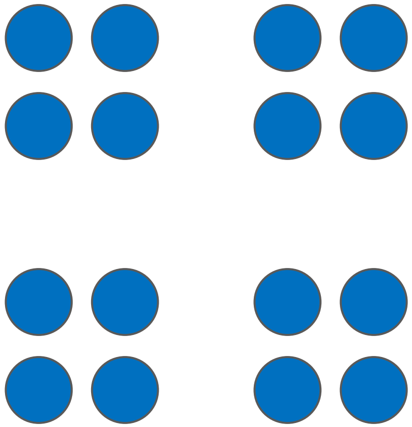
- Structural analyses
- Behavioral analyses
- **Interaction + Visualization**
- Other methods



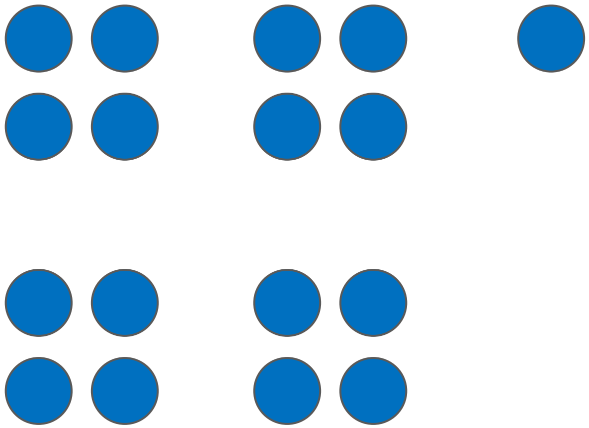


How many circles do you see?

Slide inspired by
Hendrik Strobelt



Visualization can help you understand larger patterns



BUT… Visualization can lie. It was actually 17 🙃

Outline

- Structural analyses
- Behavioral analyses
- **Interactive visualizations**
    - **Why do we want interactive visualizations?**
    - Example: Identifying neuron purpose
    - Categorizing research in visualization
    - Hands-on with a simple attention visualization
    - Future challenges and limitations
- Other methods

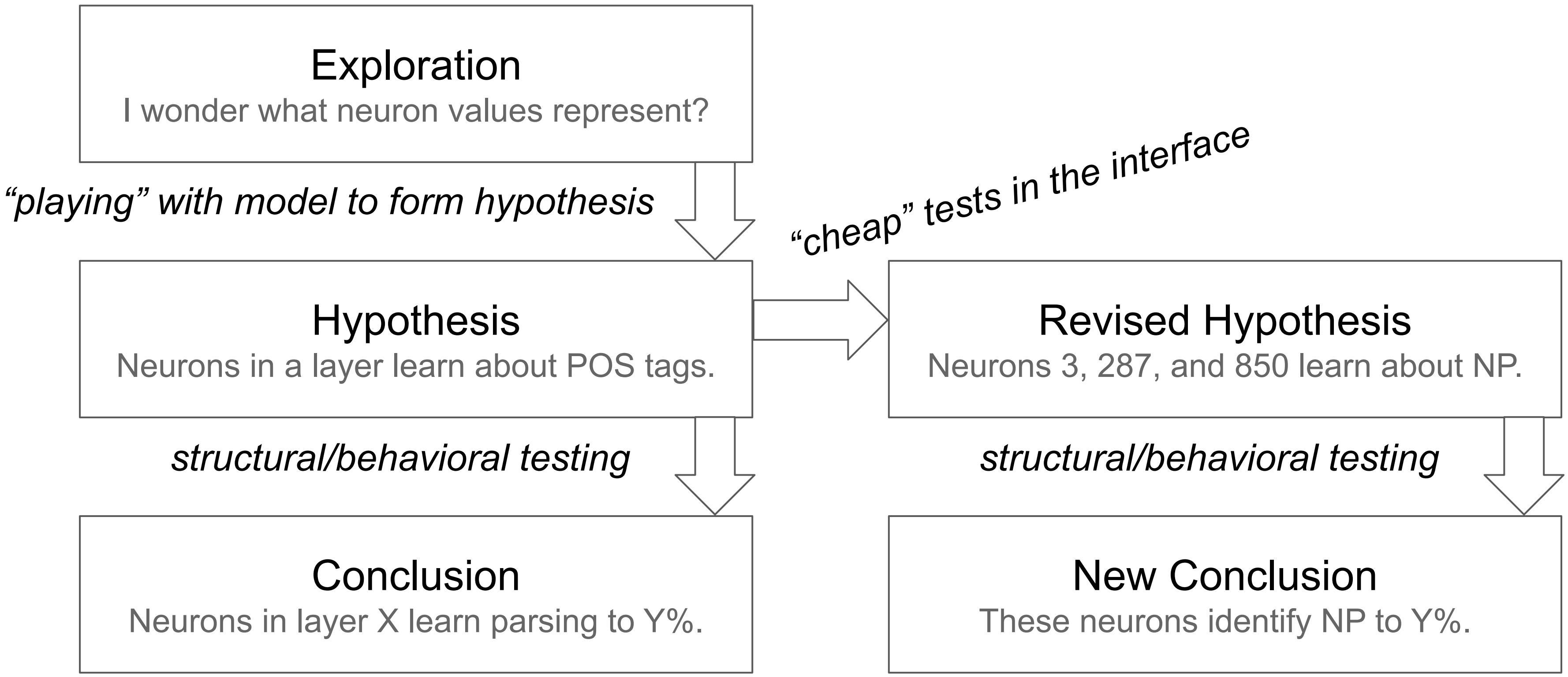
Visual Analytics

“The goal of Visual Analytics is to make our **way of processing data and information** transparent for an analytic discourse.

The visualization of these processes will provide the **means of communicating** about them”

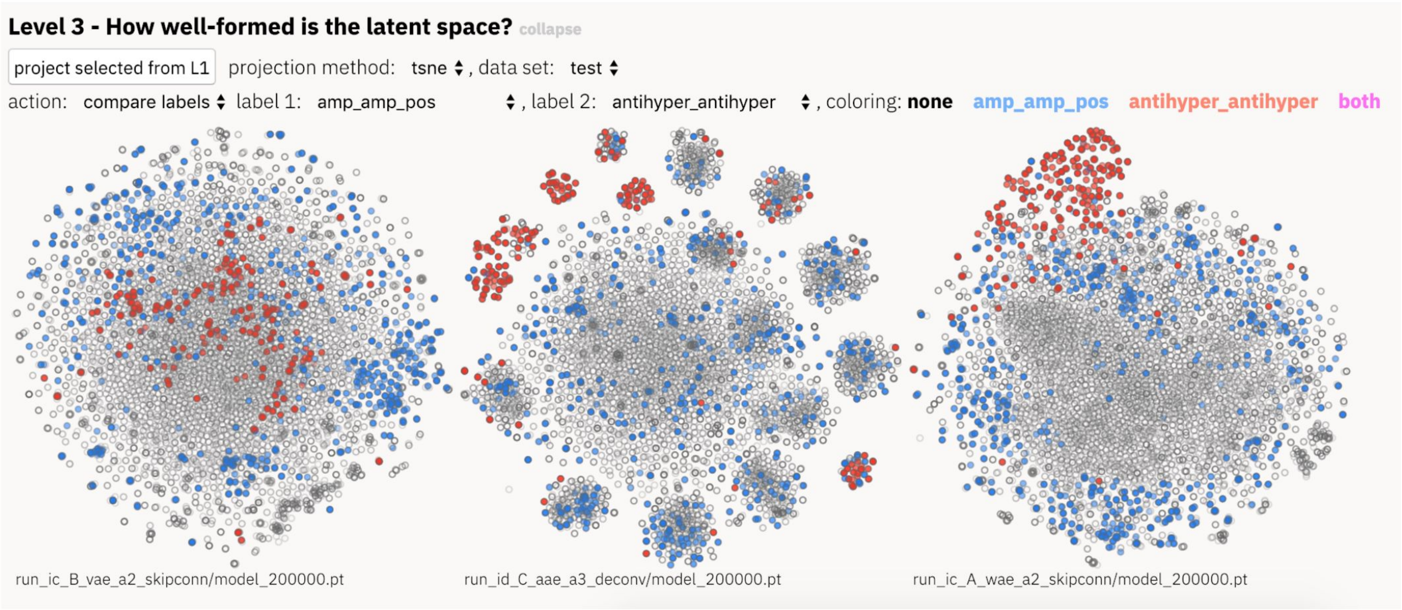
[Keim et al., 2009]

The role of interaction and visualization



Why? - Interactive methods help...

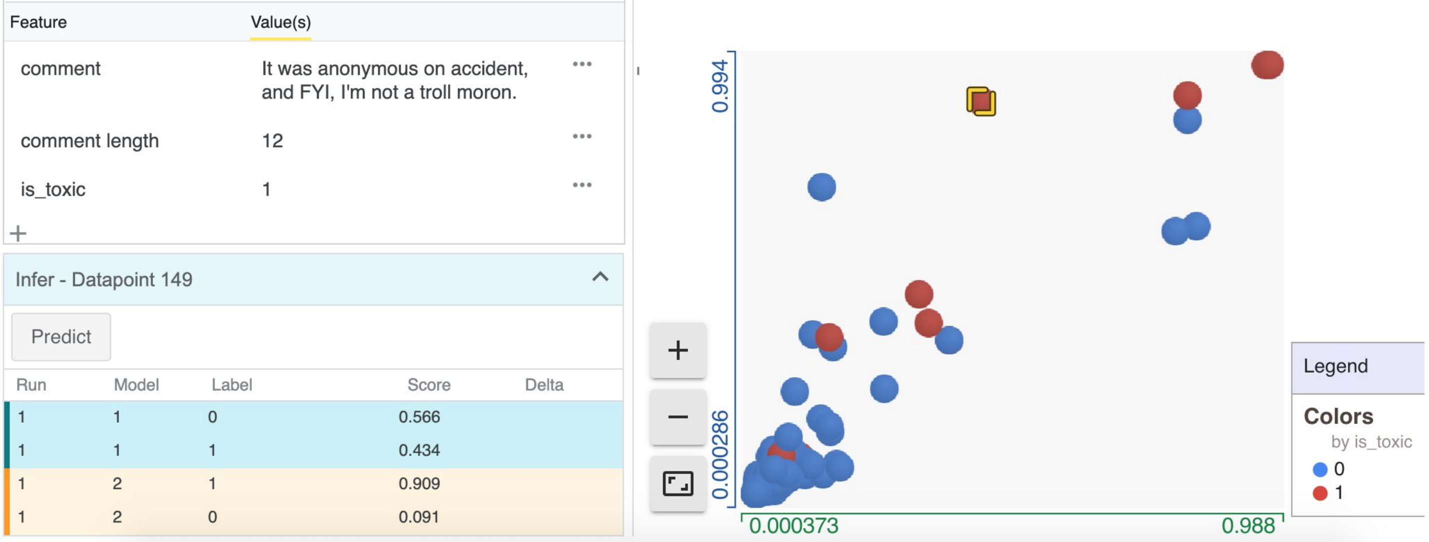
… reduce the exploration space when it is too large for brute-force methods



[Sercu et al., 2019]

Why? - Interactive methods help...

