Sparse and Constrained Attention for Neural Machine Translation

Chaitanya Malaviya\textsuperscript{1}, Pedro Ferreira\textsuperscript{2}, André F.T. Martins\textsuperscript{2,3}

\textsuperscript{1}Carnegie Mellon University, \textsuperscript{2}Instituto Superior Técnico, \textsuperscript{3}Unbabel
Adequacy in Neural Machine Translation

Source: und wir benutzen dieses wort mit solcher verachtung

Reference: and we say that word with such contempt.

Translation: and we use this word with such contempt.

Replications

Source: Ein 28-jähriger Koch, der kürzlich nach Pittsburgh gezogen war, wurde diese Woche im Treppenhaus eines örtlichen Einkaufszentrums tot aufgefunden.

Reference: A 28-year-old chef who recently moved to Pittsburgh was found dead in the staircase of a local shopping mall this week.

Translation: A 28-year-old chef who recently moved to Pittsburgh was found dead in the staircase this week.

Dropped words

Reference: Pittsburgh was found dead in the staircase of a local shopping mall this week.

Translation: Pittsburgh was found dead in the staircase this week.
Previous Work

• Conditioning on coverage vectors to track attention history (Mi, 2016; Tu, 2016).

• Gating architectures and adaptive attention to control amount of source context (Tu, 2017; Li & Zhu, 2017).

• Reconstruction Loss (Tu, 2017).

• Coverage penalty during decoding (Wu, 2016).
Main Contributions

1. Fertility-based Neural Machine Translation Model (Bounds on source attention weights)

2. Novel attention transform function: *Constrained Sparsemax* (Enforces these bounds)

3. Evaluation Metrics: REP-Score and DROP-Score
NMT + Attention Architecture
$$p_t = \text{softmax}(W[g_t; c_t])$$

$$c_t = H \alpha_t$$

**attn_score:**
- dot-product (Luong, 2015)
- bilinear function
- MLP (Bahdanau, 2014)

**attn_transform:**
- traditional softmax
- constrained softmax (Martins & Kreutzer, 2017)
- sparsemax (Martins & Astudillo, 2016)
- **constrained sparsemax** (this work)
Attention Transform Functions

• Sparsemax: Euclidean projection of $z$ provides sparse probability distributions.

$$\text{sparsemax}(z) := \arg \min_{\alpha \in \Delta^J} \| \alpha - z \|^2$$

• Constrained Softmax: Returns the distribution closest to softmax whose attention probabilities are bounded by upper bounds $u$.

$$\text{csoftmax}(z; u) := \arg \min_{\alpha \in \Delta^J} \text{KL}(\alpha \| \text{softmax}(z))$$

s.t. $\alpha \leq u$
Attention Transform Functions

• Sparsemax: Euclidean projection of z provides sparse probability distributions.

\[
\text{sparsemax}(z) := \arg \min_{\alpha \in \Delta^J} \| \alpha - z \|_2^2
\]

Sparse and Constrained?

• Constrained Softmax: Returns the distribution closest to softmax whose attention probabilities are bounded by upper bounds \( u \).

\[
\text{csoftmax}(z; u) := \arg \min_{\alpha \in \Delta^J} \text{KL}(\alpha \| \text{softmax}(z))
\]

\[
\text{s.t. } \alpha \leq u
\]
Constrained Sparsemax

- Provides sparse and bounded probability distributions.

\[
\text{csparsemax}(z; u) := \arg \min_{\alpha \in \Delta^J} \|\alpha - z\|^2 \\
\text{s.t. } \alpha \leq u.
\]

- This transformation has two levels of sparsity: over time steps & over attended words at each step.

- Efficient linear and sublinear time algorithms for forward and backward propagation.
• csparsemax provides sparse and constrained probabilities.
Fertility-based NMT Model
Fertility-based NMT

• Allocate fertilities $f$ for each source word as attention budgets that exhaust over decoding.

• Fertility Predictor: Train biLSTM model supervised by fertilities from fast_align (IBM Model 2).
Fertility-based NMT

• Fertilities incorporated as:

\[ \alpha_t = \text{csparsemax}(z_t, f - \beta_{t-1}) \]

\[ \beta_{t-1} := \sum_{\tau=1}^{t-1} \alpha_\tau \]

• Exhaustion strategy to encourage more attention for words with larger credit remaining:

\[ z'_t = z_t + cu_t \]
Experiments
Experiments

• Experiments performed on 3 language pairs: De-En (IWSLT 2014), Ro-En (Europarl), Ja-En (KFTT).

• Joint BPE with 32K merge operations.

• Default hyperparameter settings in OpenNMT-Py.

• Baselines: \textit{Softmax}, + \textit{CovPenalty} (Wu, 2016) and + \textit{CovVector} (Tu, 2016)
Evaluation Metrics: REP-Score & DROP-Score

REP Score:

• Penalizes n-gram repetitions in predicted translations.
• Normalize by number of words in reference corpus.

DROP Score:

• Find word alignments from source to reference & source to predicted.
• % of source words aligned with some word in reference, but not with any word in predicted translation.
Results

BLEU Scores

- **De-En**
  - softmax: 29.51
  - softmax+CovPenalty: 29.69
  - softmax+CovVector: 29.85

- **Ja-En**
  - softmax: 20.36
  - softmax+CovPenalty: 20.7
  - softmax+CovVector: 21.53
  - csparsemax: 21.31

- **Ro-En**
  - softmax: 29.67
  - softmax+CovPenalty: 29.81
  - softmax+CovVector: 30.08
  - csparsemax: 29.77
csparsemax yields sparse set of alignments and avoids repetitions.

softmax  csparsemax
Examples of Translations
<table>
<thead>
<tr>
<th>input</th>
<th>überlassen sie das ruhig uns.</th>
</tr>
</thead>
<tbody>
<tr>
<td>reference</td>
<td>leave that up to us.</td>
</tr>
<tr>
<td>softmax</td>
<td>give us a silence.</td>
</tr>
<tr>
<td>csparsemax</td>
<td>leave it to us.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>input</th>
<th>so ungefähr, sie wissen schon.</th>
</tr>
</thead>
<tbody>
<tr>
<td>reference</td>
<td>like that, you know.</td>
</tr>
<tr>
<td>softmax</td>
<td>so, you know, you know.</td>
</tr>
<tr>
<td>csparsemax</td>
<td>like that, you know.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>input</th>
<th>wir sehen das dazu, dass phosphor wirklich kritisch ist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>reference</td>
<td>we can see that phosphorus is really critical.</td>
</tr>
<tr>
<td>softmax</td>
<td>we see that that phosphorus is really critical.</td>
</tr>
<tr>
<td>csparsemax</td>
<td>we see that phosphorus is really critical.</td>
</tr>
</tbody>
</table>

More in the paper…
Thank You!

**Code**: www.github.com/Unbabel/sparse_constrained_attention

Questions?