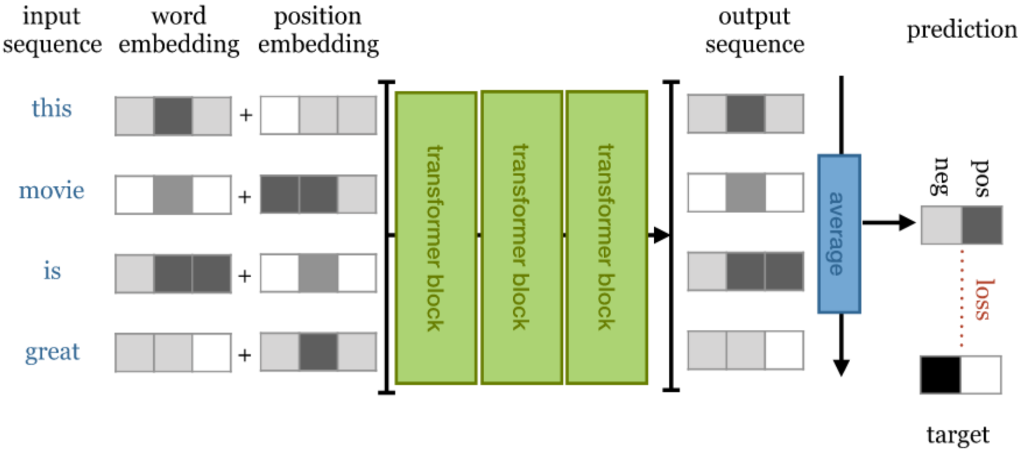
Text Data in Economics

Warwick QAPEC Summer School

10. Some Extras about Recent NLP

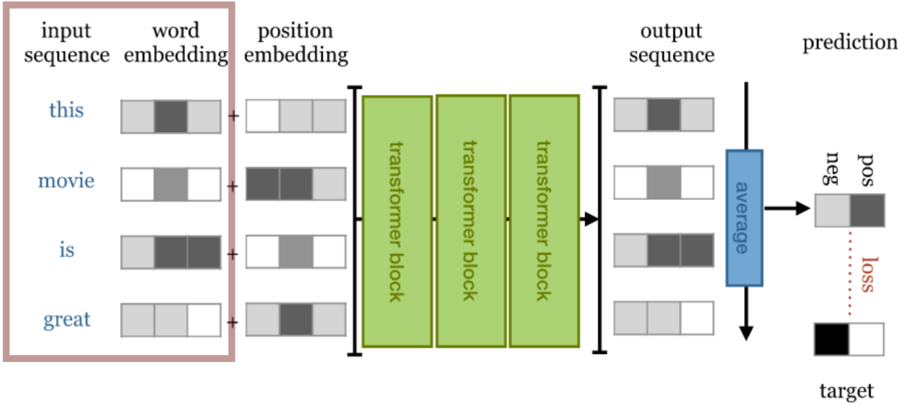
Outline

Transformer for Sentiment Classification



Transformer for Sentiment Classification

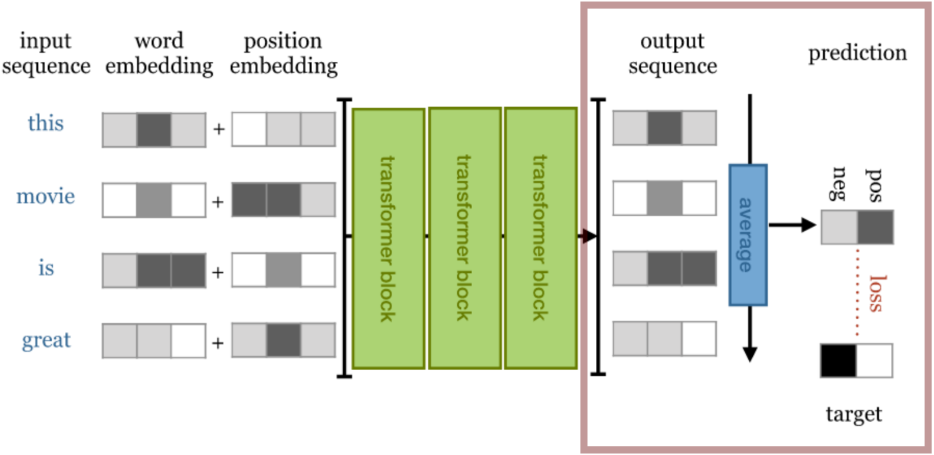
Input sequence → word embedding



- Input sequence of tokens $\{w_1, \ldots, w_i, \ldots, w_{n_L}\}$
- Trainable embedding vectors $[x_1 \ldots, x_i \ldots x_{n_L}]$