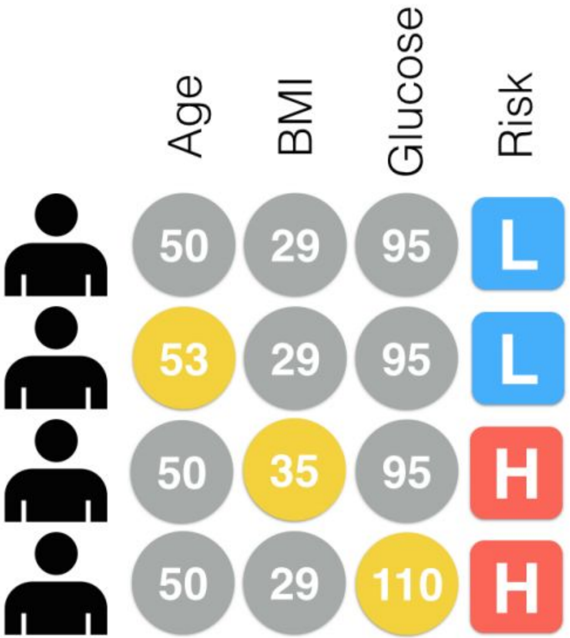
… to generate hypotheses about model behavior or a dataset

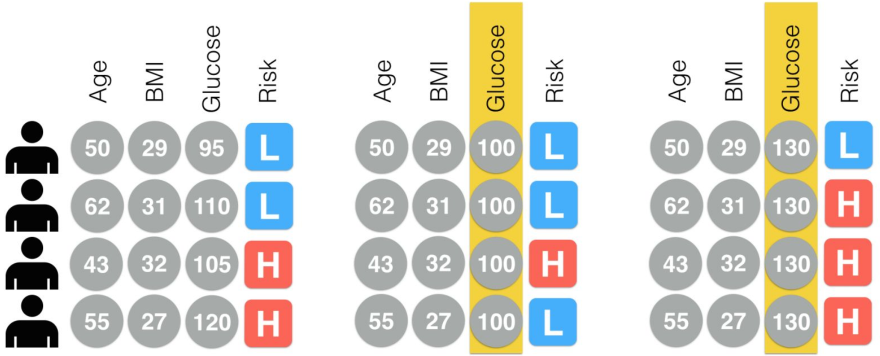


[Wexler et al., 2019]

Why? - Interactive methods help...

… asking counterfactual “what if” question to the model and data





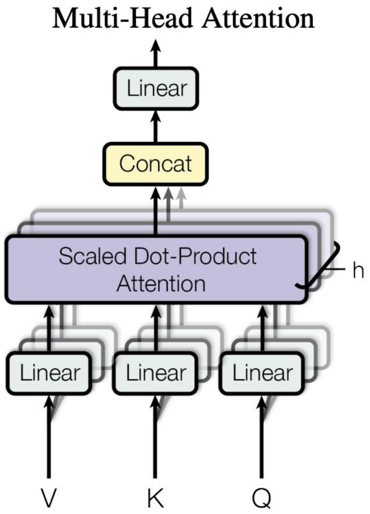
[Krause et al., 2016]

Why? - Interactive methods help...

… understand difficult concepts

A $$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

B



[Vaswani et al., 2017]

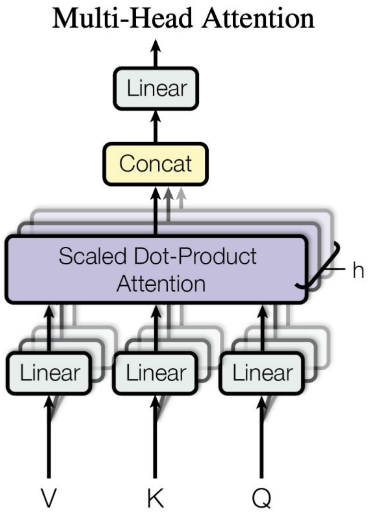
Why? - Interactive methods help...

… understand difficult concepts

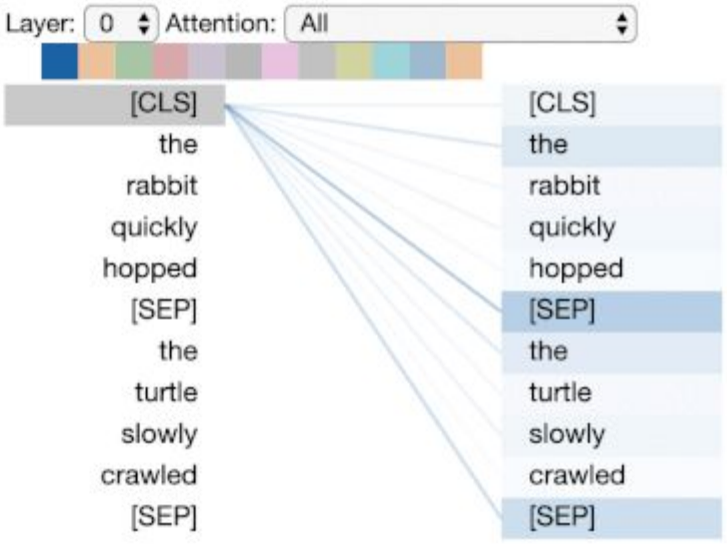
A

$$\mathrm{Attention}(Q,K,V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

B



C



[Vig, 2019]

“A key element of the visualization approach is its ability to generate **trust** in the user. Unlike pure machine learning techniques, in a data visualization the user “sees” the data and information as a part of the analysis.

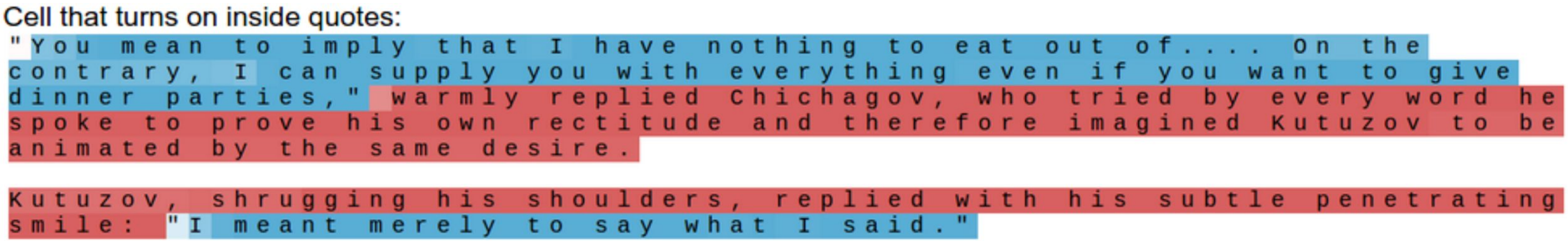
When the visualization is interactive, the user will be part of the loop and involved in driving the visualization. In such a context, the development of a **mental model** goes hand in hand with the visualization.“

[Endert et al., 2018]

Outline

- Structural analyses
- Behavioral analyses
- **Interactive visualizations**
  - Why do we want interactive visualizations?
  - **Example: Identifying neuron purpose**
  - Categorizing research in visualization
  - Hands-on with a simple attention visualization
  - Future challenges and limitations
- Other methods

Motivation: finding neurons with a purpose



Can we do this interactively? Can we do this for groups of neurons?
Exhaustive search is in O(n!).

Visualizing and Understanding Recurrent Networks
[Karpathy, et al.’16]

Interactive Visualization Questionnaire

What is the goal of the tool?

Scientific / Pedagogical / Debugging / Debiasing / **...**

Understanding model structure / model decisions / data / ...

How do you quantify an outcome? **Generated hypotheses about model behavior**

Who is your user?

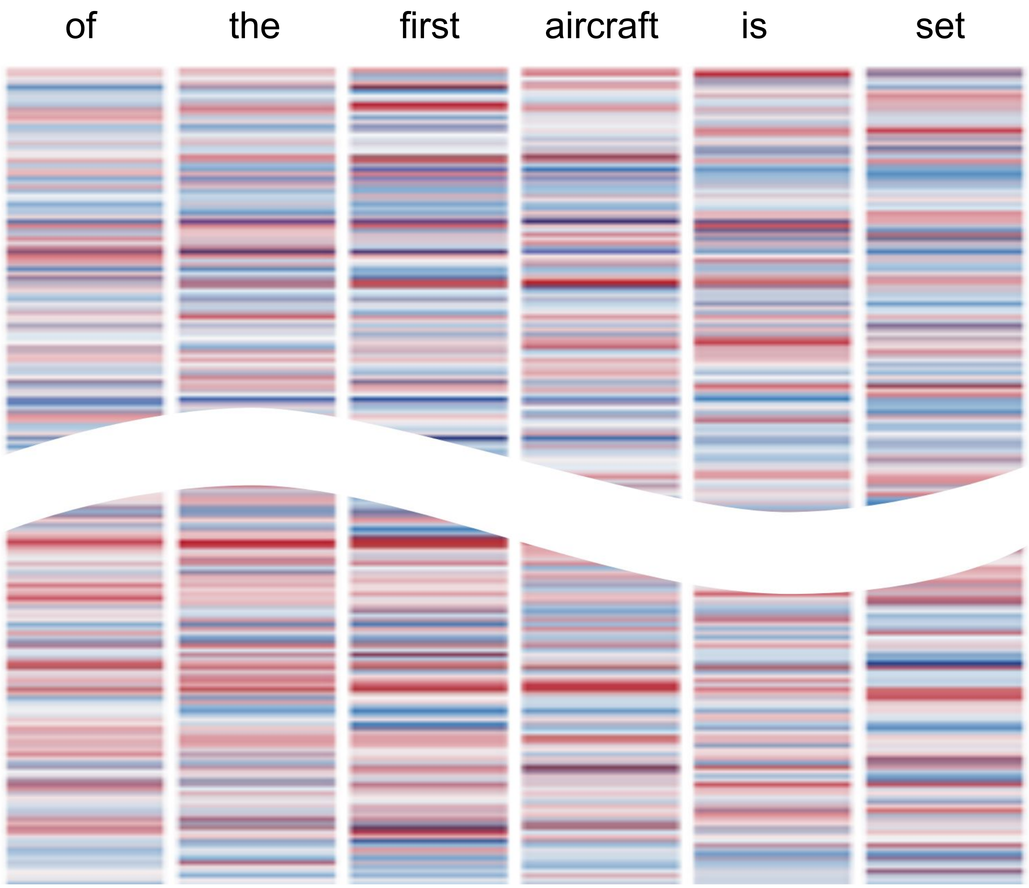
ML or NLP Expert/ Domain Expert / Student / ...

