Hard battles and easy victories in robustifying NLP models

He He

RobustSeq Workshop

December 2, 2022
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label = +1</th>
<th>label = −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the highest calibre!</td>
<td>Thank God I didn’t go to the cinema.</td>
</tr>
<tr>
<td>Definitely worth the watch!</td>
<td>Boring as hell.</td>
</tr>
<tr>
<td>A true story told perfectly!</td>
<td>I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label = +1</th>
<th>label = −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the highest calibre!</td>
<td>Thank God I didn’t go to the cinema.</td>
</tr>
<tr>
<td>Definitely worth the watch!</td>
<td>Boring as hell.</td>
</tr>
<tr>
<td>A true story told perfectly!</td>
<td>I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>

Two equally good hypotheses:

- Predict +1 if the input ends with “!”
- Predict +1 is the input gives a positive recommendation
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label = +1</th>
<th>label = −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the highest calibre!</td>
<td>Thank God I didn’t go to the cinema.</td>
</tr>
<tr>
<td>Definitely worth the watch!</td>
<td>Boring as hell.</td>
</tr>
<tr>
<td>A true story told perfectly!</td>
<td>I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>

Two equally good hypotheses:

- Predict +1 if the input ends with “!”
- Predict +1 is the input gives a positive recommendation
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label = +1</th>
<th>label = −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the highest calibre!</td>
<td>Thank God I didn’t go to the cinema.</td>
</tr>
<tr>
<td>Definitely worth the watch!</td>
<td>Boring as hell.</td>
</tr>
<tr>
<td>A true story told perfectly!</td>
<td>I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>

Two equally good hypotheses:

- Predict +1 if the input ends with "!"
- Predict +1 is the input gives a positive recommendation
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label = +1</th>
<th>label = −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the highest calibre! Thank God I didn’t go to the cinema.</td>
<td>Definitely worth the watch! Boring as hell.</td>
</tr>
<tr>
<td>Definitely worth the watch!</td>
<td>A true story told perfectly! I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>

Two equally good hypotheses:

- Predict +1 if the input ends with “!”
- Predict +1 if the input gives a positive recommendation

Distribution shift due to perturbation of the spurious feature:

Complete waste of two hours of my time! +1/ − 1?
A motivation example: sentence classification

<table>
<thead>
<tr>
<th>label= +1</th>
<th>label= −1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riveting film of the <strong>highest calibre</strong>!</td>
<td>Thank God I didn’t go to the cinema.</td>
</tr>
<tr>
<td>Definitely <strong>worth the watch</strong>!</td>
<td>Boring as hell.</td>
</tr>
<tr>
<td>A true story told <strong>perfectly</strong>!</td>
<td>I wanted to give up in the first hour...</td>
</tr>
</tbody>
</table>

Two equally good hypotheses:

- Predict +1 if the input ends with “!”
- Predict +1 if the input gives a positive recommendation

Distribution shift due to perturbation of the spurious feature:

Complete waste of two hours of my time! +1/ − 1?

Models may not generalize as expected in deployment domains
Real examples

Biases in NLP datasets:

- **NLI**: negation words $\rightarrow$ contradiction [Poliak et al., 2018]
- **NLI**: lexical overlap $\rightarrow$ entailment [McCoy et al., 2019]
- **Paraphrase identification**: lexical overlap $\rightarrow$ paraphrase [Zhang et al., 2019]
- **QA**: lexical overlap $\rightarrow$ answer sentence [Jia and Liang, 2017]
- **Co-reference**: gender $\rightarrow$ occupation [Zhao et al., 2018]

Large performance drop on OOD data where the simple heuristic fails
Challenges

What assumption should we make about spurious features (in language data)?

- $p(y \mid x)$ should be **invariant** to perturbations of the spurious feature
- The label should be **independent** of the spurious feature
- The learned **representation** should not contain information about the spurious feature

Recipe: assumptions $\rightarrow$ objective $\rightarrow$ optimization

Common assumptions may not apply to certain spurious features in NLP data
Some spurious features are irrelevant

The simple case: spurious features and core features are *disentangled*

- Changing the spurious feature doesn’t affect label
Some spurious features are irrelevant

The simple case: spurious features and core features are *disentangled*

- Changing the spurious feature doesn’t affect label

  *Spielberg’s new film is brilliant*  positive
  *Zhang’s new film is brilliant*  positive
Some spurious features are irrelevant

**The simple case:** spurious features and core features are *disentangled*

- Changing the spurious feature doesn’t affect label

  Spielberg’s new film is brilliant  positive
  Zhang’s new film is brilliant  positive

  water → waterbird  
  land → waterbird
Some spurious features are necessary for prediction

The complex case: spurious features are part of the core features

- The “spurious” feature is necessary but not sufficient for prediction
Some spurious features are necessary for prediction

The complex case: spurious features are part of the core features
- The “spurious” feature is necessary but not sufficient for prediction

I love dogs / I don’t love dogs contradiction
I love dogs / I don’t love cats neutral

stripes → zebra
stripes → crosswalk
Two ways for a word to associate with the label

Joshi et al., 2022

Titanic is great

- C: the review writer
- Y: sentiment
- Titanic has no causal relation with Y
- But they may be correlated through C: famous movies tend to receive good reviews

The spurious feature is irrelevant to predicting the label.
Two ways for a word to associate with the label

- C: the review writer
- Y: sentiment
- not causally affects Y

The spurious feature is necessary to predicting the label.
A feature is **spurious** if it is **not sufficient** for predicting the label.