Transformer for Sentiment Classification

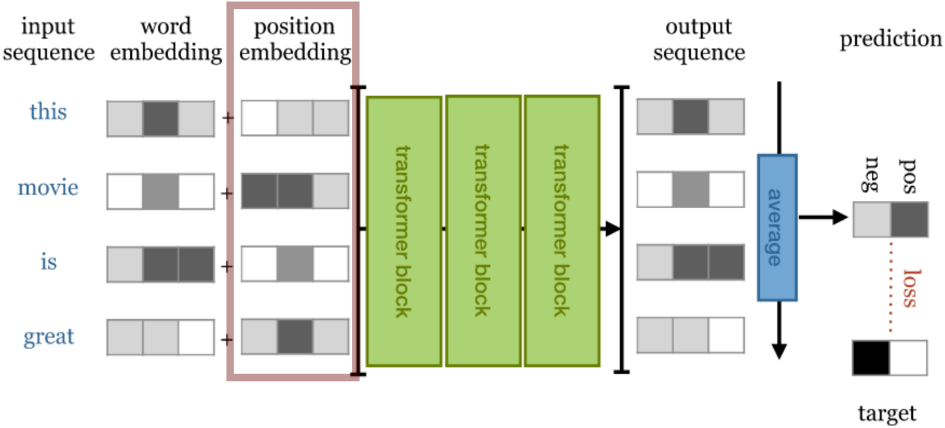
... → document embedding → sentiment score



- output sequence $\{h_1^y, \ldots, h_i^y, \ldots, h_{n_L}^y\}$
- averaged to produce **document vector** $\vec{d}$
- final output layer with sigmoid activation to produce probabilities $\hat{y}$ across positive and negative output classes.

Transformer for Sentiment Classification

... → position embedding → ...



Position Embeddings

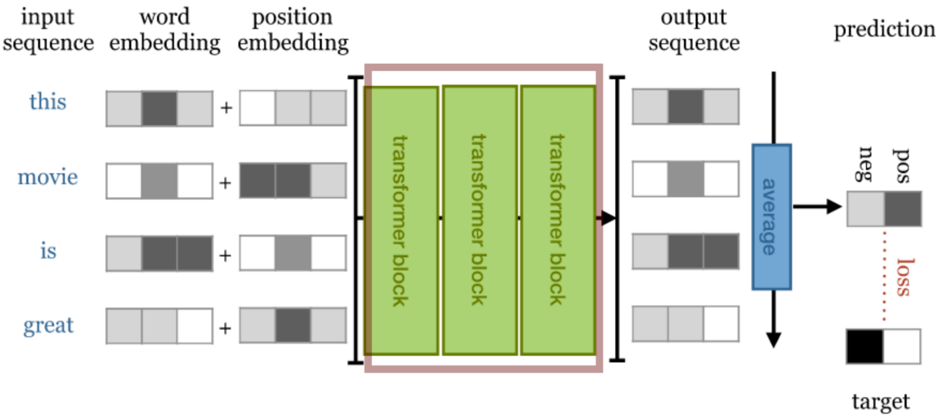
- To add word order information, transformers add a ***position embedding*** along with the ***word embedding*** as input to the attention layer.
- input to transformer block is

$$h^0 = \begin{bmatrix} x_1 & \dots & x_i & \dots & x_{n_L} \\ t_1 & \dots & t_i & \dots & t_{n_L} \end{bmatrix}$$

