Pre-train word embedding

Xing Wang

2018/07/23

- When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? (CMU)
- How Do Source-side Monolingual Word Embeddings Impact Neural Machine Translation? (Johns Hopkins University)
- Context-Attentive Embeddings for Improved Sentence Representations (Facebook AI Research)

- When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? (CMU)
- How Do Source-side Monolingual Word Embeddings Impact Neural Machine Translation? (Johns Hopkins University)
- Context-Attentive Embeddings for Improved Sentence Representations (Facebook AI Research)

When and Why

- use pretrained word embeddings for NMT in low resource scenarios
- it is still not clear as to when we can expect pre-trained embeddings to be useful in NMT, or why they provide performance improvements

• Experimental Setup

- Parallel data: a parallel corpus from TED talks transcripts
- Pre-trained word embeddings: use available pre-trained word embeddings (Bojanowski et al., 2016) trained using fastText on Wikipedia for each language.

Dataset	train	dev	test
GL → EN	10,017	682	1,007
PT → EN	51,785	1,193	1,803
AZ → EN	5,946	671	903
TR → EN	182,450	4,045	5,029
BE → EN	4,509	248	664
RU → EN	208,106	4,805	5,476

- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information.

When and Why

- Q1: Efficacy of Pre-training

Src → / → Trg	std / std	pre / std	std / pre	pre / pre
GL → EN	2.2	**13.2**	2.8	12.8
PT → EN	26.2	**30.3**	26.1	**30.8**
AZ → EN	1.3	**2.0**	1.6	**2.0**
TR → EN	14.9	17.6	14.7	**17.9**
BE → EN	1.6	2.5	1.3	**3.0**
RU → EN	18.5	**21.2**	18.7	**21.1**

When and Why

- Q1: Efficacy of Pre-training

Src → / → Trg	std std	pre std	std pre	pre pre
GL → EN	2.2	**13.2**	2.8	12.8
PT → EN	26.2	**30.3**	26.1	**30.8**
AZ → EN	1.3	**2.0**	1.6	**2.0**
TR → EN	14.9	17.6	14.7	**17.9**
BE → EN	1.6	2.5	1.3	**3.0**
RU → EN	18.5	**21.2**	18.7	**21.1**

- the majority of the gain from pre-trained word embeddings results from a better encoding of the source sentence.