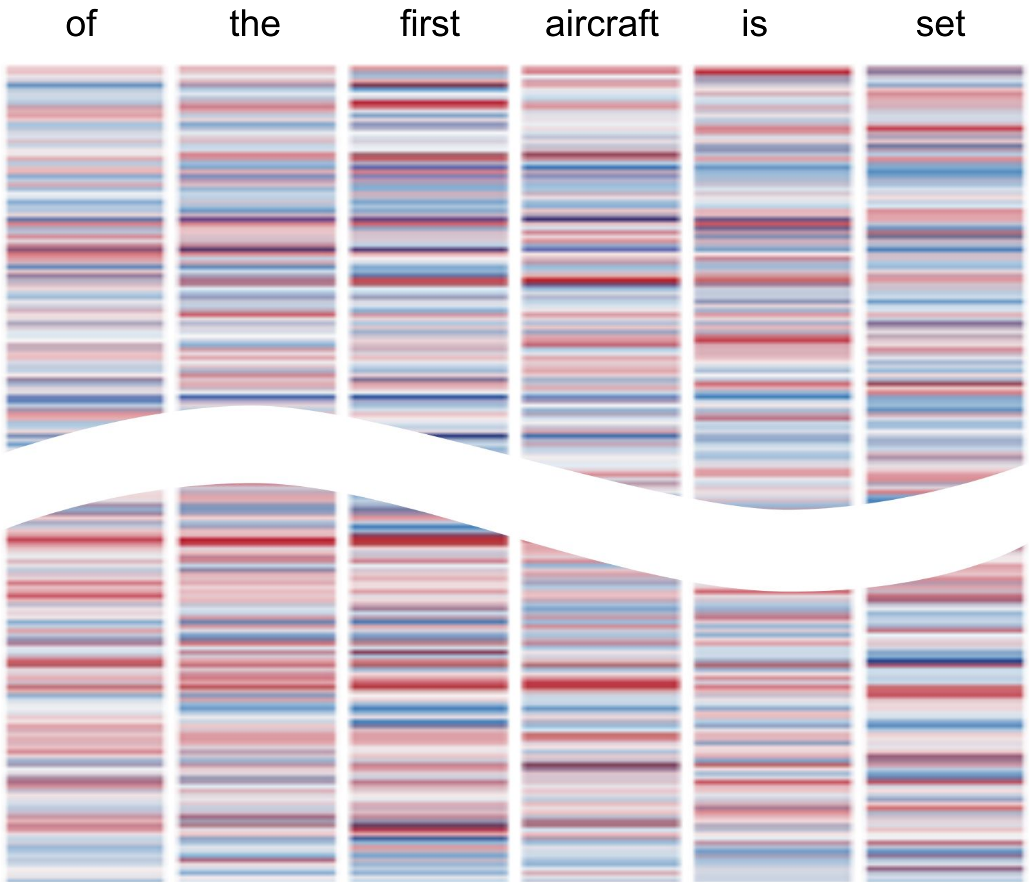
How much domain/ model knowledge do they have? **Enough to understand metadata**

The answers will inform the following implementation questions:

Does the tool require interaction with the model? With the data? **Needs to interact with extracted data**

Can you change the model structure or model decisions? **No**


of
the
first
aircraft
is
set



Issues

Does not scale to large d_{hid}.

Hidden states are position-invariant.

Does not allow investigation of neuron groups.

No filtering.

No tying to meta-data (like POS-tags, nesting, etc.)

Example: finding neurons with a purpose

Consider a text with words $w_1, \ldots, w_n$.

Let $\mathbf{h}_t$ be a hidden state vector with d_{hid} dimensions at timestep *t*.

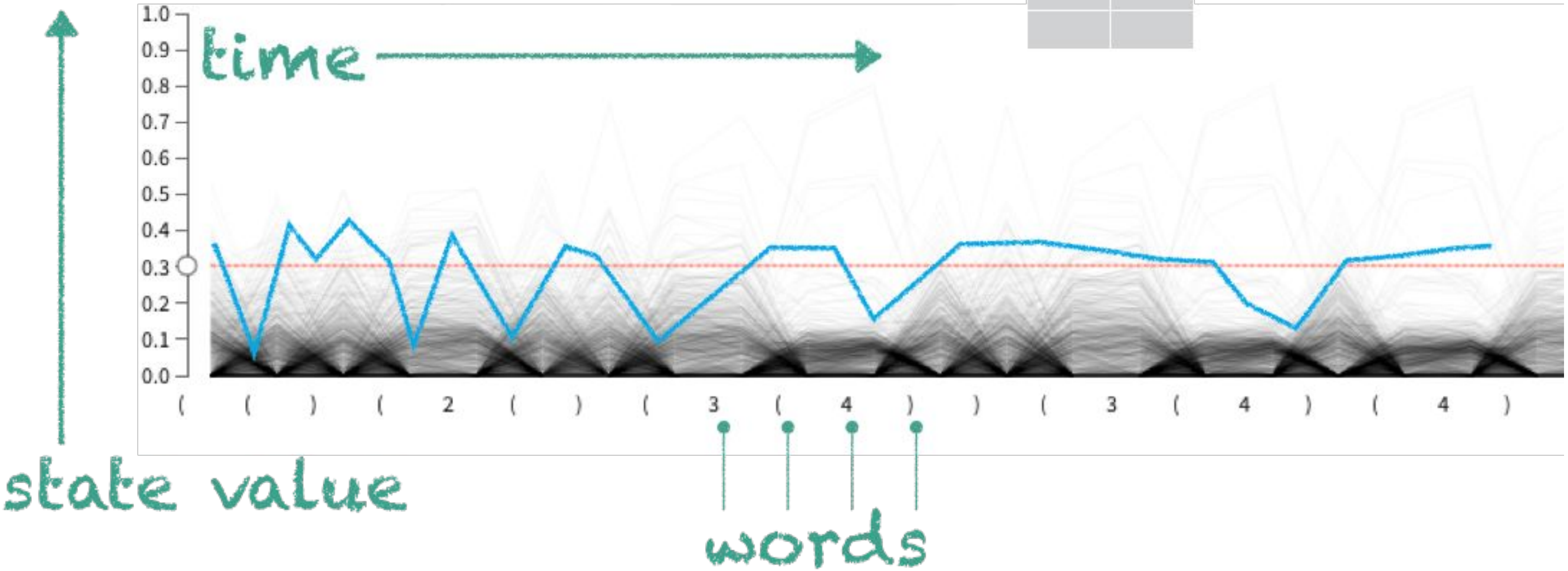
Let *D* be the set of of possible hidden state indices.
A selection $\mathcal{S} \subseteq D$ is a subset of the indices.

For a span *(a,b)* in the text, compute $\mathcal{S}$ as the set of neurons with an activation above a threshold *l:*

$$\mathscr{S}_2 = \{c \in \{1 \ldots D\} : h_{t,c} \geq \ell \text{ for all } a \leq t \leq b\}$$

$h_{i...j}$

h_{t-1}	h_t
0.12	-0.4
0.01	0.07
0.1	0.3
...	



[Strobelt, Gehrmann et al. ‘16]

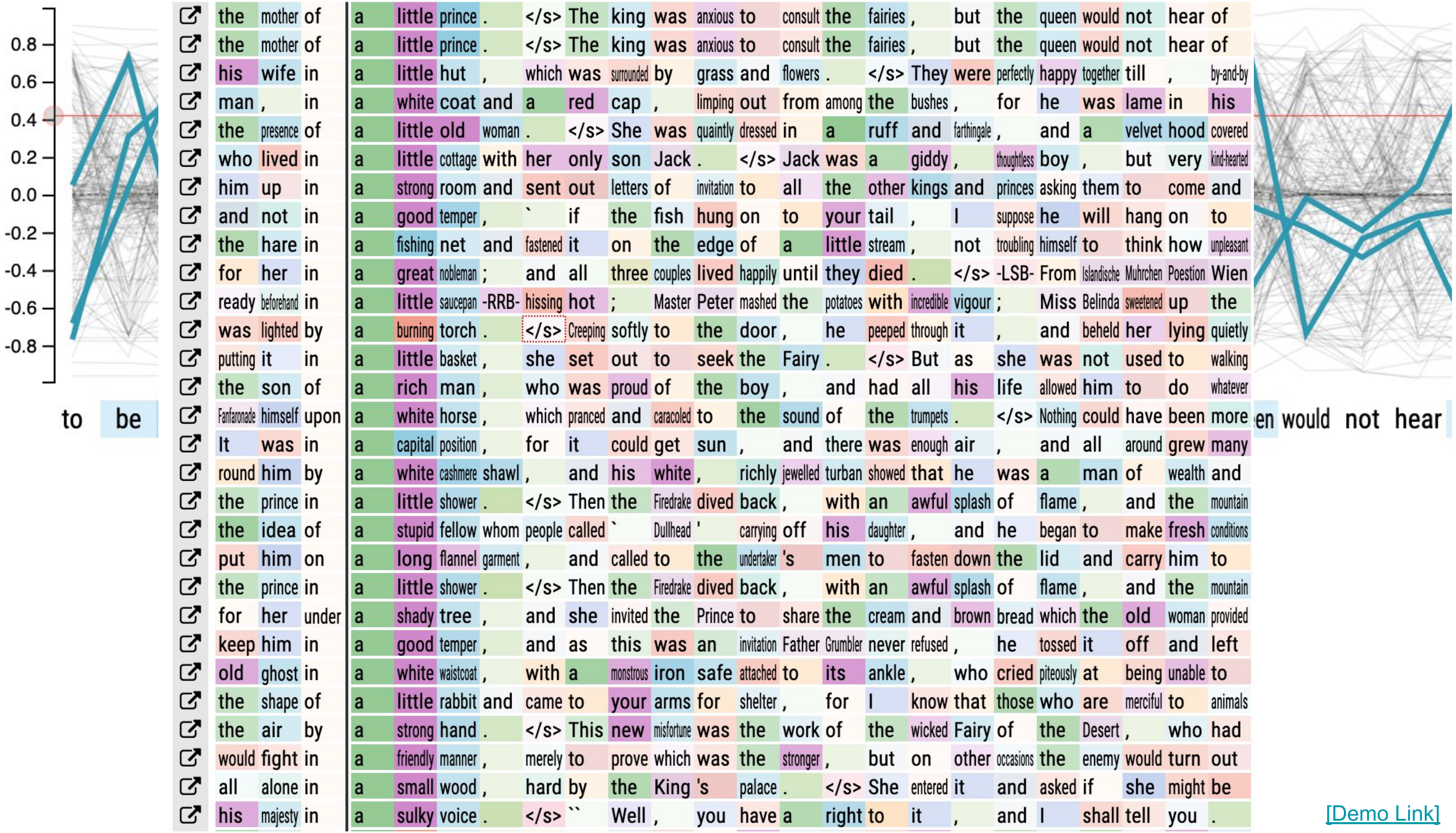
You have a fast selection interface, now what?

Following structural analysis, we could train a probe on only information in $\boldsymbol{\mathcal{S}}$.
But this is costly and thus doesn't allow rapid hypothesis testing.

An interactive system can help by quickly rejecting hypotheses...