But it may be necessary for prediction:

<table>
<thead>
<tr>
<th>Irrelevant</th>
<th>Necessary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Titanic is great</td>
<td>I don’t like the movie</td>
</tr>
</tbody>
</table>
A feature is **spurious** if it is **not sufficient** for predicting the label.

But it may be necessary for prediction:

<table>
<thead>
<tr>
<th>Irrelevant</th>
<th>Necessary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Titanic is great</td>
<td>I don’t like the movie</td>
</tr>
<tr>
<td>Has no causal relation with the label</td>
<td>Causally affect the label</td>
</tr>
<tr>
<td>Model should be invariant to them</td>
<td>Model should be sensitive to them</td>
</tr>
</tbody>
</table>

More common in NLP (messier...)

How well do existing methods work on different types of spurious features?
Setup

- **Dataset**: MNLI
- **Model**: finetuned RoBERTa-Large
Setup

• **Dataset**: MNLI
• **Model**: finetuned RoBERTa-Large

• **Spurious features**:
  • **Irrelevant**: adding !! to the end of neutral
    The kids are playing football.
    The kids are shouting!! (neutral)
  • **Necessary**: lexical overlap and entailment [McCoy et al., 2019]
    The woman is selling sweets to the kids.
    The woman is selling sweets. (entailment)

• **Methods**:
  • Data balancing through subsampling
  • Representation debiasing
Data balancing

**Assumption:** the label should be independent of the spurious feature

**Method:** subsample the data s.t. label is independent of the feature [Sagawa et al., 2020; i.a.]

high overlap examples

Does the model generalize well if the spurious feature is independent of the label?
Breaking the spurious correlation is not enough

<table>
<thead>
<tr>
<th>ID</th>
<th>OOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>high overlap</td>
<td>low overlap</td>
</tr>
<tr>
<td>has punctuation</td>
<td>no punctuation</td>
</tr>
</tbody>
</table>

\[ y \perp x_{\text{spurious}} \]

**Takeaway**: feature-label independence leads to
- good OOD performance for **irrelevant features**
- but we still see large ID-OOD performance gap for **necessary features**
Effect of data balancing

**Irrelevant** spurious features:

- Breaking the correlation allows the model to learn the core features
- Core features are the same with and without the spurious feature

Titanic is great

Necessary spurious features:

- Core features vary with the spurious feature
- The model encounters new/rare features on OOD examples

It's not C good

---

Effect of data balancing

**Irrelevant** spurious features:
- Breaking the correlation allows the model to learn the core features
- Core features are the same with and without the spurious feature

**Necessary** spurious features:
- Core features vary with the spurious feature
- The model encounters new/rare features on OOD examples

---

Removing necessary features from the representation hurt performance

What if we remove spurious features from the learned representation?

Takeaway: removing spurious features
• does not affect task accuracy for irrelevant features
• but degrades task accuracy for necessary features
Removing necessary features from the representation hurt performance

What if we remove spurious features from the learned representation?

![Graph showing accuracy over iterations of INLP for different types of accuracy measures: word-overlap bias - task accuracy, punctuation bias - task accuracy, word-overlap probing accuracy, punctuation probing accuracy.]

**Probing accuracy**: lower $\rightarrow$ the feature gets removed

Takeaway:
- Removing spurious features does not affect task accuracy for irrelevant features.
- But degrades task accuracy for necessary features.
Removing necessary features from the representation hurt performance

What if we remove spurious features from the learned representation?

Task accuracy: higher → the representation is useful for NLI

Probing accuracy: lower → the feature gets removed
Removing necessary features from the representation hurt performance

What if we remove spurious features from the learned representation?

**Task accuracy**: higher $\rightarrow$ the representation is useful for NLI

Removal of necessary features degrades task performance

**Probing accuracy**: lower $\rightarrow$ the feature gets removed

Removing necessary features from the representation hurt performance

What if we remove spurious features from the learned representation?

**Task accuracy**: higher → the representation is useful for NLI

Removal of necessary features degrades task performance

**Probing accuracy**: lower → the feature gets removed

**Takeaway**: removing spurious features

- does not affect task accuracy for irrelevant features
- but degrades task accuracy for necessary features
Summary so far

What assumption should we make about spurious features (in language data)?