When and Why

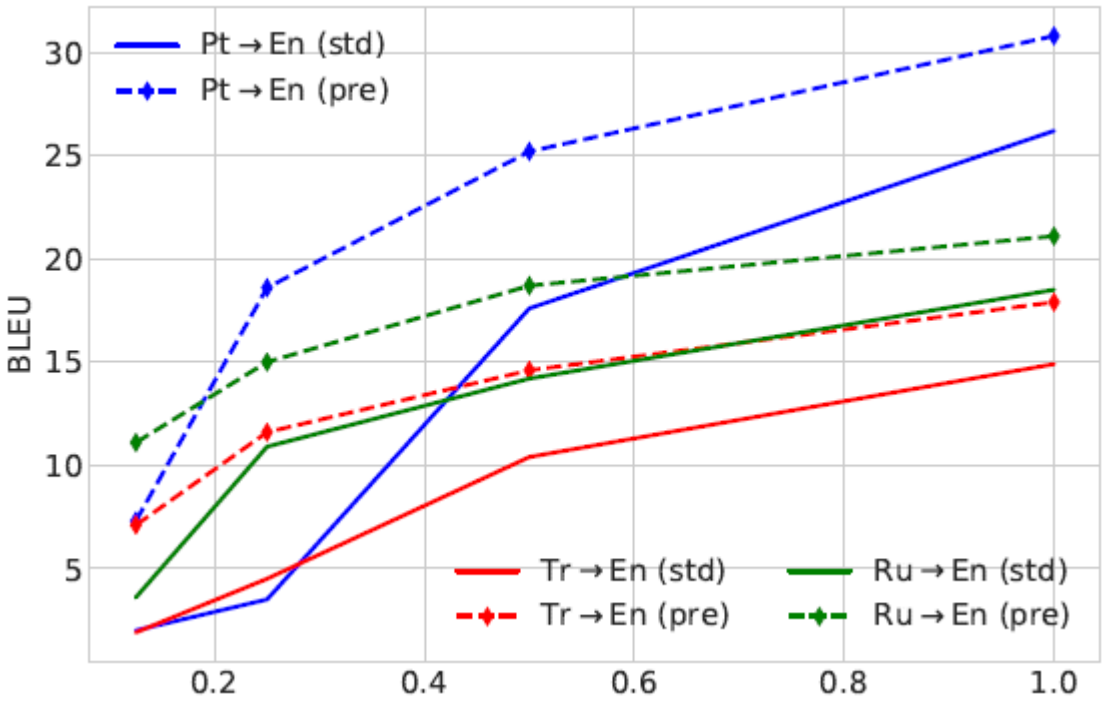
- Q1: Efficacy of Pre-training

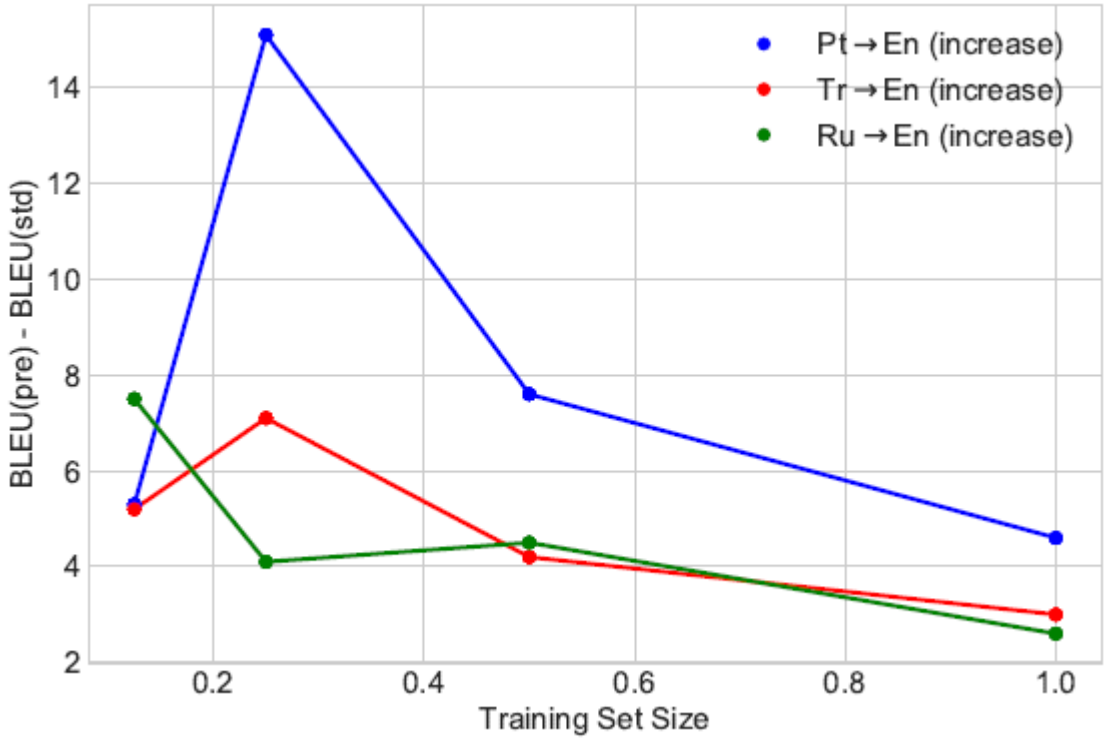
Src → / → Trg	std / std	pre / std	std / pre	pre / pre
GL → EN	2.2	**13.2**	2.8	12.8
PT → EN	26.2	**30.3**	26.1	**30.8**
AZ → EN	1.3	**2.0**	1.6	**2.0**
TR → EN	14.9	17.6	14.7	**17.9**
BE → EN	1.6	2.5	1.3	**3.0**
RU → EN	18.5	**21.2**	18.7	**21.1**

- word embeddings may be particularly useful to bootstrap models that are on the threshold of being able to produce reasonable translations

When and Why

- Q2: Effect of Training Data Size





When and Why

- Q3: Effect of Language Similarity

Dataset	Lang. Family	std	pre
ES → PT	West-Iberian	17.8	24.8 (+**7.0**)
FR → PT	Western Romance	12.4	18.1 (+5.7)
IT → PT	Romance	14.5	19.2 (+4.7)
RU → PT	Indo-European	2.4	8.6 (+6.2)
HE → PT	*No Common*	3.0	11.9 (+**8.9**)