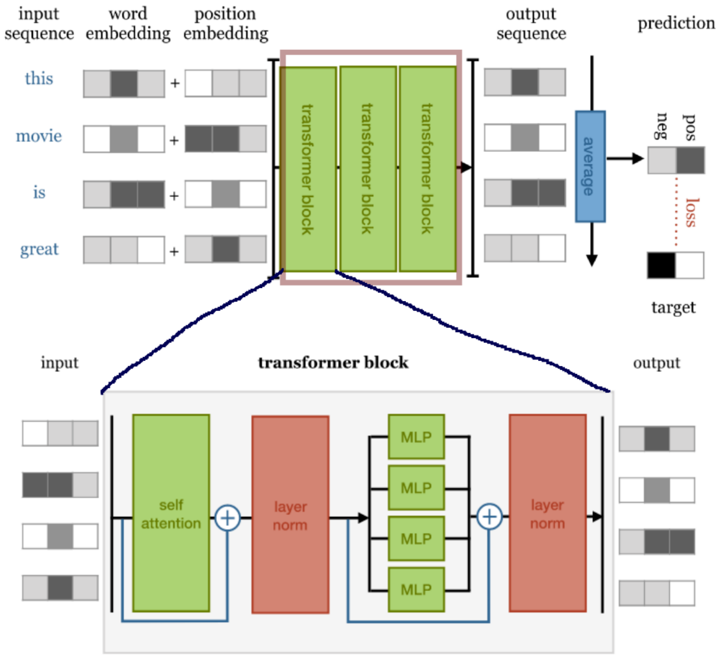
  which includes
  - word embeddings $\{x_1, ..., x_i, ..., x_{n_L}\}$ with dimension n_E
  - stacked with $\{t_1, ..., t_i, ..., t_{n_L}\}$, learnable categorical embeddings with dimension n_t for each index number i itself.
- Note:
  - puts a hard limit on sequence lengths
  - Positional encodings (or any direct information on word order) often not necessary after all (Irie et al 2019; Schlag et al 2021, Sinha et al 2021).

Transformer for Sentiment Classification

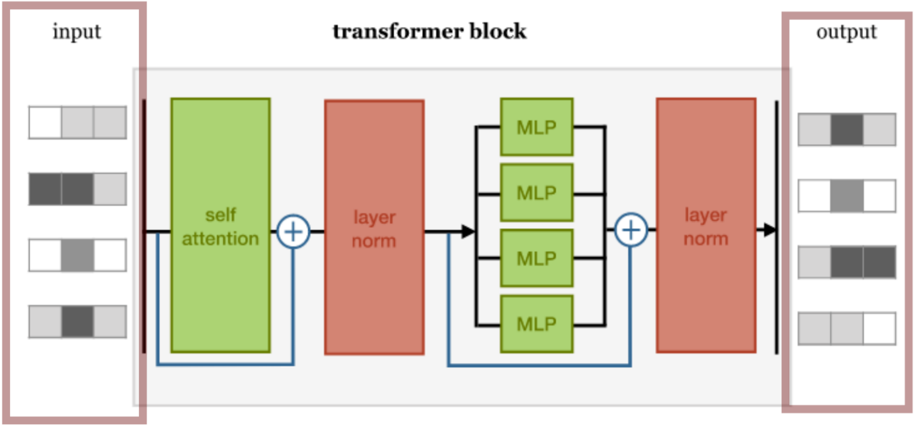
... → transformer blocks → ...



A **transformer** consists of **stacked transformer blocks**

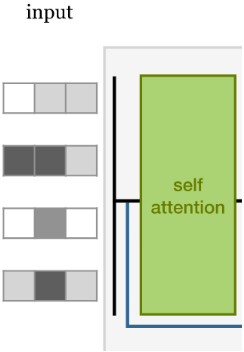


Transformer block (input and output)



- Each transformer block $l \in \{0, \ldots, n_y\}$ takes as input a sequence of vectors $h^l_{1:n_L}$ and outputs a sequence of vectors $h^l_{1:n_L}$, which become the input for the next transformer block.

