Beyond Weight Tying: Learning Joint Input-Output Embeddings for Neural Machine Translation

Nikolaos Pappas\textsuperscript{1}, Lesly Miculicich\textsuperscript{1,2}, James Henderson\textsuperscript{1}

\textsuperscript{1}Idiap Research Institute, Switzerland
\textsuperscript{2}École polytechnique fédérale de Lausanne (EPFL)

October 31, 2018
Output layer parametrization

- NMT systems predict one word at a time given context $h_t \in \mathbb{R}^{d_h}$, weights $W \in \mathbb{R}^{d_h \times |\mathcal{V}|}$ and bias $b \in \mathbb{R}^{|\mathcal{V}|}$ by modeling:

$$p(y_t|Y_{1:t-1}, X) \propto \exp(W^T h_t + b)$$

- Parametrization depends on the vocabulary ($C_{base} = |\mathcal{V}| \times d_h + |\mathcal{V}|$) which creates training and out-of-vocabulary word issues
  - sub-word level modeling (Sennrich et al., 2016)
  - output layer approximations (Mikolov et al., 2013)
  - weight tying (Press & Wolf, 2017)

→ Lack of semantic grounding and composition of output representations
Weight tying

- Shares target embedding $E \in \mathbb{R}^{|\mathcal{V}| \times d}$ with $\mathcal{W}$ (Press & Wolf, 2017):
  $$p(y_t|Y_{1:t-1}, X) \propto \exp(Eh_t + b)$$
  - Parametrization depends less on the vocabulary ($C_{tied} = |\mathcal{V}|$).

- Assuming that bias is zero and $E$ learns linear word relationships implicitly ($E \approx E_l \mathcal{W}$) (Mikolov et al., 2013):
  $$p(y_t|Y_{1:t-1}, X) \propto \exp(E_l \mathcal{W} h_t)$$
  - Equivalent to bilinear form of zero-shot models (Nam et al., 2016).

→ Imposes implicit linear structure on the output
→ This could explain its sample efficiency and effectiveness
Zero-shot models

- Learn a joint input-output space with a bilinear form given weight matrix $W \in \mathbb{R}^{d \times d_h}$ (Socher et al., 2013, Nam et al., 2016):

$$g(E, h_t) = E W h_t$$

- Useful properties
  - Grounding outputs to word descriptions and semantics
  - Explicit output relationships or structure ($C_{bilinear} = d \times d_h + |V|$)
  - Knowledge transfer across outputs especially low-resource ones
Examples of learned structure

<table>
<thead>
<tr>
<th>Query</th>
<th>NMT Input</th>
<th>NMT Output</th>
<th>NMT-tied Input/Output</th>
<th>Ours Input</th>
<th>Ours Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>visited</td>
<td>attacked</td>
<td>visiting</td>
<td>visits</td>
<td>visiting</td>
<td>attended</td>
</tr>
<tr>
<td>(Verb past tense)</td>
<td>conquered</td>
<td>attended</td>
<td>attended</td>
<td>attended</td>
<td>witnessed</td>
</tr>
<tr>
<td></td>
<td>contacted</td>
<td>visit</td>
<td>visiting</td>
<td>visits</td>
<td>discussed</td>
</tr>
<tr>
<td></td>
<td>occupied</td>
<td>visits</td>
<td>frequented</td>
<td>visit</td>
<td>recognized</td>
</tr>
<tr>
<td></td>
<td>consulted</td>
<td>discovered</td>
<td>frequented</td>
<td>frequented</td>
<td>demonstrated</td>
</tr>
<tr>
<td>generous</td>
<td>modest</td>
<td>spacious</td>
<td>generosity</td>
<td>spacious</td>
<td>friendly</td>
</tr>
<tr>
<td>(Adjective)</td>
<td>extensive</td>
<td>generosity</td>
<td>generously</td>
<td>generosity</td>
<td>flexible</td>
</tr>
<tr>
<td></td>
<td>substantial</td>
<td>generously</td>
<td>generously</td>
<td>flexible</td>
<td>brilliant</td>
</tr>
<tr>
<td></td>
<td>ambitious</td>
<td>massive</td>
<td>lavish</td>
<td>generously</td>
<td>fantastic</td>
</tr>
<tr>
<td></td>
<td>sumptuous</td>
<td>huge</td>
<td>massive</td>
<td>massive</td>
<td>massive</td>
</tr>
<tr>
<td>friend</td>
<td>wife</td>
<td>friends</td>
<td>colleague</td>
<td>colleague</td>
<td>colleague</td>
</tr>
<tr>
<td>(Noun)</td>
<td>husband</td>
<td>colleague</td>
<td>friends</td>
<td>friends</td>
<td>fellow</td>
</tr>
<tr>
<td></td>
<td>colleague</td>
<td>Fri@@</td>
<td>neighbour</td>
<td>neighbour</td>
<td>supporter</td>
</tr>
<tr>
<td></td>
<td>friends</td>
<td>fellow</td>
<td>girlfriend</td>
<td>girlfriend</td>
<td>partner</td>
</tr>
<tr>
<td></td>
<td>painter</td>
<td>friendship</td>
<td>companion</td>
<td>husband</td>
<td>manager</td>
</tr>
</tbody>
</table>

Top-5 most similar words based on cosine distance. Inconsistent words are marked in red.
Contributions