When and Why

- Q4: Effect of Word Embedding Alignment

Dataset	unaligned	aligned
GL → EN	12.8	11.5 (−1.3)
PT → EN	30.8	30.6 (−0.2)
AZ → EN	2.0	2.1 (+0.1)
TR → EN	17.9	17.7 (−0.2)
BE → EN	3.0	3.0 (+0.0)
RU → EN	21.1	21.4 (+0.3)

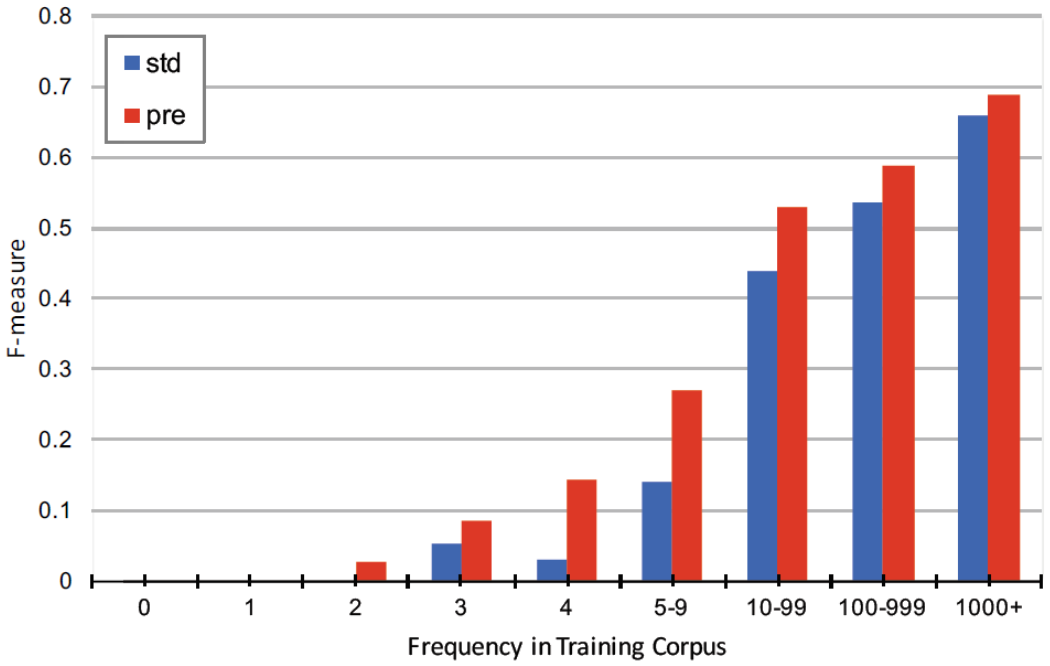
When and Why

- Q5: Effect of Multilinguality

Train	Eval	bi	std	pre	align
GL + PT	GL	2.2	17.5	20.8	**22.4**
AZ + TR	AZ	1.3	5.4	5.9	**7.5**
BE + RU	BE	1.6	**10.0**	7.9	9.6

When and Why

- Analysis: F-measure of Target Words



When and Why

- Conclusion:
- there is a sweet-spot where word embeddings are most effective, where there is very little training data but not so little that the system cannot be trained at all
- pre-trained embeddings seem to be more effective for more similar translation pairs
- a priori alignment of embeddings may not be necessary in bilingual scenarios, but is helpful in multi-lingual training scenarios

- When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? (CMU)

- How Do Source-side Monolingual Word Embeddings Impact Neural Machine Translation? (Johns Hopkins University)

- Context-Attentive Embeddings for Improved Sentence Representations (Facebook AI Research)

How

- use pre-trained source-side monolingual word embedding in NMT
- propose a novel strategy called dual-embedding that blends the fixing and updating strategies

