Encoding Logic Rules in Sentiment Classification

Kalpesh Krishna*
UMass Amherst

Preethi Jyothi
IIT Bombay

Mohit Iyyer
UMass Amherst

* Work done at IIT Bombay
Sentiment Classification
Sentiment Classification

Classify a sentence as positive or negative
Sentiment Classification

Classify a sentence as **positive** or **negative**

*this movie has a **great** story*
Sentiment Classification

Classify a sentence as **positive** or **negative**

*this movie has a **great** story*

Sentiment = **Positive**
Not Always Easy!

this movie has a great story

Solution :- Lexicons, Bag of Words
Not Always Easy!

Easy!

this movie has a great story

Solution :- Lexicons, Bag of Words
Not Always Easy!

Easy!

*this movie has a great story*

Solution :- Lexicons, Bag of Words

Contrastive - *this movie is funny, but horribly directed*

Negation - *this is not a movie worth waiting for*
Not Always Easy!

Easy!

this movie has a great story

Solution :- Lexicons, Bag of Words

Much Harder!

Contrastive - this movie is funny, but horribly directed

Negation - this is not a movie worth waiting for
this movie is funny, but horribly directed
A-but-B
Logic Rules

this movie is funny, but horribly directed

A-but-B

sentiment(A-but-B) = sentiment(B)
<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
</table>

Neural Nets + Logic Rules?
Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td></td>
</tr>
</tbody>
</table>
# Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td>Contribution #1&lt;br&gt;replication study finds <strong>incorrect</strong> conclusions</td>
</tr>
</tbody>
</table>
# Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td>Contribution #1 replication study finds <strong>incorrect</strong> conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td><strong>Contribution #1</strong> replication study finds <strong>incorrect</strong> conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td>Peters et al. (NAACL 2018)</td>
<td></td>
</tr>
</tbody>
</table>
## Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td><strong>Contribution #1</strong> replication study finds incorrect conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td>Peters et al. (NAACL 2018)</td>
<td><strong>Contribution #2</strong> ELMo embeddings learn logic rules <strong>without</strong> explicit supervision</td>
</tr>
</tbody>
</table>
Neural Nets + Logic Rules?

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td>Contribution #1 replication study finds incorrect conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td>Peters et al. (NAACL 2018)</td>
<td>Contribution #2 ELMo embeddings learn logic rules without explicit supervision</td>
</tr>
</tbody>
</table>

**Contribution #3**
Robustness of explicit / implicit methods to varying annotator agreement in A-but-B sentences
Digression: Reproducibility
Digression: Reproducibility

**Small** benchmark datasets (SST, MR, CR)

**Significant variation** in performance every run (due to random initialization / GPU parallelization)
Digression: Reproducibility

**Small** benchmark datasets (SST, MR, CR)

**Significant variation** in performance every run (due to random initialization / GPU parallelization)

**Solution**: Average performance over a large number of random seeds (Reimers and Gurevych 2017)
Large Variation (100 seeds)
Large Variation (100 seeds)
Large Variation (100 seeds)
# Outline

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td>Contribution #1 replication study finds incorrect conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td>Peters et al. (NAACL 2018)</td>
<td>Contribution #2 ELMo embeddings learn logic rules without explicit supervision</td>
</tr>
</tbody>
</table>

**Contribution #3**
Robustness of explicit / implicit methods to varying annotator agreement in A-but-B sentences
Model in Hu et al. 2016

**E**: Projection (Ganchev et al. 2010)

**M**: Distillation (Hinton et al. 2014)
E: Projection (Ganchev et al. 2010)
E: Projection (Ganchev et al. 2010)

this movie is funny, but horribly directed
E: Projection (Ganchev et al. 2010)

this movie is funny, but horribly directed

\[ p_{\theta}(y|x) \]

negative = 0.34
positive = 0.66
**E: Projection** (Ganchev et al. 2010)

this movie is **funny**, **but** horribly directed

\[ p_\theta(y|x) \]

negative = 0.34

positive = 0.66
E: Projection (Ganchev et al. 2010)

this movie is funny, but horribly directed

\[ p_\theta(y|x) \quad \text{project} \quad q_\theta(y|x) \]

negative = 0.34
positive = 0.66

negative = 0.77
positive = 0.23
E: Projection (Ganchev et al. 2010)

this movie is funny, but horribly directed

\[ p_\theta(y|x) \quad \text{project} \quad q_\theta(y|x) \]