- **The nice setting**: we know the spurious feature, and it is irrelevant to prediction
  - Break the feature-label correlation (subsampling, reweighting, invariance etc.)
Summary so far

What assumption should we make about spurious features (in language data)?

• **The nice setting**: we know the spurious feature, and it is irrelevant to prediction
  • Break the feature-label correlation (subsampling, reweighting, invariance etc.)
• **The real setting**: we don’t know the spurious feature, there are many of them, and they may be necessary for prediction
  • Existing independence or invariance assumptions do not apply
  • Learn patterns on the long tail

Next, what’s the role of pretraining in robust language understanding?
Robustness in the era of large language models

Supervised learning → zero-shot / in-context learning

- What’s the underlying distribution shift?
- What’s the inductive bias of pretraining and prompting?

Our study: Is in-context learning robust to spurious correlations in the demonstration?
• **Prompt** with *spurious correlation:*
  
  Input: Riveting film of the highest calibre! Label: +1
  Input: Thank God I didn’t go to the cinema. Label: -1
  Input: Definitely worth the watch! Label: +1
  Input: Boring as hell. Label: -1
Setup

- **Prompt** with spurious correlation:
  - Input: Riveting film of the highest calibre! Label: +1
  - Input: Thank God I didn’t go to the cinema. Label: -1
  - Input: Definitely worth the watch! Label: +1
  - Input: Boring as hell. Label: -1

- **Features**: semi-synthesized spurious features (punctuation, n-grams etc.)
Setup

- **Prompt** with spurious correlation:
  
  - Input: Riveting film of the highest calibre! Label: +1
  - Input: Thank God I didn’t go to the cinema. Label: -1
  - Input: Definitely worth the watch! Label: +1
  - Input: Boring as hell. Label: -1

- **Features**: semi-synthesized spurious features (punctuation, n-grams etc.)

- **Metric ↓**: gap between bias-support and bias-countering examples on test set
  
  - Input: A story told perfectly! Label:
  - Input: Complete waste of time! Label:
Setup

• **Prompt** with spurious correlation:
  Input: Riveting film of the highest calibre! Label: +1
  Input: Thank God I didn’t go to the cinema. Label: -1
  Input: Definitely worth the watch! Label: +1
  Input: Boring as hell. Label: -1

• **Features**: semi-synthesized spurious features (punctuation, n-grams etc.)

• **Metric** ↓: gap between bias-support and bias-countering examples on test set
  Input: A story told perfectly! Label:
  Input: Complete waste of time! Label:

• **Models**: Curie (13B), Davinci (175B, original GPT-3)
Is in-context learning robust to biases in the demonstration?

[Si et al., 2022]

Controlling the strength of prompt bias:

- Prevalence: % examples with the spurious feature
- Strength: $p(y \mid x_{\text{spurious}})$

![Graph showing the prevalence and strength of prompt bias for different datasets and models]

Dataset = SST-2 (Curie)

Punc

N-Gram

Spurious Gap

50

25

20

0

Dataset = SST-2 (Davinci)

Punc

N-Gram

Prevalence/Strength (%)

50/100

20/90

Preliminary work
Is in-context learning robust to biases in the demonstration?

[Si et al., 2022]

Controlling the strength of prompt bias:

• Prevalence: % examples with the spurious feature
• Strength: \( p(y \mid x_{\text{spurious}}) \)

- GPT-3 suffers from **extreme bias** in the prompt, but less so under **weaker bias**.
- The **larger model** seems to infer the intended task even under extreme bias.

Preliminary work
Is in-context learning robust to biases in the demonstration?

What’s the effect of prompt engineering?

*Input: Riveting film of the highest calibre! Label: positive*
*Input: Thank God I didn’t go to the cinema. Label: negative*

- Using **verbalized labels** helps the model learn the target task to some extent
Is in-context learning robust to biases in the demonstration?

What if we give it more in-context examples?

• Still with uninformative labels (0/1) and extreme bias (100% predictive)

More (biased) examples help the model infer the intended task
• which is different from supervised learning

Preliminary work
Summary so far

- Scaling up LLMs improve robustness without explicit assumptions about the spurious features
- They are still susceptible to spurious correlations in the demo examples
- But proper prompt design can mitigate the problem by informing the model of the intended task
Parting remarks

Takeaways:
• Tackling all sorts of spurious features in NLP tasks is a hard battle
• Pretraining and scaling have consistently improved model robustness so far

Open questions:
• What is OOD wrt to pretraining (long-tail events, human biases)?
• How does prompting or in-context learning work?
• How does human interaction / feedback help?
Collaborators

Nitish Joshi  Xiang Pan  Shi Feng  Danqi Chen  Dan Friedman  Chenglei Si

Thank you!