Experimental Setup

- Parallel data: ZH-EN 2.1M / DE-EN WMT 2016

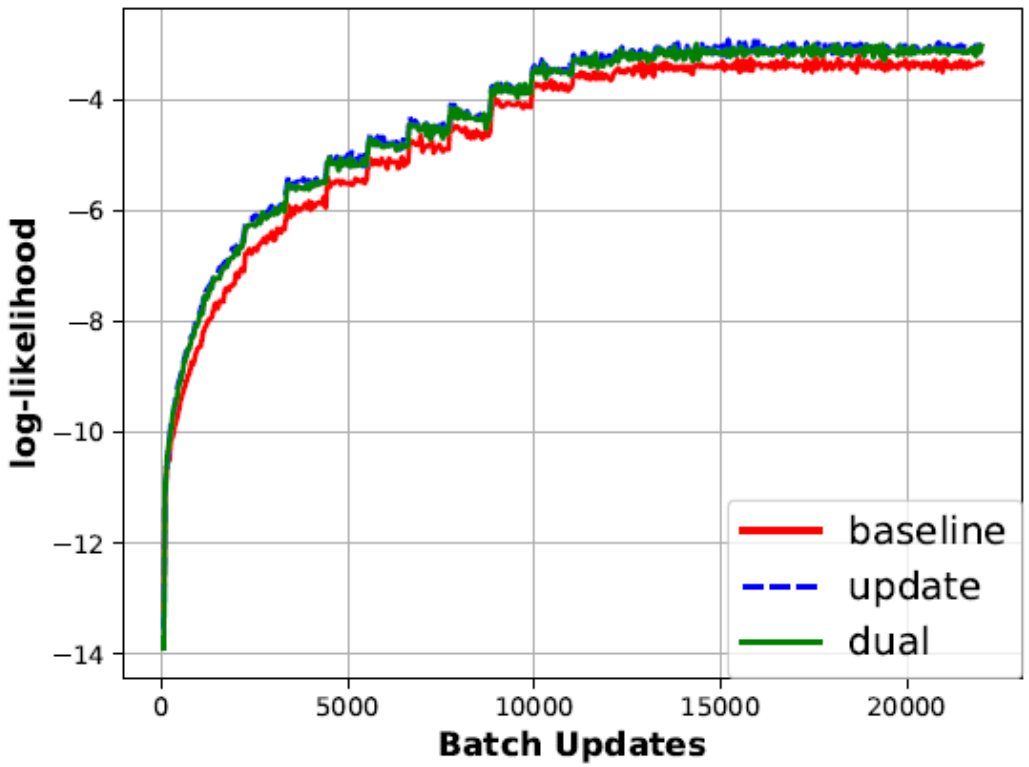
- Pre-trained word embeddings:
  - Data size:
    small
    extended

  - Different strategies:
    fixed initialization
    update initialization
    dual embedding

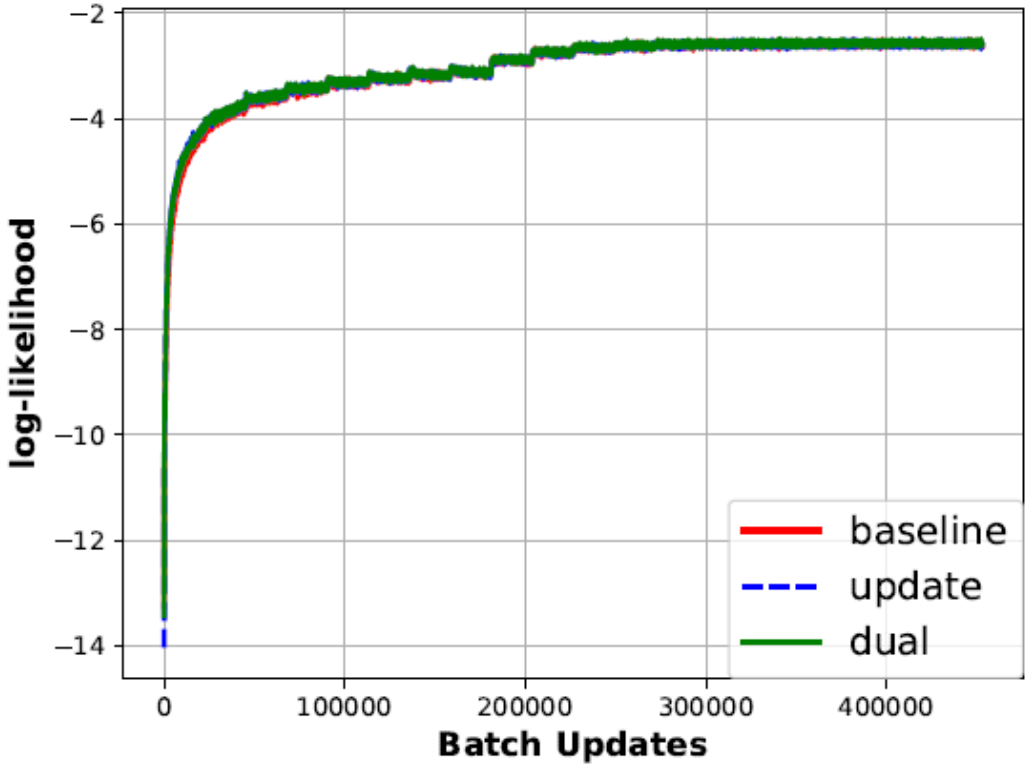
Sample Size	ZH-EN		DE-EN	
	Pre-BPE	BPE	Pre-BPE	BPE
100,000	15.1%	1.73%	24.0%	2.41%
250,000	10.4%	.809%	17.5%	.894%
500,000	8.49%	.667%	14.4%	.480%
1,000,000	6.21%	.398%	11.6%	.332%
Unsampled	5.11%	.284%	7.36%	.0923%