21

)	2	)	(	(	)	2	(	)	(	3	)	2	(	)	2	2	)	(	(	3	(	)	)
(	2	)	(	2	2	(	)	2	(	)	(	)	(	3	)	)	(	(	3	)	(	(	4
)	)	)	(	2	2	(	)	(	)	(	)	2	(	3	)	(	)	(	)	(	3	3	(
)	(	)	(	2	(	)	(	)	2	(	)	2	(	(	4	4	)	3	)	2	)	1	)
2	2	)	(	2	(	)	2	2	(	)	2	2	(	3	3	3	)	2	2	(	)	)	1
)	)	1	(	2	(	)	(	3	)	(	3	3	3	(	)	(	4	)	3	3	(	4	4
(	(	)	(	(	)	(	3	)	2	2	2	2	(	)	)	1	(	(	)	)	1	)	0
1	(	)	(	(	3	)	2	(	)	2	2	2	(	(	)	)	(	(	4	)	)	(	)
(	)	)	(	(	3	)	2	2	2	2	2	2	(	)	(	)	(	3	)	)	1	)	0
2	)	1	(	(	)	(	3	3	)	(	)	(	(	4	)	(	4	4	4	)	)	(	(
)	)	1	(	2	2	2	(	)	(	3	)	(	(	)	3	(	)	)	(	(	4	4	4
)	0	(	(	2	2	2	2	2	(	)	(	3	(	)	)	)	1	1	(	2	2	(	)
(	)	1	(	(	3	3	)	2	(	)	(	3	(	4	4	4	4	)	)	2	2	2	)
)	)	)	(	2	(	)	(	3	)	(	)	(	(	4	)	)	)	(	)	)	0	(	)
0	(	1	(	2	2	(	)	(	3	)	(	(	4	4	)	(	4	4	4	)	)	(	)
0	0	(	(	2	2	(	)	2	2	(	)	)	)	0	(	)	(	(	(	)	2	2	(
(	)	1	(	(	)	(	)	2	(	)	(	(	4	)	3	(	4	)	(	)	(	)	(
(	(	)	(	(	)	(	3	)	(	)	2	)	)	0	(	)	(	)	(	)	(	1	)
2	)	1	(	2	(	)	(	)	(	)	(	(	)	(	)	(	)	(	4	)	)	)	)
)	2	)	(	2	2	2	(	)	2	2	(	(	4	)	3	3	(	4	4	4	4	4	4
1	(	)	(	2	(	)	(	)	2	2	(	3	(	4	4	)	)	)	)	0	(	(	2
(	)	1	(	(	)	(	3	)	(	3	3	3	3	)	2	2	)	)	0	0	(	)	0
)	(	)	(	2	(	)	2	2	2	2	(	3	)	)	)	(	1	1	1	(	(	3	(


0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.8
to be
the mother of a little prince . </s> The king was anxious to consult the fairies , but the queen would not hear of
the mother of a little prince . </s> The king was anxious to consult the fairies , but the queen would not hear of
his wife in a little hut , which was surrounded by grass and flowers . </s> They were perfectly happy together till , by-and-by
man , in a white coat and a red cap , limping out from among the bushes , for he was lame in his
the presence of a little old woman . </s> She was quaintly dressed in a ruff and farthingale , and a velvet hood covered
who lived in a little cottage with her only son Jack . </s> Jack was a giddy , thoughtless boy , but very kind-hearted
him up in a strong room and sent out letters of invitation to all the other kings and princes asking them to come and
and not in a good temper , ` if the fish hung on to your tail , I suppose he will hang on to
the hare in a fishing net and fastened it on the edge of a little stream , not troubling himself to think how unpleasant
for her in a great nobleman ; and all three couples lived happily until they died . </s> -LSB- From Islandische Muhrchen Poestion Wien
ready beforehand in a little saucepan -RRB- hissing hot ; Master Peter mashed the potatoes with incredible vigour ; Miss Belinda sweetened up the
was lighted by a burning torch . </s> Creeping softly to the door , he peeped through it , and beheld her lying quietly
putting it in a little basket , she set out to seek the Fairy . </s> But as she was not used to walking
the son of a rich man , who was proud of the boy , and had all his life allowed him to do whatever
Fanfaronade himself upon a white horse , which pranced and caracoled to the sound of the trumpets . </s> Nothing could have been more
It was in a capital position , for it could get sun , and there was enough air , and all around grew many
round him by a white cashmere shawl , and his white , richly jewelled turban showed that he was a man of wealth and
the prince in a little shower . </s> Then the Firedrake dived back , with an awful splash of flame , and the mountain
the idea of a stupid fellow whom people called ` Dullhead ' carrying off his daughter , and he began to make fresh conditions
put him on a long flannel garment , and called to the undertaker 's men to fasten down the lid and carry him to
the prince in a little shower . </s> Then the Firedrake dived back , with an awful splash of flame , and the mountain
for her under a shady tree , and she invited the Prince to share the cream and brown bread which the old woman provided
keep him in a good temper , and as this was an invitation Father Grumbler never refused , he tossed it off and left
old ghost in a white waistcoat , with a monstrous iron safe attached to its ankle , who cried piteously at being unable to
the shape of a little rabbit and came to your arms for shelter , for I know that those who are merciful to animals
the air by a strong hand . </s> This new misfortune was the work of the wicked Fairy of the Desert , who had
would fight in a friendly manner , merely to prove which was the stronger , but on other occasions the enemy would turn out
all alone in a small wood , hard by the King 's palace . </s> She entered it and asked if she might be
his majesty in a sulky voice . </s> `` Well , you have a right to it , and I shall tell you .
en would not hear

[Demo Link]

Outline

- Structural analyses
- Behavioral analyses
- **Interactive visualizations**
  - Why do we want interactive visualizations?
  - Example: Identifying neuron purpose
  - **Categorizing research in visualization**
  - Hands-on with a simple attention visualization
  - Future challenges and limitations
- Other methods

User+Task analysis

Understand - Diagnose - Refine

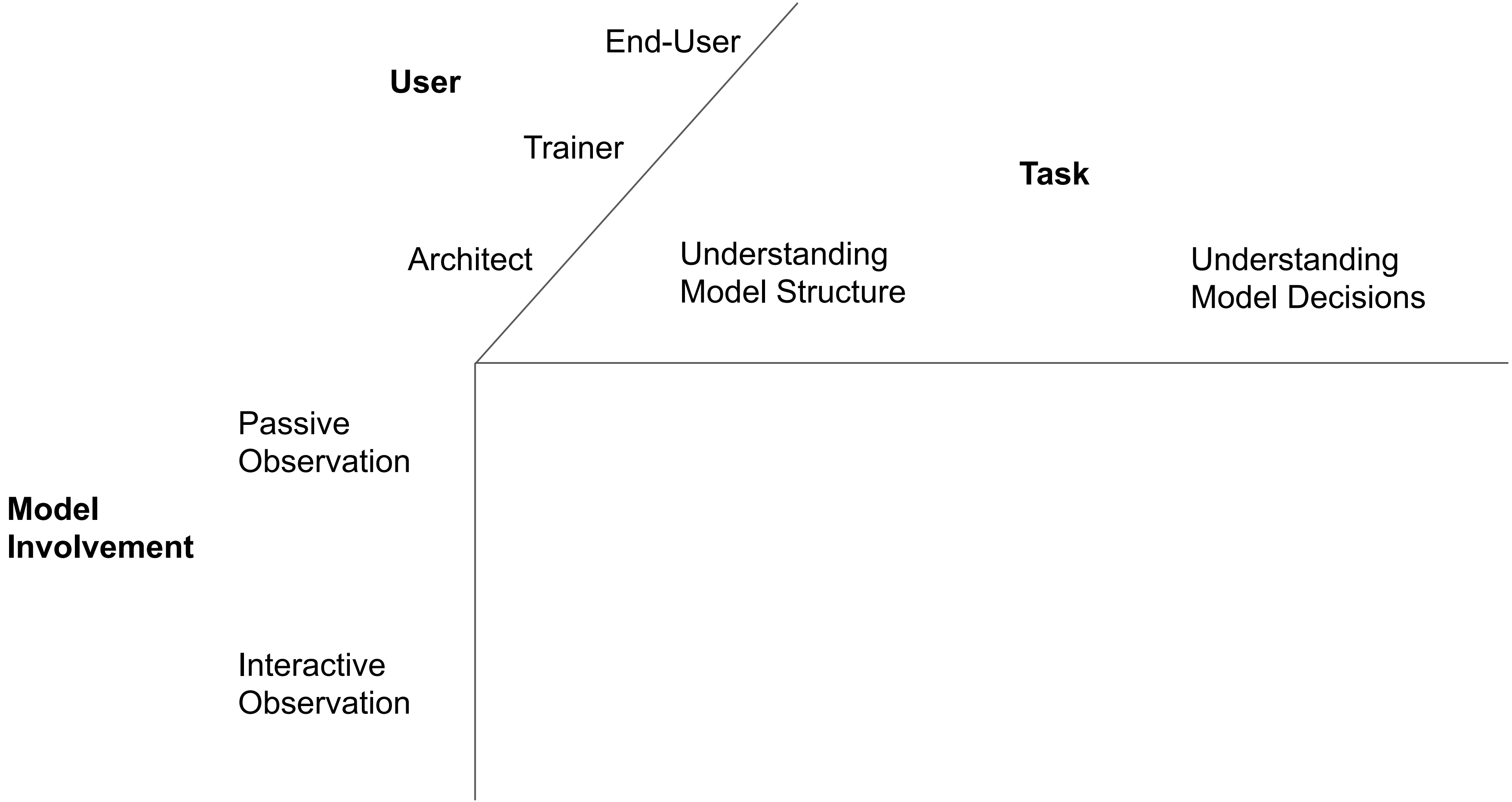
Towards better analysis of machine learning models:
A visual analytics perspective.
[Liu et al.'17]

Architect - Trainer - End-User

LSTMVis: A Tool for Visual Analysis of Hidden State Dynamics in Recurrent Neural Networks
[Strobelt, Gehrmann, et al. '16]

