Saliency-driven Word Alignment Interpretation for NMT

Shuoyang Ding    Hainan Xu    Philipp Koehn

The Fourth Conference on Machine Translation
Florence, Italy
August 1st, 2019
Revisiting Six Challenges

• poor out-of-domain performance
• poor low-resource performance
• low frequency words
• long sentences
• attention is not word alignment
• large beam does not help

[Koehn and Knowles 2017]
Revisiting Six Challenges

- poor out-of-domain performance
- poor low-resource performance
- low frequency words
- long sentences
- **attention is not word alignment**
- large beam does not help

[Koehn and Knowles 2017]
We do not believe that we should.

Wir glauben nicht, daß wir nur rosinen herauspicken sollen.

We believe not, that we only raisin pick should.
We do not believe that we should.

Wir glauben nicht, daß wir nur rosinen herauspicken sollen.

We believe not, that we only raisin pick should.
Related Findings Outside MT

• “Attention is not Explanation” [Jain and Wallace NAACL 2019]
• “Is Attention Interpretable?” (Spoiler: No) [Serrano and Smith ACL 2019]
• We also have empirical results that corroborate these findings.
• ... and we have method that works better!
Saliency:
Identifying Important Features
Recap

We do not believe that we should.

Wir glauben nicht, dass wir nur Rosinen herauspicken sollten.

We believe not, that we only raisin pick should.
Recap

We do not believe that we should
Focus on solten

We do not believe that we should

A Great NMT Model
Perturbation

We do not believe that we should.

A Great NMT Model

$\Delta x$
Perturbation

We do not believe that we...
The output score is more sensitive to perturbations in important features.
E.g.

We do not believe that we **should**

Wir glauben nicht, daß wir nur rosinen herauspicken sollten.

We believe not, that we only raisin pick should.
E.g.

We do not believe that we should.

A Great NMT Model

Sie glauben nicht, daß wir nur rosinen herauspicken sollten.

They believe not, that we only raisin pick should.
E.g.

We do not believe that we will.

Wir glauben nicht, daß wir nur rosen herauspicken werden.

We believe not, that we only raisin pick will.
Saliency

\[
\frac{\Delta y}{\Delta x}
\]
Saliency

\[
\frac{\Delta y}{\Delta x}
\]

when \( \Delta x \to 0 \):

\[
\frac{\Delta y}{\Delta x} \to \frac{\partial y}{\partial x}
\]
Saliency

\[ \frac{\partial y}{\partial x} \]
What’s good about this?

1. Derivatives are easy to obtain for any DL toolkit
2. Model-agnostic
3. Adapts with the choice of output words
Prior Work on Saliency

• Widely used and studied in Computer Vision!
  [Simonyan et al. 2013][Springenberg et al. 2014]
  [Smilkov et al. 2017]

• Also in a few NLP work for qualitative analysis
  [Aubakirova and Bansal 2016][Li et al. 2016][Ding et al. 2017]
SmoothGrad

- Gradients are very local measure of sensitivity.
- Highly non-linear models may have pathological points where the gradients are noisy.
- Solution: calculate saliency for multiple copies of the same input corrupted with gaussian noise, and average the saliency of copies.

[Smilkov et al. 2017]
Establishing Saliency for Words
“Feature” in Computer Vision

Photo Credit: Hainan Xu
It’s straight-forward to compute saliency for a single dimension of the word embedding.
“Feature” in NLP

But how to \textit{compose} the saliency of each dimension into the saliency of a \textit{word}?
Consider word embedding look-up as a dot product between the embedding matrix and an one-hot vector.
Our Proposal

The 1 in the one-hot vector denotes the identity of the input word.
Let’s perturb that 1 like a real value! i.e. **take gradients** with regard to the 1.
Our Proposal

\[
\sum_i e_i \cdot \frac{\partial y}{\partial e_i}
\]

range: \((−\infty, \infty)\)
Experiment
Evaluation

• Evaluation of interpretations is tricky!

• Fortunately, there’s human judgments to rely on.

• Need to do force decoding with NMT model.
Setup

- **Architecture**: Convolutional S2S, LSTM, **Transformer** (with fairseq default hyper-parameters)

- **Dataset**: Following Zenkel et al. [2019], which covers **de-en**, **fr-en** and **ro-en**.

- **SmoothGrad hyper-parameters**: $N=30$ and $\sigma=0.15$
Baselines

- **Attention weights**

- **Smoothed Attention**: forward pass on multiple corrupted input samples, then average the attention weights over samples

- **[Li et al. 2016]**: compute element-wise absolute value of embedding gradients, then average over embedding dimensions

- **[Li et al. 2016] + SmoothGrad**
Convolutional S2S on de-en

![Bar chart showing AER for different methods.](chart.png)
Attention on de-en

![Graph showing AER values for different models: Conv, LSTM, Transformer, fast-align, Zenkel et al. [2019], and GIZA++.

- Transformer has the highest AER value.
- Conv and LSTM have similar AER values.
- fast-align, Zenkel et al. [2019], and GIZA++ have lower AER values with GIZA++ having the lowest.]

Saliency-driven Word Alignment Interpretation for NMT
Ours+SmoothGrad on de-en

Saliency-driven Word Alignment Interpretation for NMT
Li vs. Ours

(a) Attention

(b) Li

(c) Ours
Li vs. Ours

(a) Attention

(b) Li

(c) Ours
Conclusion
Conclusion

• Saliency + proper word-level score formulation is a better interpretation method than attention

• NMT models do learn interpretable alignments. We just need to properly uncover them!

[QR Code for Paper, Code, Slides]

https://github.com/shuoyangd/meerkat