Table 1: Percentage of OOV Types in the Test Sets

How

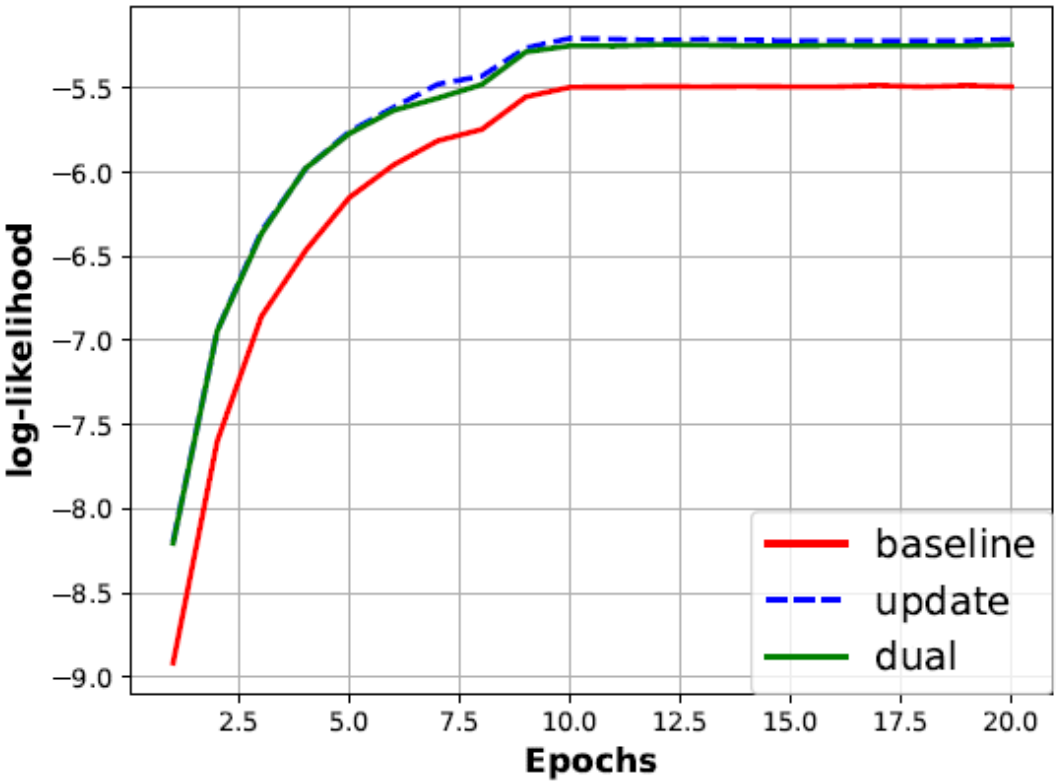


(a) Training Curve on 100k Sample

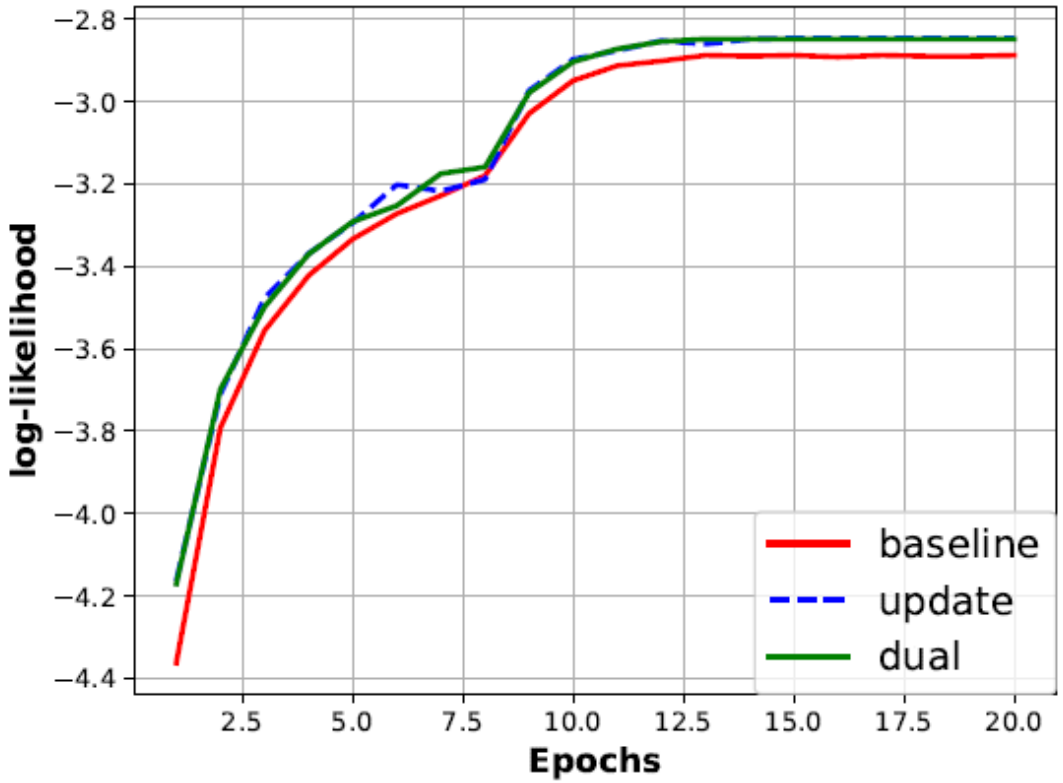


(b) Training Curve on Unsampled Data

How



(c) Development Curve on 100k Sample



(d) Development Curve on Unsampled Data

How

- Chinese-English experiment Results with Different Data Sizes

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	15.99	15.36†	16.44	**16.89**†	15.90	17.55†	**17.56**†
250,000	21.72	22.05	**22.16**	22.08	22.05	23.04†	**23.38**†
500,000	26.28	25.48†	**26.51**	26.29	26.13	**27.09**†	26.76†
1,000,000	29.75	29.54	29.59	**29.83**	29.39	30.6†	**30.89**†
2,013,142	32.44	31.40†	**33.21**†	**33.21**	32.61	33.26†	**34.03***†

How

- German-English experiment Results with Different Data Sizes

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	13.55	14.02†	14.75†	**14.91**†	15.14†	15.28†	**15.96***†
250,000	20.11	20.18	**20.92***†	20.47	20.74†	20.76†	**20.95**†
500,000	23.09	23.46	**24.14**†	24.11†	23.55†	24.14†	**24.64***†
1,000,000	26.17	25.95	**26.29**	26.21	26.27	26.19	**26.59***†
4,562,102	29.71	28.73†	30.00†	**30.07**	29.17†	29.13†	**30.00***