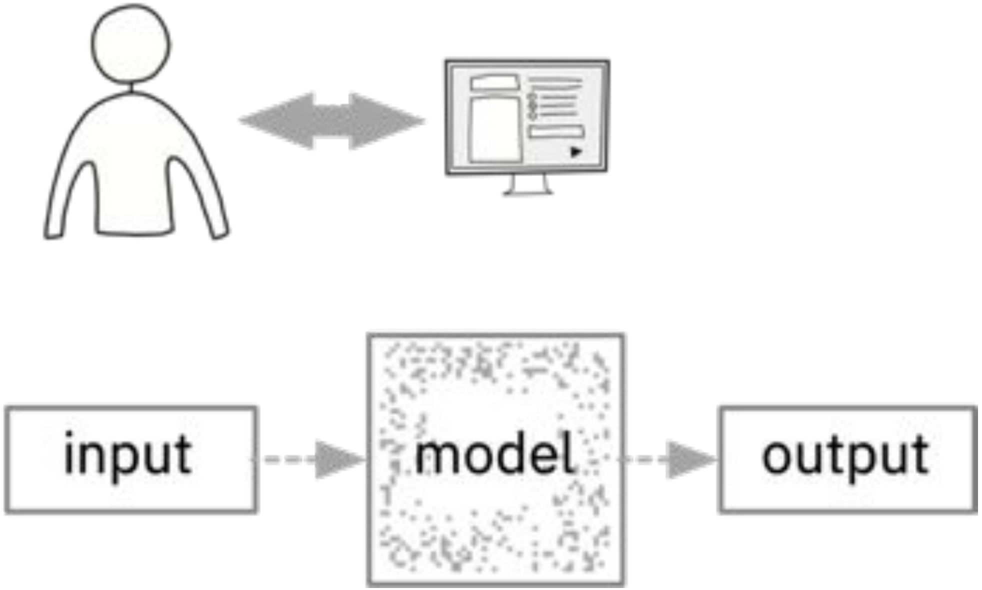
4.1	Interpretability & Explainability	WHY
4.2	Debugging & Improving Models	
4.3	Comparing & Selecting Models	
4.4	Teaching Deep Learning Concepts	
5.1	Model Developers & Builders	WHO
5.2	Model Users	
5.3	Non-experts	
6.1	Computational Graph & Network Architecture	WHAT
6.2	Learned Model Parameters	
6.3	Individual Computational Units	
6.4	Neurons in High-dimensional Space	
6.5	Aggregated Information	

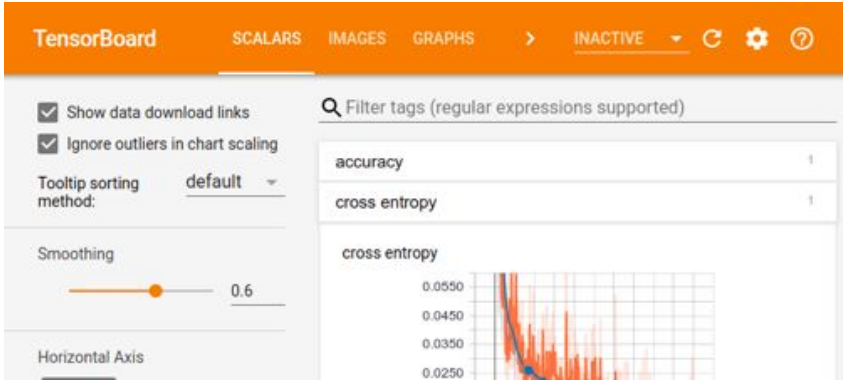
Visual analytics in deep learning:
An interrogative survey for the next frontiers.
[Hohman et al. '18]



[Gehrmann, Strobelt, et al.'19]

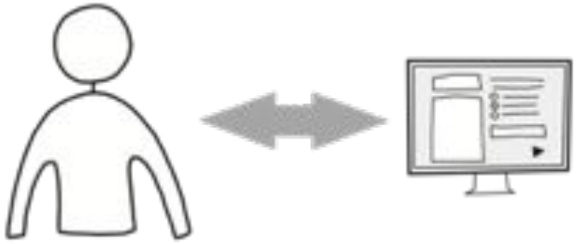
Examples: **Passive Observation**

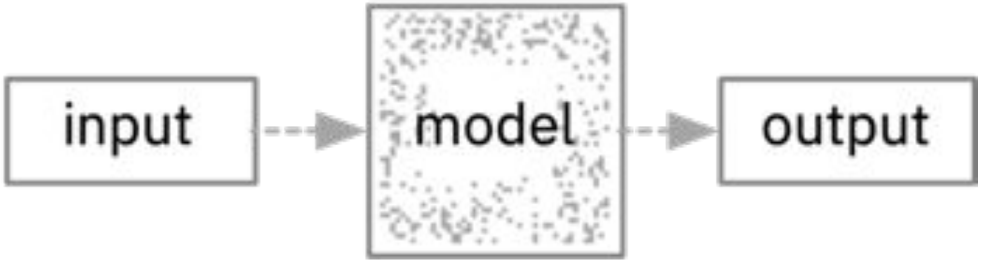




The previous two parts of this tutorial

Examples: **Passive Observation**





Understanding Model Structure

Exploring Neural Networks with Activation Atlases
[Carter, et al.'19]

Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow
[Wongsuphasawat et al. '18]

Understanding Model Decisions

"Why Should I Trust You?" Explaining the Predictions of Any Classifier
[Ribeiro et al. '16]

Rationalizing Neural Predictions
[Lei et al. '16]

Tools:
Captum
AllenNLP Interpret

Examples: **Passive Observation**

EMNLP 2020 Tutorial: Interpreting Predictions of NLP Models
Eric Wallace, Matt Gardner and Sameer Singh

Understanding Model Structure

Exploring Neural Networks with Activation Atlases
[Carter, et al.'19]

Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow
[Wongsuphasawat et al. '18]

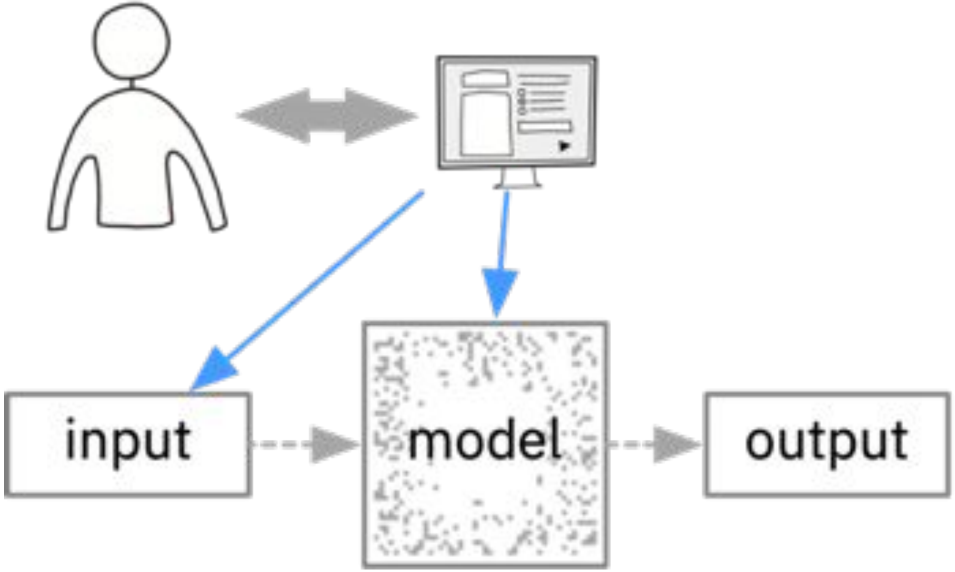
Understanding Model Decisions

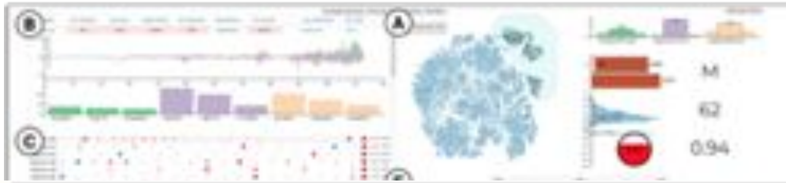
"Why Should I Trust You?" Explaining the Predictions of Any Classifier
[Ribeiro et al. '16]

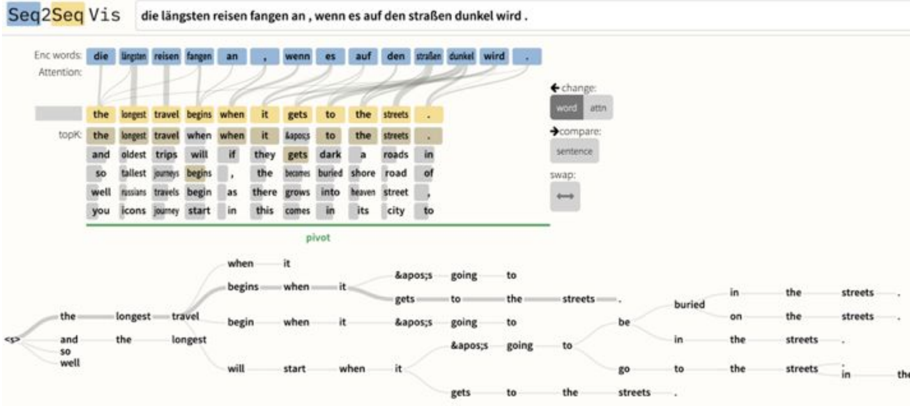
Rationalizing Neural Predictions
[Lei et al. '16]

Tools:
Captum
AllenNLP Interpret

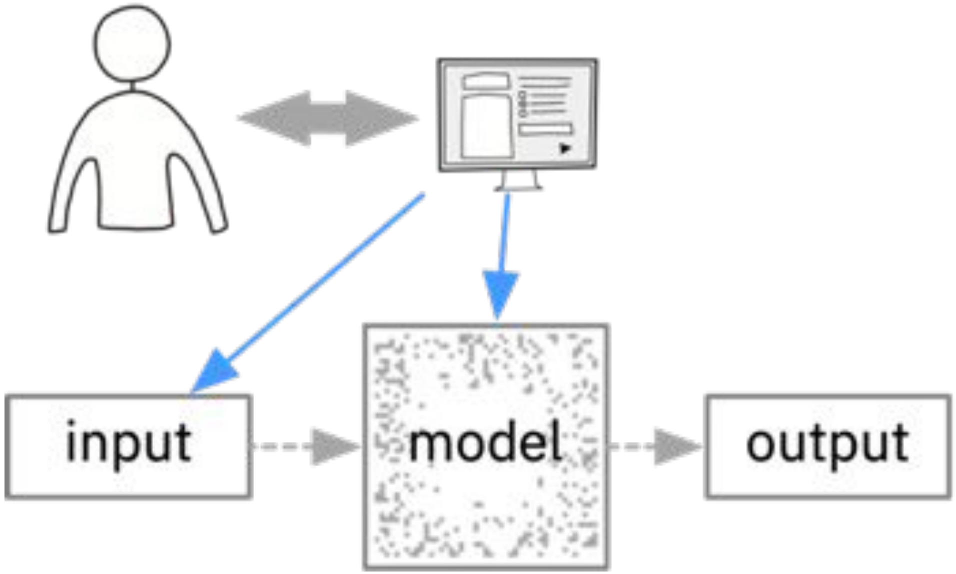
Examples: **Interactive Observation**







Examples: **Interactive Observation**



Understanding Model Structure

LSTMVis: A tool for visual analysis of hidden state dynamics in recurrent neural networks
[Strobelt, Gehrmann, et al.'16]

Understanding Hidden Memories of Recurrent Neural Networks
[Ming et al. '17]

Understanding Model Decisions

RNNbow: Visualizing Learning via Backpropagation Gradients in Recurrent Neural Networks
[Cashman et al. '18]

A Workflow for Visual Diagnostics of Binary Classifiers using Instance-Level Explanations
[Krause et al. '17]

UX and **Evaluation** of Interaction and Visualization

Guidelines for Human-AI Interaction
[Amershi et al. ‘19]

Machine Learning as a UX Design Material:
How Can We Imagine Beyond Automation,
Recommenders, and Reminders?
[Yang et al. ‘18]

Agency plus automation:
Designing artificial intelligence into interactive systems
[Heer, ‘19]

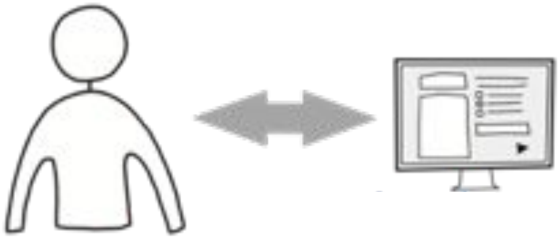
Beyond Accuracy: The Role of Mental Models in
Human-AI Team Performance
[Bansal et al. 19]

Human Evaluation of Models Built for Interpretability
[Lage et al., ‘19]

Proxy Tasks and Subjective Measures Can Be
Misleading in Evaluating Explainable AI Systems
[Buçinca et al. ‘20]

Principles of Explanatory Debugging to Personalize
Interactive Machine Learning
[Kulesza et al. ‘15]

On Human Predictions with Explanations and
Predictions of Machine Learning Models:
A Case Study on Deception Detection
[Lai et al. 19]



Many more in Proceedings of *IEEE Vis*, *CHI*, FAccT, IUI, and the *VisXAI* Workshop!