- Learning explicit non-linear output and context relationships
  - New family of joint space models that generalize weight tying
  
  \[ g(E, h_t) = g_{out}(E) \cdot g_{inp}(h_t) \]

- Flexibly controlling effective capacity
  - Two extremes can lead to under or overparametrized output layer

\[ C_{tied} < C_{bilinear} \leq C_{joint} \leq C_{base} \]

→ Identify key limitations in existing output layer parametrizations
→ Propose a joint input-output model which addresses them
→ Provide empirical evidence of its effectiveness
Introduction
  Background
  Motivation

Proposed Output Layer
  Joint Input-Output Embedding
  Unique properties
  Scaling Computation

Evaluation
  Data and Settings
  Quantitative Results

Conclusion
Joint input-output embedding

- Two non-linear projections with $d_j$ dimensions of any context $h_t$ and output in $E$:

  \[
  g_{out}(E) = \sigma(UE^T + b_u)
  \]

  \[
  g_{inp}(h_t) = \sigma(Vh_t + b_v)
  \]

- The conditional distribution becomes:

  \[
  p(y_t|Y_{1:t-1}, X) \propto \exp(g_{out}(E) \cdot g_{inp}(h_t) + b)
  \]

  \[
  \propto \exp(\sigma(UE^T + b_u) \cdot \sigma(Vh_t + b_v) + b)
  \]

  Output struct.  Context struct.
Unique properties

1. Learns explicit non-linear output and context structure
2. Allows to control capacity freely by modifying $d_j$
3. Generalizes the notion of weight tying
   - Weight tying emerges as a special case by setting $g_{inp}(\cdot), g_{out}(\cdot)$ to the identity function $I$:
     \[
p(y_t | Y_{1:t-1}, X) \propto \exp(g_{out}(E) \cdot g_{inp}(h_t) + b) \\
     \propto \exp((lE)(lh_t) + b) \\
     \propto \exp(Eh_t + b)
     \]

Scaling computation

- Prohibitive for a large vocabulary or joint space: $U \cdot E^T$
- Sampling-based training which uses a subset of $\mathcal{V}$ to compute softmax (Mikolov et al., 2013)

<table>
<thead>
<tr>
<th>Model</th>
<th>$d_j$</th>
<th>50%</th>
<th>25%</th>
<th>5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMT</td>
<td>-</td>
<td>4.3K</td>
<td>5.7K</td>
<td>7.1K</td>
</tr>
<tr>
<td>NMT-tied</td>
<td>-</td>
<td>5.2K</td>
<td>6.0K</td>
<td>7.8K</td>
</tr>
<tr>
<td>NMT-joint</td>
<td>512</td>
<td>4.9K</td>
<td>5.9K</td>
<td>7.2K</td>
</tr>
<tr>
<td>NMT-joint</td>
<td>2048</td>
<td>2.8K</td>
<td>4.2K</td>
<td>7.0K</td>
</tr>
<tr>
<td>NMT-joint</td>
<td>4096</td>
<td>1.7K</td>
<td>2.9K</td>
<td>6.0K</td>
</tr>
</tbody>
</table>

Target tokens per second on English-German, $|\mathcal{V}| \approx 128K$. 
Introduction
  Background
  Motivation

Proposed Output Layer
  Joint Input-Output Embedding
  Unique properties
  Scaling Computation

Evaluation
  Data and Settings
  Quantitative Results

Conclusion
Data and settings

Controlled experiments with LSTM sequence-to-sequence models

- English-Finish (2.5M), English-German (5.8M) from WMT
- Morphologically rich and poor languages as target
- Different vocabulary sizes using BPE: 32K, 64K, ~128K

Baselines

- NMT: softmax + linear unit
- NMT-tied: softmax + linear unit + weight tying

| Input: 512, Depth: 2-layer, 512, Attention: 512, Joint dim.: 512, 2048, 4096, Joint act.: Tanh, Optimizer: ADAM, Dropout: 0.3, Batch size: 96, Metrics: BLEU, METEOR |
Translation performance

- Weight tying is as good as the baseline but not always
- Joint model has more consistent improvements
Evaluation Quantitative Results

Translation performance by output frequency

<table>
<thead>
<tr>
<th>Output frequency bins</th>
<th>METEOR (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>30</td>
</tr>
<tr>
<td>medium</td>
<td>25</td>
</tr>
<tr>
<td>low</td>
<td>20</td>
</tr>
</tbody>
</table>

English-German and German-English, $|\mathcal{V}| \approx 32K$.

- Vocabulary is split in three sets of decreasing frequency
- Joint model transfers knowledge across high and lower-resource bins
Do we need to learn both output and context structure?

Ablation results show that both are essential.

German-English, $|\mathcal{V}| \approx 32K$. 
What is the effect of increasing the output layer capacity?

Varying joint space dimension ($d_j$), $|\mathcal{V}| \approx 32K$.

- Higher capacity was helpful in most cases.
Conclusion

• Joint space models generalize weight tying and have more robust results against baseline overall
• Learn explicit non-linear output and context structure
• Provide flexible way to control capacity

Future work:

→ Use crosslingual, contextualized or descriptive representations
→ Evaluate on multi-task and zero-resource settings
→ Find more efficient ways to increase output layer capacity
Thank you! Questions?

http://github.com/idiap/joint-embedding-nmt

Acknowledgments