Transformer Block (Self-Attention Layer)

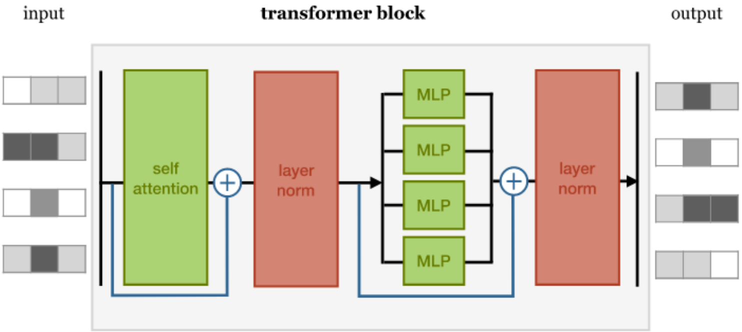


the "self attention" layer:

- input:
  - for the first block, includes the word embeddings and position embeddings h^0
  - for the later blocks, includes the output of the previous block h^l
- output:
  - matrix of self-attention-transformed vectors where item i is

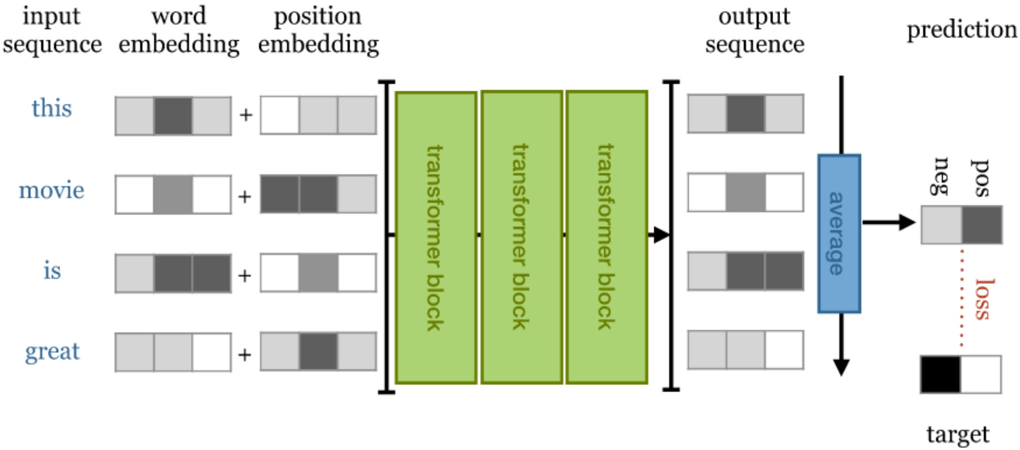
$$\sum_{j=1}^{n_L} a(h_i^l, h_j^l) h_j^l$$

The Transformer Block (Dense Layers)



- self-attention layer's outputs are normalized
  - we will come back to residual connections (blue line with $\oplus$) and "layer normalization" next week.
- piped to a multi-layer perceptron (MLP) with two hidden layers, with ReLU activation after the first layer.
- normalized again then output to h^{l+1}:
  - either to the next transformer block, or to the output layer h^{n_y}.

Transformer for Sentiment Classification



- will get state-of-the-art performance, and much faster to train than a bidirectional LSTM.

Outline

The Transformer Architecture

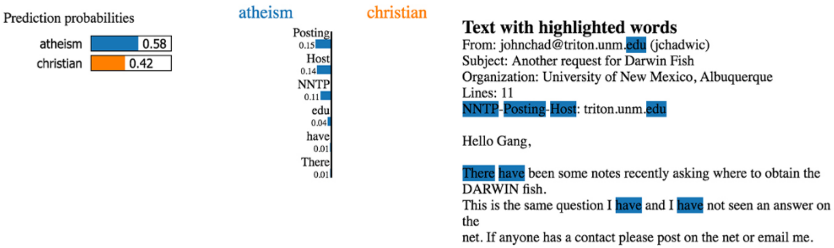
Advanced NLP Tasks

Bias in NLP Systems

Interpreting Black Box Text Classifiers using LIME

1. Generate new texts by randomly *removing* words from the original document.
2. Form predictions $\hat{y}$ from black box model for these perturbed documents.
3. Train lasso on dataset of binary features for each word, equaling one if word appears, to predict $\hat{y}$.
   - weight by proximity to initial data point (one minus the proportion of words dropped)

```
exp = explainer.explain_instance(test_example,
classifier.predict_proba, num_features=6)
```



- **Extractive summarization:**
  - create the summary from phrases or sentences in the source document(s)
  - e.g. MemSum (Gu et al, ACL 2022) is a light-weight reinforcement-learning model that scores sentences and then stops summarizing based on the extraction history.