Outline

- Structural analyses
- Behavioral analyses
- **Interactive visualizations**
  - Why do we want interactive visualizations?
  - Example: Identifying neuron purpose
  - Categorizing research in visualization
  - **Hands-on with a simple attention visualization**
  - Future challenges and limitations
- Other methods

Hands-on: developing an attention visualization

Interactive Visualization Questionnaire

What is the goal of the tool?

Scientific / **Pedagogical** / Debugging / Debiasing / **...**

Understanding model structure / **model decisions** / data / ...

How do you quantify an outcome? **Better understanding of self-attention**

Who is your user?

ML or NLP Expert/ Domain Expert / **Student** / ...

How much domain/ model knowledge do they have? **Very limited**

The answers will inform the following implementation questions:

Does the tool require interaction with the model? With the data? **Needs to extract attention at inference-time**

Can you change the model structure or model decisions? **No**

The 1-day JS Prototype

checkout github: http://bit.ly/SIDN-AttnVis

```
git clone https://github.com/SIDN-IAP/attnvis.git
cd attnvis
```

install dependencies:

```
conda env create -f environment.yml
```

get server to start without errors

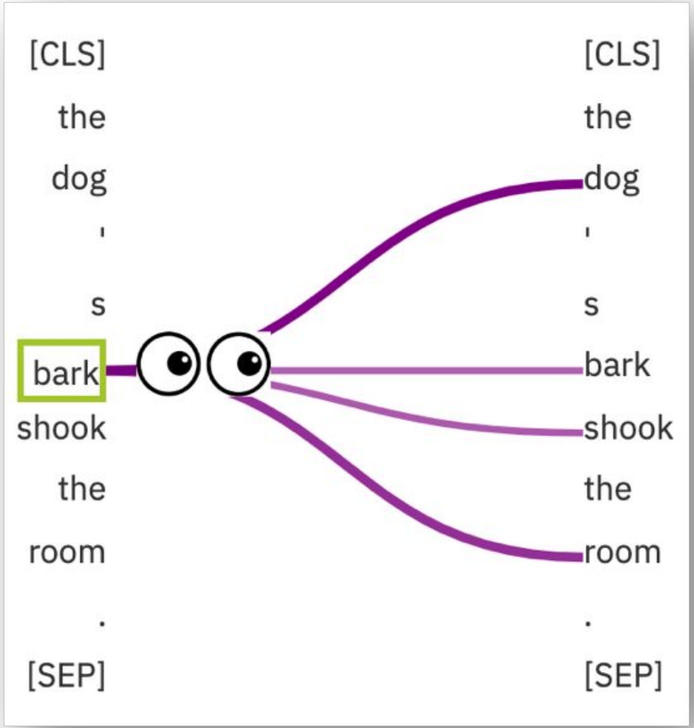
```
conda activate attnvis
python server.py
```

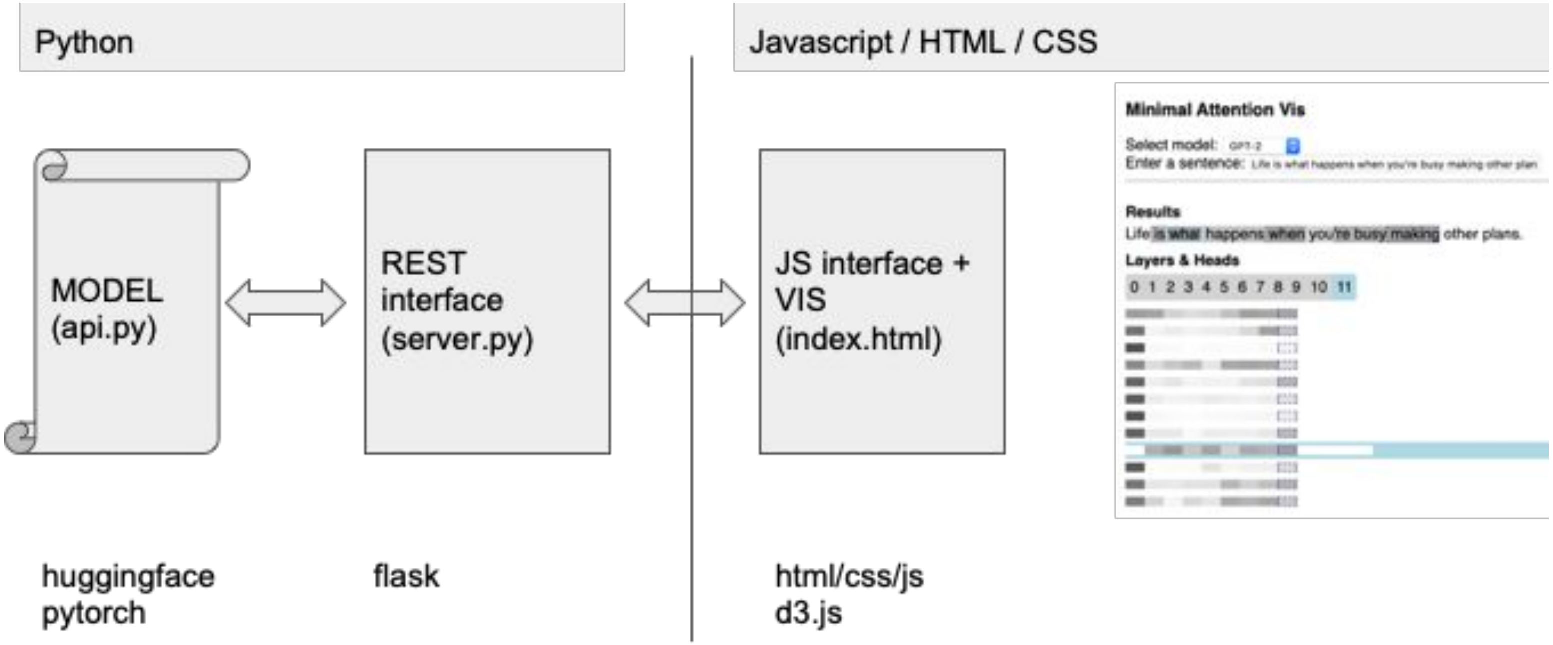
Challenges compared to seq2seq attention

Filtering: We now have 100+ heads

Aggregation: How do we combine multiple attentions?

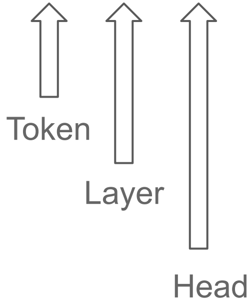
Key/Value/Query: What do we do with that?



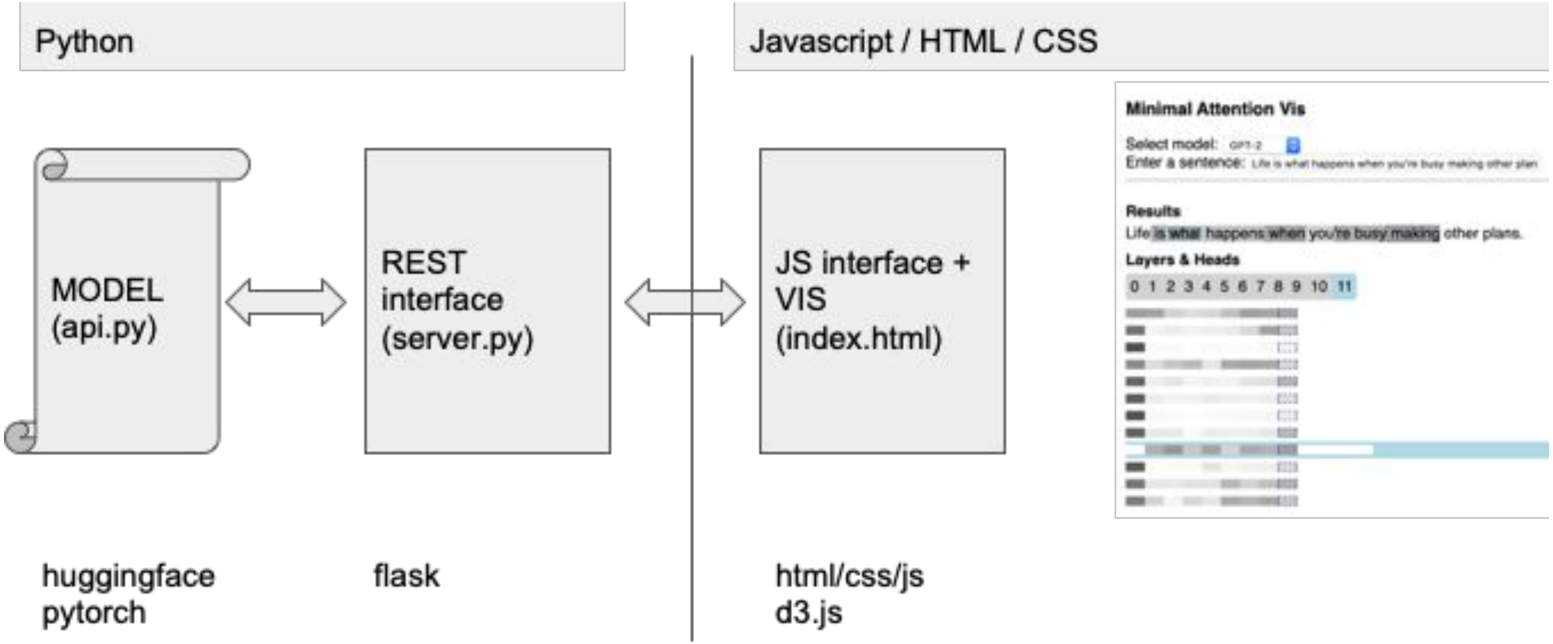

Python
Javascript / HTML / CSS
MODEL
(api.py)
REST
interface
(server.py)
JS interface +
VIS
(index.html)
Minimal Attention Vis
Select model:
Enter a sentence:
Results
Life is what happens when you're busy making other plans.
Layers & Heads
0 1 2 3 4 5 6 7 8 9 10 11
huggingface
pytorch
flask
html/css/js
d3.js

Step 1 Agree on an API between backend and visualization

```
{
  "tokens": List[unicode string],
  "attention": List[List[List[float32]]]
}
```



Note: this API does not support batching!