negative = 0.34
positive = 0.66

negative = 0.77
positive = 0.23

projection is a convex optimization problem
E: Projection (Ganchev et al. 2010)

this movie is funny, but horribly directed

\[ p_\theta(y|x) \quad \text{project} \quad q_\theta(y|x) \]

negative = 0.34
positive = 0.66

negative = 0.77
positive = 0.23

projection is a convex optimization problem
new distribution consistent with logic rules
M: Distillation (Hinton et al. 2014)

\[ L = \lambda H(p_{\text{truth}}, p_{\theta}) + (1 - \lambda) H(q_{\theta}, p_{\theta}) \]

train model with projected distribution as soft-label
Hu et al. 2016 algorithm

\textbf{E}: Projection

\textbf{M}: Distillation

\texttt{forall minibatch (x,y) \{ \\
                    p = forward(x) \\
                    q = project(p) \\
                    theta += grad-update(p, q, y) \\
\}}
Conclusions in Hu et al. 2016

1) Distilled model better than single projection

2) Distilled neural network has significant gain on SST2 as it learns A-but-B rule
Our Conclusions

1) Distilled model **better** than single projection

2) Distilled neural network has **significant gain** on SST2 as it **learns A-but-B rule**

1) A **single projection** is a good way to explicitly encode logic rules

2) Distilled neural nets **aren't learning logic rules**
Distillation is ineffective
Single Projection good!

- Reported
- Averaged

Graph showing:
- Distillation: Reported 1.75
- Distillation + Single Projection: Reported 2.25
- Single Projection: Averaged 1.5
Single Projection sufficient!
Single Projection sufficient!

- Distillation
- Single Projection
- Distill + Single Project

Reported
Averaged
Consistent Trend on A-but-B

<table>
<thead>
<tr>
<th>Averaged Gain %</th>
<th>Reported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distillation = 1.9%</td>
<td>N / A</td>
</tr>
<tr>
<td>Single Projection = 9.3%</td>
<td></td>
</tr>
<tr>
<td>Distillation + Projection = 8.9%</td>
<td></td>
</tr>
</tbody>
</table>

Again, a **single** projection at test time is sufficient!
Our Conclusions

1) Distilled model better than single projection

2) Distilled neural network has significant gain on SST2 as it learns A-but-B rule

1) A single projection is a good way to explicitly encode logic rules

2) Distilled neural nets aren't learning logic rules
## Outline

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit</td>
<td>Hu et al. (ACL 2016)</td>
<td><strong>Contribution #1</strong> replication study finds <strong>incorrect</strong> conclusions</td>
</tr>
<tr>
<td>Implicit?</td>
<td>Peters et al. (NAACL 2018)</td>
<td><strong>Contribution #2</strong> ELMo embeddings learn logic rules <strong>without</strong> explicit supervision</td>
</tr>
</tbody>
</table>

**Contribution #3**
Robustness of explicit / implicit methods to varying annotator agreement in A-but-B sentences
ELMo Representations

Embeddings from Language Models
ELMo Representations

Embeddings from Language Models

large language model trained on the 1 Billion Words dataset
ELMo Representations

Embeddings from Language Models

large language model trained on the 1 Billion Words dataset

learnt representations used for downstream task
ELMo Representations

Embeddings from Language Models

large language model trained on the 1 Billion Words dataset

learnt representations used for downstream task

Unlike word2vec, these embeddings are contextual
ELMo Results (100 seeds)
ELMo Results (100 seeds)

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
<th>A-but-B + negation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN (Baseline)</td>
<td>86.0</td>
<td>78.7</td>
<td>80.1</td>
</tr>
<tr>
<td>CNN + ELMo</td>
<td>88.9</td>
<td>86.5</td>
<td>87.2</td>
</tr>
<tr>
<td>Gain %</td>
<td>2.9</td>
<td>7.8</td>
<td>7.1</td>
</tr>
</tbody>
</table>
## ELMo Results (100 seeds)

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
<th>A-but-B + negation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN (Baseline)</td>
<td>86.0</td>
<td>78.7</td>
<td>80.1</td>
</tr>
<tr>
<td>CNN + ELMo</td>
<td>88.9</td>
<td>86.5</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td><strong>2.9</strong></td>
<td><strong>7.8</strong></td>
<td><strong>7.1</strong></td>
</tr>
</tbody>
</table>

**Significant** improvement, *even after averaging!*
Is ELMo Learning Logic Rules?

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
<th>A-but-B + negation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN (Baseline)</td>
<td>86.0</td>
<td>78.7</td>
<td>80.1</td>
</tr>
<tr>
<td>CNN + ELMo</td>
<td>88.9</td>
<td>86.5</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td><strong>2.9</strong></td>
<td><strong>7.8</strong></td>
<td><strong>7.1</strong></td>
</tr>
</tbody>
</table>

60% of the improvement is on A-but-B sentences and negations.
Is ELMo Learning Logic Rules?