- **Extractive summarization:**
  - create the summary from phrases or sentences in the source document(s)
  - e.g. MemSum (Gu et al, ACL 2022) is a light-weight reinforcement-learning model that scores sentences and then stops summarizing based on the extraction history.
- **Abstractive summarization:**
  - express the ideas in the source documents using (at least in part) different words
  - e.g., fine-tune **Big Bird Pegasus** to reconstruct provided summaries.

Open Question Answering and Claim Verification

Perhaps the most difficult global semantics tasks:

- Open question answering:
  - Answer any question.
  - “What are the responsibilities of the mayor of Zurich?”
- Open claim verification:
  - Check whether a plain-text claim is true or false.
  - “Zurich has the second-highest per-capita income of any city in Europe.”

Open Question Answering and Claim Verification

Perhaps the most difficult global semantics tasks:

- Open question answering:
  - Answer any question.
  - "What are the responsibilities of the mayor of Zurich?"
- Open claim verification:
  - Check whether a plain-text claim is true or false.
  - "Zurich has the second-highest per-capita income of any city in Europe."
- Both problems are solved using information retrieval pipelines:
  - search large corpora or knowledge graphs for evidence
  - use evidence to answer the question or check the claim

Outline

Bias in NLP Systems

Sentiment Analysis

```
text_to_sentiment("Let's go get Italian food")
2.0429166109
text_to_sentiment("Let's go get Chinese food")
1.4094033658
text_to_sentiment("Let's go get Mexican food")
0.3880198556
```

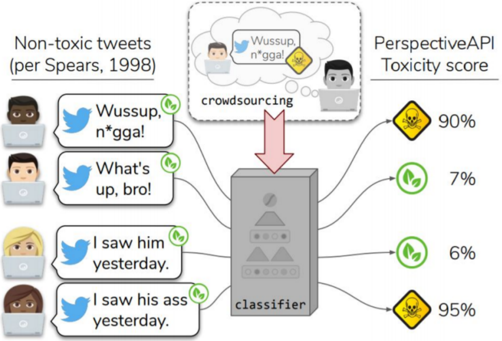
```
text_to_sentiment("My name is Emily")
2.2286179365
text_to_sentiment("My name is Heather")
1.3976291151
text_to_sentiment("My name is Yvette")
0.9846380213
text_to_sentiment("My name is Shaniqua")
-0.4704813178
```

Is this sentiment model racist?

Source: Kareem Carr slides.

Bias in NLP Systems

Toxicity Detection



Within dataset proportions

Dataset	Group	Acc.	% false identification: None	% false identification: Offensive	% false identification: Hate
DWMW17	AAE	94.3	1.1	**46.3**	0.8
	White	87.5	**7.9**	9.0	**3.8**
	Overall	91.4	2.9	17.9	2.3

Dataset	Group	Acc.	% false identification: None	% false identification: Abusive	% false identification: Hateful
FDCL18	AAE	81.4	4.2	**26.0**	**1.7**
	White	82.7	**30.5**	4.5	0.8
	Overall	81.4	20.9	6.6	0.8

Is this toxicity detection model racist?

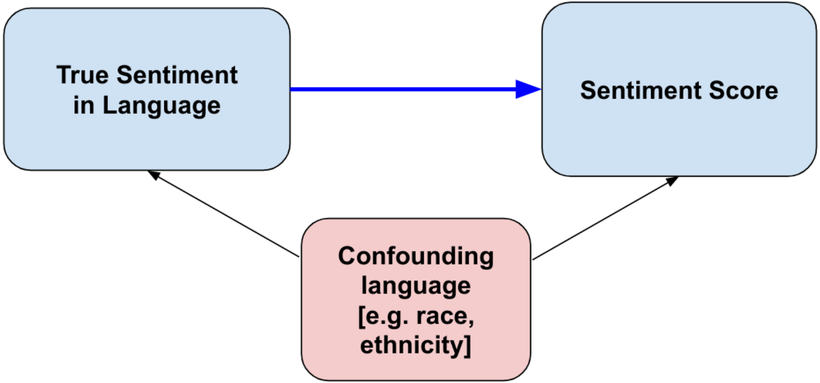
Source: Jacobs and Wallach slides.

NLP "Bias" is statistical bias

- Sentiment scores that are trained on annotated datasets also learn from the correlated non-sentiment information.