Python
Javascript / HTML / CSS
MODEL
(api.py)
REST
interface
(server.py)
JS interface +
VIS
(index.html)
Minimal Attention Vis
Select model:
Enter a sentence:
Results
Life is what happens when you're busy making other plans.
Layers & Heads
0 1 2 3 4 5 6 7 8 9 10 11
huggingface
pytorch
flask
html/css/js
d3.js

```python
import torch
from transformers import AutoTokenizer, AutoModel

class AttentionGetter:
    '''Wrapper Class to store model object.'''
    def __init__(self, model_name: str):
        super().__init__()
        self.device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
        self.model = AutoModel.from_pretrained(model_name, output_attentions=True).to(
            self.device
        )
        self.tokenizer = AutoTokenizer.from_pretrained(model_name)
```

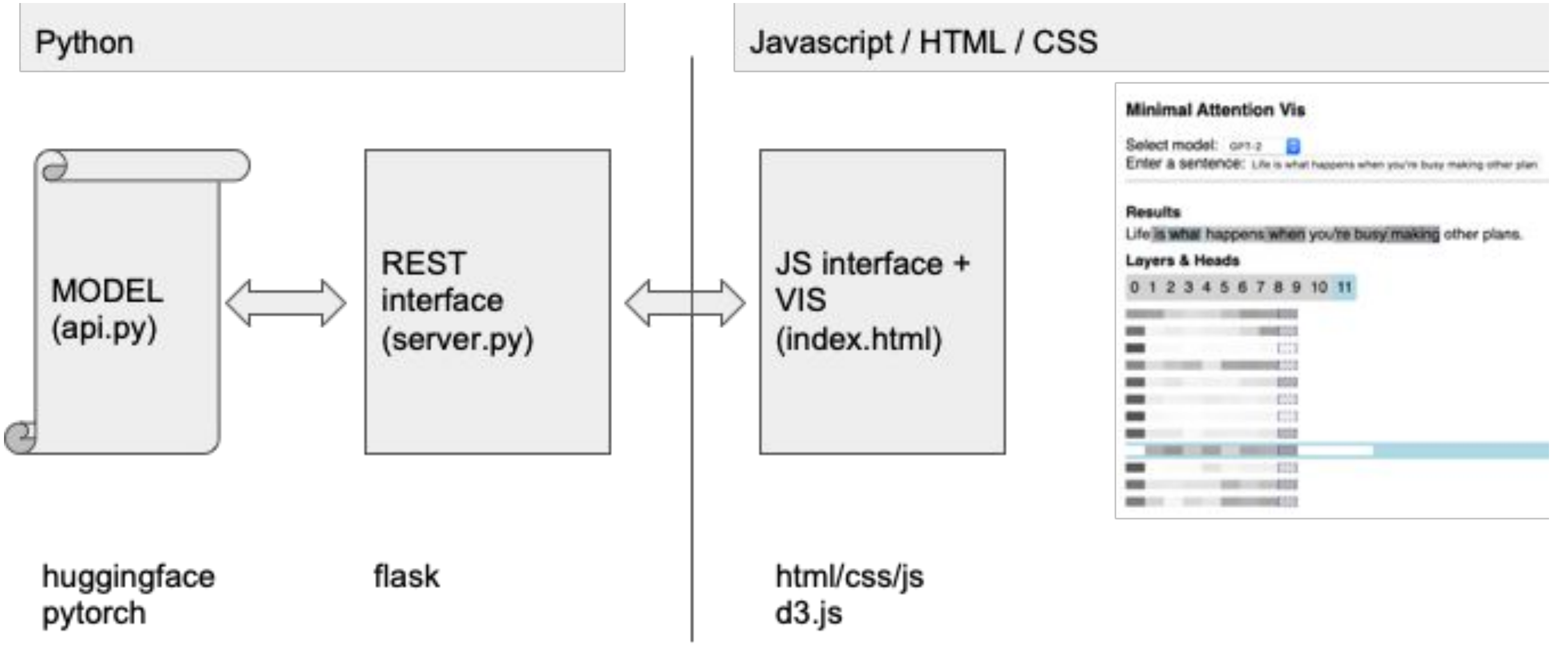
api.py

```python
def bert_analyze_text(self, text: str):
    """Works for BERT Style models"""
    # Tokenize input.
    toked = self.tokenizer.encode(text)
    # Build tensor and unsqueeze batch dimension.
    context = torch.tensor(toked).unsqueeze(0).long()
    # Extract attention.
    attn = self._grab_attn(context)
    # Build payload.
    return {
        "tokens": self.tokenizer.convert_ids_to_tokens(toked),
        "attention": attn,
    }
```

api.py

```python
def _grab_attn(self, context):
    '''
    function to get the attention for a model.
    First runs a forward pass and then extracts and formats attn.
    '''
    output = self.model(context)
    # Grab the attention from the output tuple.
    # Format as Layer x Head x From/To
    attn = torch.cat([l for l in output[-1]], dim=0)
    format_attn = [
        [
            [[str(round(att * 100)) for att in head] for head in layer]
            for layer in tok
        ]
        for tok in attn.cpu().tolist()
    ]
    return format_attn
```

api.py


Python
Javascript / HTML / CSS
MODEL
(api.py)
REST
interface
(server.py)
JS interface +
VIS
(index.html)
Minimal Attention Vis
Select model:
Enter a sentence:
Results
Life is what happens when you're busy making other plans.
Layers & Heads
0 1 2 3 4 5 6 7 8 9 10 11
huggingface
pytorch
flask
html/css/js
d3.js

```python
import json
import os

from flask import Flask as Flask,
from flask import request, redirect

from api import AttentionGetter

app = Flask(__name__)

# Set up cache for model wrappers.
loaded_models = {}

# redirect requests from root to index.html
@app.route('/')
def hello_world():
    return redirect('client/index.html')

if __name__ == '__main__':
    app.run()
```

server.py

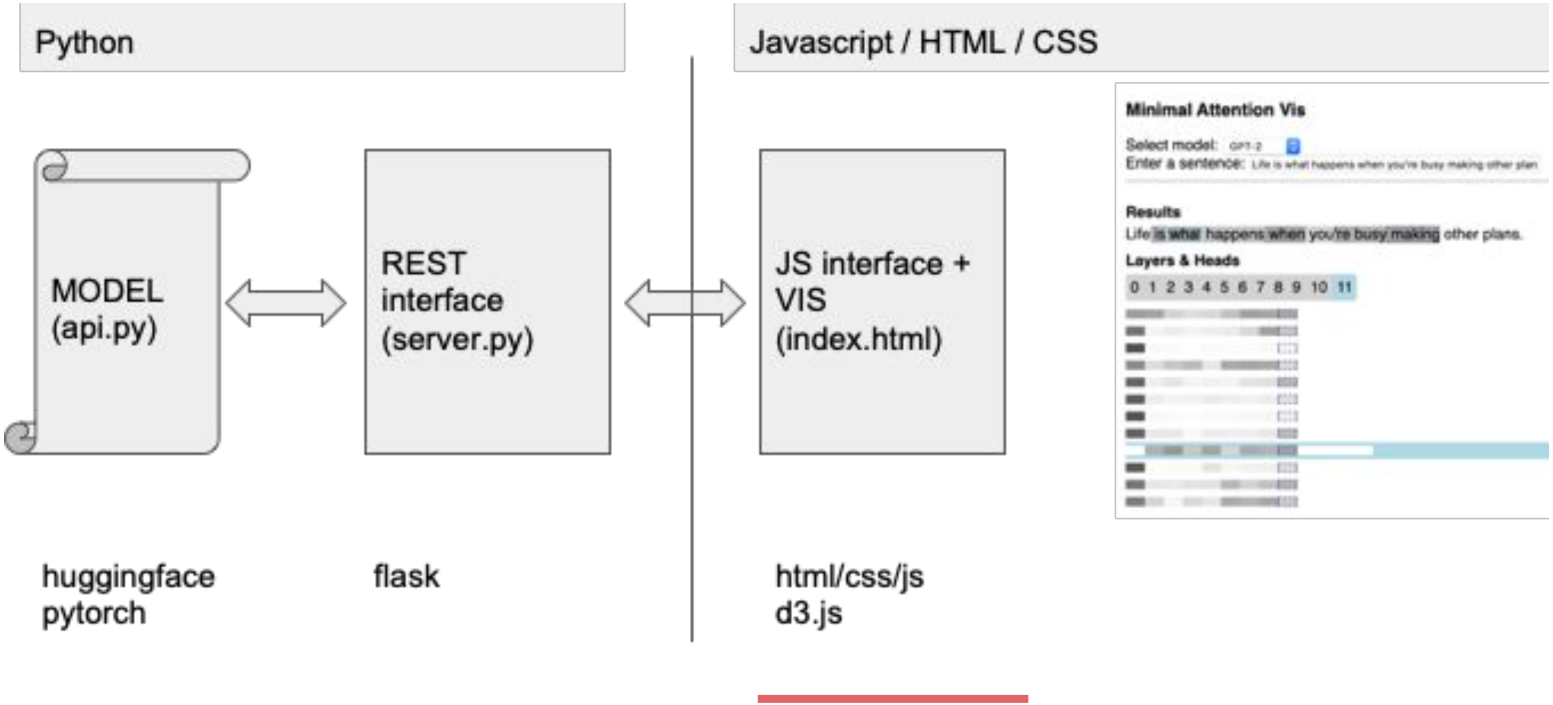
```python
@app.route('/api/attn', methods=['POST'])
def attn():
    sentence = request.json['sentence']
    model_name = request.json.get('model_name', 'distilbert-base-uncased')

    # lazy loading.
    if model_name not in loaded_models:
        loaded_models[model_name] = AttentionGetter(model_name)
    model = loaded_models[model_name]

    # Call on the model to get attention
    results = model.bert_analyze_text(sentence)

    # return object with request (sentence, model_name) and results.
    return json.dumps({
        "request": {"sentence": sentence, 'model_name': model_name},
        "results": results
    })
```