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
<th>A-but-B + negation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN (Baseline)</td>
<td>86.0</td>
<td>78.7</td>
<td>80.1</td>
</tr>
<tr>
<td>CNN + ELMo</td>
<td>88.9</td>
<td>86.5</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td>2.9</td>
<td>7.8</td>
<td>7.1</td>
</tr>
</tbody>
</table>

60% of the improvement is on A-but-B sentences and negations

(Only 24.5% of corpus is A-but-B / negations)
ELMo + Explicit (Projection)

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELMo</td>
<td>88.9</td>
<td>86.5</td>
</tr>
<tr>
<td>ELMo + project</td>
<td>89.0</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td><strong>0.1</strong></td>
<td><strong>0.7</strong></td>
</tr>
</tbody>
</table>
ELMo + Explicit (Projection)

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELMo</td>
<td>88.9</td>
<td>86.5</td>
</tr>
<tr>
<td>ELMo + project</td>
<td>89.0</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td>0.1</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Test-time projection is **ineffective** for ELMo
ELMo + Explicit (Projection)

<table>
<thead>
<tr>
<th>Model</th>
<th>SST2</th>
<th>A-but-B</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELMo</td>
<td>88.9</td>
<td>86.5</td>
</tr>
<tr>
<td>ELMo + project</td>
<td>89.0</td>
<td>87.2</td>
</tr>
<tr>
<td><strong>Gain %</strong></td>
<td>0.1</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Test-time projection is **ineffective** for ELMo

Distance between ELMo distribution and projected distribution is **0.13** (vs **0.26** distillation, **0.27** baseline)
Clustering ELMo Vectors

Cosine similarity between every pair of words
Clustering ELMo Vectors

Cosine similarity between every pair of words

_**Contrastive** (A-but-B)_

there are slow and repetitive parts, _but_ it has just enough spice to keep it interesting
there are slow and repetitive parts, but it has just enough spice to keep it interesting
there are slow and repetitive parts, but it has just enough spice to keep it interesting
there are slow and repetitive parts, but it has just enough spice to keep it interesting

**word2vec**

**ELMo**

**Clustering** for A part and B part in A-*but*-B sentences for ELMo embeddings
there are slow and repetitive parts, **but it has just enough spice to keep it interesting**
ELMo Representations learn the scope of a contrastive conjunction!
# Outline

<table>
<thead>
<tr>
<th>Method</th>
<th>Previous Work</th>
<th>Our Contributions</th>
</tr>
</thead>
</table>
| Explicit  | Hu et al. (ACL 2016)   | **Contribution #1**
replication study finds
**incorrect** conclusions |
| Implicit? | Peters et al. (NAACL 2018) | **Contribution #2**
ELMo embeddings
learn logic rules **without**
explicit supervision |

**Contribution #3**
Robustness of explicit / implicit methods to varying annotator agreement in A-but-B sentences
Sentiment is Ambiguous!

beautiful film, but those who have read the book will be disappointed
Sentiment is Ambiguous!

beautiful film, but those who have read the book will be disappointed

nine crowd-workers label each A-but-B sentence as positive / negative / neutral
Sentiment is Ambiguous!

beautiful film, but those who have read the book will be disappointed

nine crowd-workers label each A-but-B sentence as positive / negative / neutral

we test our models on subsets of varying agreement
Consistent trends on all levels of agreement
Projection degrades accuracy on high agreement sentences!
Key Takeaways

• Carefully perform sentiment classification research

• variation across runs - average across several seeds

• ambiguous sentences - benchmark on subsets of varying annotator agreement

• ELMo embeddings *implicitly* learn logic rules for sentiment classification

Code + Data

github.com/martiansideofthemoon/logic-rules-sentiment
Key Takeaways

• Carefully perform sentiment classification research

• variation across runs - average across several seeds

• ambiguous sentences - benchmark on subsets of varying annotator agreement

• ELMo embeddings implicitly learn logic rules for sentiment classification

Code + Data

github.com/martiansideofthemoon/logic-rules-sentiment