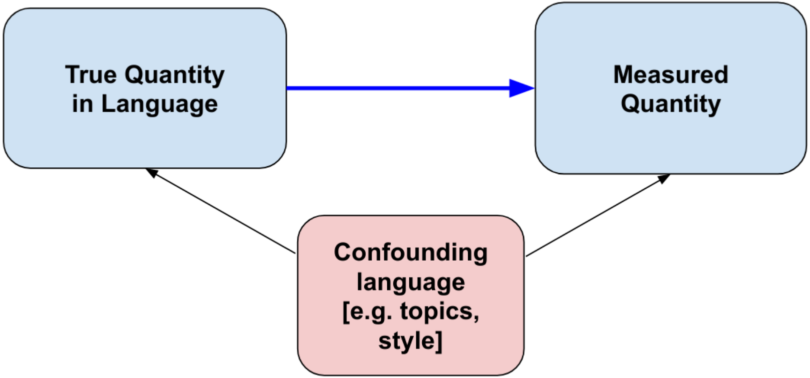
NLP "Bias" is statistical bias

- Sentiment scores that are trained on annotated datasets also learn from the correlated non-sentiment information.



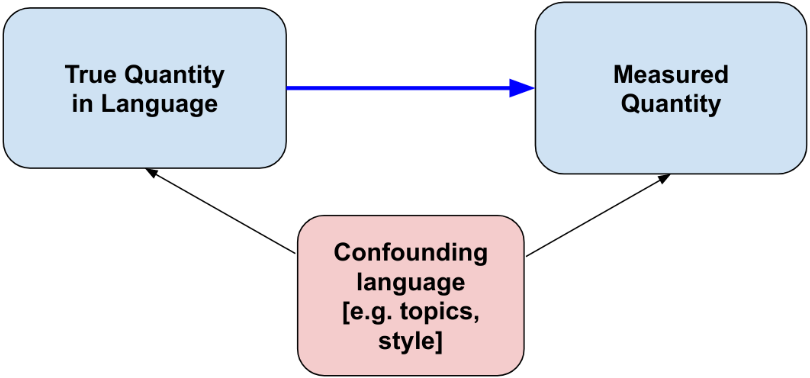
- Supervised sentiment models are confounded by correlated language factors.
  - e.g., in the training set maybe people complain about Mexican food more often than Italian food.

This is a universal problem



- supervised models (classifiers, regressors) learn features that are correlated with the label being annotated.
- unsupervised models (topic models, word embeddings) learn correlations between topics / contexts.

This is a universal problem



- supervised models (classifiers, regressors) learn features that are correlated with the label being annotated.
- unsupervised models (topic models, word embeddings) learn correlations between topics / contexts.
- An important exception: dictionary methods (perhaps explaining why they are often used by economists). But they have other serious limitations.

Examples: Confounders in Measurement from Text

What quantity do we care about? vs. What do we measure?

Examples: Confounders in Measurement from Text

What quantity do we care about? vs. What do we measure?

- Positive/negative sentiment $\rightarrow$ Count positive/negative words, or predict text annotations.
- Toxicity $\rightarrow$ Count toxic words, or predict text annotations.

confounders?

Examples: Confounders in Measurement from Text

What quantity do we care about? vs. What do we measure?

- Positive/negative sentiment → Count positive/negative words, or predict text annotations.
- Toxicity → Count toxic words, or predict text annotations.

confounders?

- Student performance → predicted essay grade based on labeled essay documents.
- Credit worthiness → predicted probability of default based on loan application documents.

confounders?

Examples: Confounders in Measurement from Text

What quantity do we care about? vs. What do we measure?

- Positive/negative sentiment $\rightarrow$ Count positive/negative words, or predict text annotations.
- Toxicity $\rightarrow$ Count toxic words, or predict text annotations.

confounders?

- Student performance $\rightarrow$ predicted essay grade based on labeled essay documents.
- Credit worthiness $\rightarrow$ predicted probability of default based on loan application documents.

confounders?

- Political partisanship $\rightarrow$ predicted probability being Democrat/Republican based on speeches.

confounders?

Examples: Confounders in Measurement from Text

What quantity do we care about? vs. What do we measure?

- Positive/negative sentiment $\rightarrow$ Count positive/negative words, or predict text annotations.
- Toxicity $\rightarrow$ Count toxic words, or predict text annotations.

confounders?

