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

Nikolaos Pappas[1], Lesly Miculicich[1,2], James Henderson[1]

[1]Idiap Research Institute, Switzerland
[2]École polytechnique fédérale de Lausanne (EPFL)

October 31, 2018

Output layer parametrization

- NMT systems predict one word at a time given context $h_t \in \mathrm{I\!R}^{\mathrm{d_h}}$, weights $W \in \mathrm{I\!R}^{\mathrm{d_h} \times |\mathcal{V}|}$ and bias $b \in \mathrm{I\!R}^{|\mathcal{V}|}$ 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)

$\rightarrow$ Lack of semantic grounding and composition of output representations

Weight tying

- Shares target embedding $E \in \mathbb{R}^{|\mathcal{V}| \times \mathrm{d}}$ with 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).

$\rightarrow$ Imposes implicit linear structure on the output

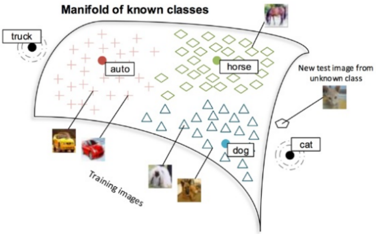
$\rightarrow$ This could explain its sample efficiency and effectiveness

Zero-shot models

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

$$g(E, h_t) = E \underbrace{\mathcal{W}}_{Structure} h_t$$

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



Examples of learned structure

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

Top-5 most similar words based on cosine distance.
Incosistent 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

$$\mathcal{C}_{tied} < \mathcal{C}_{bilinear} \leq \mathcal{C}_{joint} \leq \mathcal{C}_{base}$$

$\rightarrow$ Identify key limitations in existing output layer parametrizations

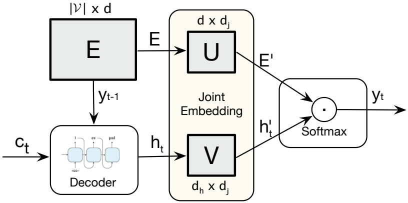
$\rightarrow$ Propose a joint input-output model which addresses them

$\rightarrow$ Provide empirical evidence of its effectiveness

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:

$$p(y_t|Y_{1:t-1}, X) \propto \exp\big(g_{out}(E) \cdot g_{inp}(h_t) + b\big)$$
$$\propto \exp\big(\underbrace{\sigma(UE^T + b_u)}_{\text{Output struct.}} \cdot \underbrace{\sigma(Vh_t + b_v)}_{\text{Context struct.}} + b\big)$$

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:

$$\begin{aligned} p(y_t|Y_{1:t-1}, X) &\propto \exp\big(g_{out}(E) \cdot g_{inp}(h_t) + b\big) \\ &\propto \exp\big((IE)\ (Ih_t) + b\big) \\ &\propto \exp\big(Eh_t + b\big)\ \square \end{aligned}$$

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)

Model	d_j	50%	25%	5%
NMT	-	4.3K	5.7K	7.1K
NMT-`tied`	-	5.2K	6.0K	7.8K
NMT-`joint`	512	4.9K	5.9K	7.2K
NMT-`joint`	2048	2.8K	4.2K	7.0K
NMT-`joint`	4096	1.7K	2.9K	6.0K

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

Data and settings

Controlled experiments with LSTM sequence-to-sequence models

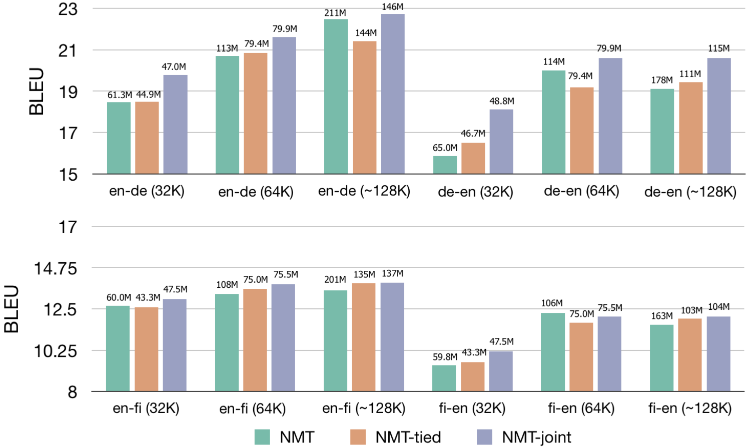
- English-Finish (2.5M), English-German (5.8M) from WMT
- Morphologically rich and poor languages as target
- Different vocabulary sizes using BPE: 32K, 64K, ∼128K

Baselines

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

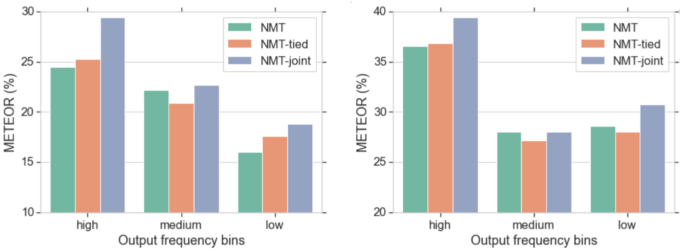
Input: 512, Depth: 2-layer, 512, Attention: 512, Joint dim.: 512, 2048, 4096, Joint act.: Tanh, Optimizer: ADAM, Dropout: 0.3, Batch size: 96 Metrics: BLEU, METEOR

Translation performance



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

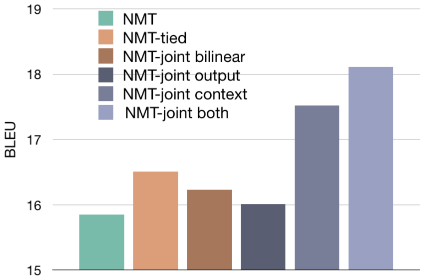
Translation performance by output frequency



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

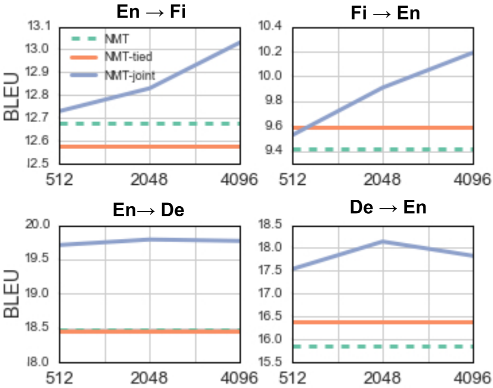
Do we need to learn both output and context structure?



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

- Ablation results show that both are essential.

What is the effect of increasing the output layer capacity?



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

- 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