server.py


Python
Javascript / HTML / CSS
MODEL
(api.py)
REST
interface
(server.py)
JS interface +
VIS
(index.html)
Minimal Attention Vis
Select model:
Enter a sentence:
Results
Life is what happens when you're busy making other plans.
Layers & Heads
0 1 2 3 4 5 6 7 8 9 10 11
huggingface
pytorch
flask
html/css/js
d3.js

```html
<h3>Minimal Attention Vis</h3>
<div class="header">
    Select model: <select name="" id="model_select">
    <option value="gpt2"> GPT-2</option>
    <option value="distilbert-base-uncased"> DistilBert</option>
</select>
    <br>
    Enter a sentence:
    <input type="text" id="inputText"
           value="I dropped my pen in the mashed potatoes.">
    <button id="sendButton"> send </button>
</div>
<hr>
<div style="padding-top: 5px;">
    <div style="font-weight: bold; padding-top: 10px;">Results</div>
    <div id="results" style="padding-top: 5px;">

    </div>
    <div style="font-weight: bold; padding-top: 10px;">Layers & Heads</div>
    <div id="layers" style="padding-top: 5px;">
    </div>
    <div id="heads" style="padding-top: 5px;">
    </div>
</div>
```

client/index.html

```
// select input field.
const myInput = d3.select("#inputText");
// act when content changes.
myInput.on('change', () => triggerServerRequest());
// also act on clicking the send button.
d3.select('#sendButton').on('click', triggerServerRequest);

function triggerServerRequest (){
    // get input content and bind to var.
    const input_sentence = myInput.property('value');
    const model_name = d3.select('#model_select').property('value');

    // send everyth
    // and return a
    const server_qu
        method: "PO
        body: JSON.
            sentenc
            model_n
        }),
        headers: {
            "Conten
        }
    })
```

```
        // if Promise is fullfilled (aka: server response is back) then...
        server_query.then(response => {
            currentModel = response.request.model_name;
            currentResults = response.results;

            // don't change selectedToken unless text ist shorter
            selectedToken = Math.min(selectedToken,
              response.results.tokens.length - 1);

            // update layer buttons, heads visualization, text visualization
            updateLayerBtns(currentResults.attention.length);
            updateHeadsVis();
            updateTextVis();
        });
    }
```

client/index.html

```javascript
/**
 * update the layer buttons
 * @param no_btns -- number of buttons
 */
const updateLayerBtns = (no_btns) => {
    // create/update as many buttons as there are layers
    d3.select('#layers').selectAll('.btn').data(d3.range(no_btns))
      .join('div')
      .attr('class', 'btn')
      // most left/right buttons have round corners
      .classed('btn_l', d => d === 0)
      .classed('btn_r', d => d === (no_btns - 1))
      .text(d => d)
      .on('click', d => {
          // if clicked... set selected layer and update all VIS
          selectedLayer = d;
          updateLayerSelection();
          updateHeadsVis();
          updateTextVis();
      })

    updateLayerSelection();
};
```

1) *.selectAll* Select all .btn elements

   [btn1, btn2, …]

2) *.data* Set their data to the index value

   [(btn1, 0), (btn2, 1), …]

3) *.join* create/delete elements to match data

   [(btn1, 0), (btn2, 1), …]

4) *.classed* Conditionally set classes
5) *.text* Set their text to the index
6) *.on* Set their onClick handler

https://www.d3indepth.com/datajoins/

```js
const updateLayerSelection = () => {
    d3.select('#layers').selectAll('.btn')
      .classed('selected', d => d === selectedLayer);
}
```

1) *.selectAll* Select all .btn elements

   [btn1, btn2, …]

2) *.classed* Conditionally set classes

```
// defines gray scale for all heatmaps:
const colorScale = d3.scaleLinear().domain([0, 10, 100])
  .range(['#fff', '#aaa', '#4d4d4d'])

function updateHeadsVis() {
    // select heads vis root DOM node
    const headsDOM = d3.select('#heads');

    // in each heads Row add each token vis
    heads.selectAll('.attBox').data(d => d[selectedToken])
      .join('div')
      .attr('class', 'attBox')
      // highlight the token that has been selected
      .classed('selected', (d, i) => i === selectedToken)
      .style('background-color', d => colorScale(d))

}
```

Define a linear color scale variable

1) *.selectAll* Get all attention head elements
2) *.data* Filter attn values to those of the selected token and bind to head elements
3) *.join* Create/delete elements to match number of attention links
4) *.attr* Make sure all divs (even the just created one's) have the correct class
5) *.classed* highlight the selected token
6) *.style* Set the color to the color scale value

Minimal Attention Vis

Select model: Distil Bert

Enter a sentence: I dropped my pen in the mashed potatoes. send

---

Results

Layers & Heads

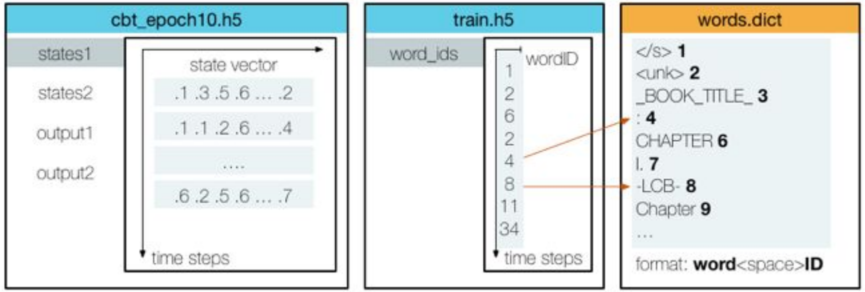
Call for Reproducibility and Public Adoption: open source with **documentation**

Adding Your Own Data

If you want to train your own data first, please read the Training document. If you have your own data at hand, adding it to LSTMVis is very easy. You only need three files:

- HDF5 file containing the state vectors for each time step (e.g. `cbt_epoch10.h5`)
- HDF5 file containing a word ID for each time step (e.g. `train.h5`)*
- Dict file containing the mapping from word ID to word (e.g. `words.dict`)*

A schematic representation of the data:



*If you don't have these files yet, but a space-separated `.txt` file of your training data instead, check out our text conversion tool

Config File

a simple example of an `lstm.yml` is:

```
name: children books  # project name
description: children book texts from the Gutenberg project # little description

files: # assign files to reference name
  states: cbt_epoch10.h5 # HDF5 files have to end with .h5 or .hdf5 !!!
  word_ids: train.h5
  words: words.dict # dict files have to end with .dict !!

word_sequence: # defines the word sequence
  file: train # HDF5 file
  path: word_ids # path to table in HDF5
  dict_file: words # dictionary to map IDs from HDF5 to words

states: # section to define which states of your model you want to look at
  file: states # HDF5 files containing the state for each position
  types: [
    {type: state, layer: 1, path: states1}, # type={state, output}, layer=[1..x], path = HDF5 path
    {type: state, layer: 2, path: states2},
    {type: output, layer: 2, path: output2}
  ]
```

Outline

- Structural analyses
- Behavioral analyses
- **Interactive visualizations**
  - Why do we want interactive visualizations?
  - Example: Identifying neuron purpose
  - Categorizing research in visualization
  - Hands-on with a simple attention visualization
  - **Future challenges and limitations**
- Other methods

Interaction and visualization matters at every step!

Understanding

Communicating challenging concepts
Awareness of limitations and flaws of an approach

Forming hypotheses

It reduces the exploration space
It helps us create hypotheses about data and models

Testing hypotheses

Counterfactual analysis
Connecting small insights to more expensive computation

Advantages of visual analytics

Understanding ← The design of the infrastructure of a VA tool *can* be easily extensible to new models

Forming hypotheses ← Much **faster** with interactive tools

Testing hypotheses ← More accessible through “**playing**” with a model

Disadvantages of visual analytics

🕒 💰 The development of VA tools is **expensive and time consuming**.

It is almost impossible to make tools **useful** across tasks.

💯 Accepting a hypothesis is often not possible without a full investigation,
a VA tool can thus often only be used as **additional step** in an analysis

Research opportunities in Interactive visualization

Human-in-the-Loop Model Correction
[Law et al. ‘20] [Cabrera et al. 19] [Lyytinen et al. ‘19]

Causality and Counterfactual What-If Analyses
[Strobelt et al. ‘18] [Wexler et al., ‘19]

Tighter integration of model + interface development
[Liu et al. ‘17] [Heer, ‘19] [Gehrmann et al.’19]

Evaluation for Usability and Utility
[Hohman et al., ‘18]

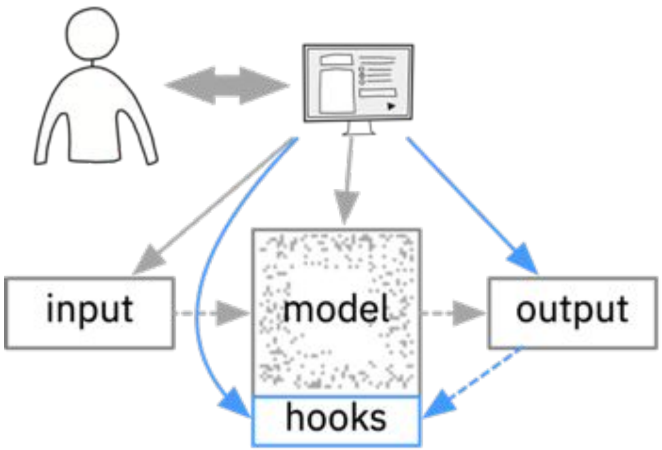
Research opportunities in Interactive visualization

Human-in-the-Loop Model Correction
[Law et al. ‘20] [Cabrera et al. 19] [Lyytinen et al. ‘19]

Causality and Counterfactual What-If Analyses
[Strobelt et al. ‘18] [Wexler et al., ‘19]

Tighter integration of model + interface development
[Liu et al. ‘17] [Heer, ‘19] [Gehrmann et al.’19]

Evaluation for Usability and Utility
[Hohman et al., ‘18]



Outline

- Structural analyses
- Behavioral analyses
- Interaction + Visualization
- **Other methods**

Other Topics

- Adversarial examples
  - Can point to model weaknesses
  - Challenges with text input (and output)
    - How to calculate gradients
    - How to measure similarity to real examples
  - Survey papers: Belinkov and Glass 2019, Wang et al. 2019, Zhang et al. 2019
- Generating explanations
  - Annotated explanations (Zaidan et al. 2007, Zhang et al. 2016)
  - Rationales: erasure-based (Li et al. 2016), latent variables (Lei et al. 2016)
  - Self-explaining models (Narang et al. 2020), translating neuralese (Andreas et al. 2017)
- Formal languages as models of language
  - For example: can LSTMs learn context-free languages?
  - Long line of research starting in the 1980s (Tonkes and Wiles 1997, Süzgün et al. 2019)

Conclusion

- Two broad approaches to interpreting neural NLP models:
  - Structural probing to analyze model representations and
  - Challenge sets to analyze structure
- Visualization techniques can speed up exploration of both structural/behavioral properties of models
- These techniques differ in their goals and assumptions
- Questionnaire can help assess contribution of a study or to choose appropriate approach for a given problem

Conclusion

- Open questions and directions for future work:
  - How can we make insights from these techniques actionable?
  - What is the connection between representations' structure (measured by probing techniques, visualizations) and model decisions (measured by challenge sets)?
  - Can techniques like probing classifiers be adapted to measure something less correlational, and more causal?
- Want more? See EMNLP tutorial on Interpreting Predictions of NLP Models (Wallace, Gardner, and Singh)