- Student performance $\rightarrow$ predicted essay grade based on labeled essay documents.
- Credit worthiness $\rightarrow$ predicted probability of default based on loan application documents.

confounders?

- Political partisanship $\rightarrow$ predicted probability being Democrat/Republican based on speeches.

confounders?

- Policy priorities $\rightarrow$ predicted probability of speeches/laws being about a particular policy topic.

confounders?

When is measurement confounding important?

- By itself, producing measurements that are biased by confounders might not be a problem.
- e.g.:
  - an NLP-based credit score that learns confounders $\rightarrow$ not a problem unless debtors learn about it and strategically alter their documents.
    - similarly with automated essay grading

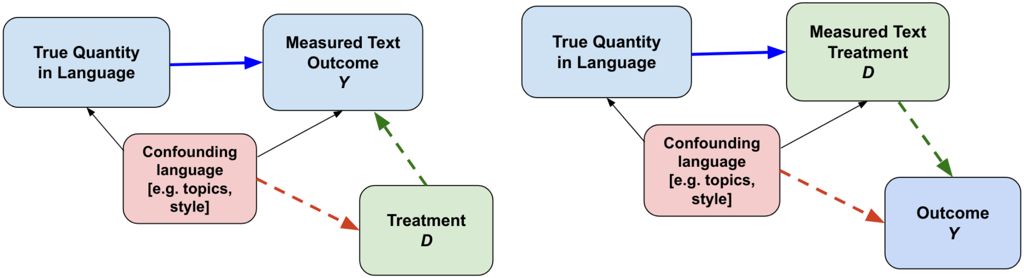
When is measurement confounding important?

- By itself, producing measurements that are biased by confounders might not be a problem.
- e.g.:
  - an NLP-based credit score that learns confounders $\rightarrow$ not a problem unless debtors learn about it and strategically alter their documents.
    - similarly with automated essay grading
- for measuring political divisiveness or policy priorities
  - probably won't matter for in-domain summary statistics
  - but would matter a lot for summary statistics in a new domain

When is measurement confounding important?

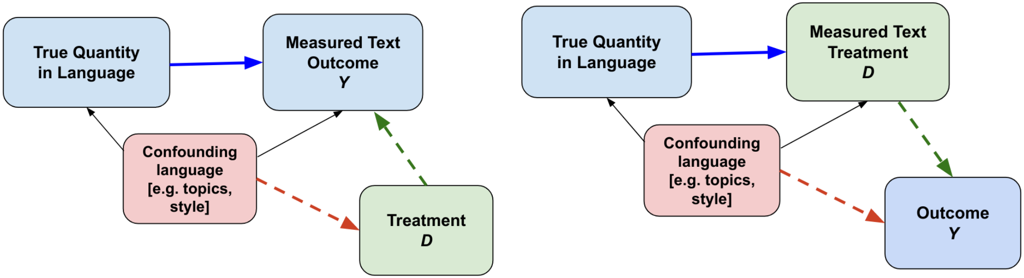
- By itself, producing measurements that are biased by confounders might not be a problem.
- e.g.:
  - an NLP-based credit score that learns confounders $\rightarrow$ not a problem unless debtors learn about it and strategically alter their documents.
    - similarly with automated essay grading
- for measuring political divisiveness or policy priorities
  - probably won't matter for in-domain summary statistics
  - but would matter a lot for summary statistics in a new domain
- even in domain, will matter for assessing the causal effect of a treatment, e.g. the electoral cycle:
  - elections might cause politicians to focus on social issues rather than economic issues,
  - if social/economic issues are confounded with partisanship, the resulting estimates are biased.

When is measurement confounding important?



- When text is outcome, the confounders cannot be correlated with the treatment.
- When text is treatment, the confounders cannot be correlated with the outcome.

When is measurement confounding important?



- When text is outcome, the confounders cannot be correlated with the treatment.
- When text is treatment, the confounders cannot be correlated with the outcome.
  - e.g.: estimating the effect of politician speech sentiment on his/her reelection chances?

Steps for de-biasing

- Language features that are often confounded with the quantity of interest:
  - stopwords
  - named entities: person/organization/place names
- These can be dropped during pre-processing to reduce the influence of confounders in subsequent measurements.
- Can control for topic or style features or other potential confounders in regressions.