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	15.99	15.36†	16.44	**16.89**†	15.90	17.55†	**17.56**†
250,000	21.72	22.05	**22.16**	22.08	22.05	23.04†	**23.38**†
500,000	26.28	25.48†	**26.51**	26.29	26.13	**27.09**†	26.76†
1,000,000	29.75	29.54	29.59	**29.83**	29.39	30.6†	**30.89**†
2,013,142	32.44	31.40†	**33.21**†	**33.21**	32.61	33.26†	**34.03***†

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	13.55	14.02†	14.75†	**14.91**†	15.14†	15.28†	**15.96***†
250,000	20.11	20.18	**20.92***†	20.47	20.74†	20.76†	**20.95**†
500,000	23.09	23.46	**24.14**†	24.11†	23.55†	24.14†	**24.64***†
1,000,000	26.17	25.95	**26.29**	26.21	26.27	26.19	**26.59***†
4,562,102	29.71	28.73†	30.00†	**30.07**	29.17†	29.13†	**30.00***

- fixed initialization does not only almost consistently generate worst BLEU score among the three, but also significantly hurts the performance over the baseline sometimes.

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	15.99	15.36†	16.44	**16.89**†	15.90	17.55†	**17.56**†
250,000	21.72	22.05	**22.16**	22.08	22.05	23.04†	**23.38**†
500,000	26.28	25.48†	**26.51**	26.29	26.13	**27.09**†	26.76†
1,000,000	29.75	29.54	29.59	**29.83**	29.39	30.6†	**30.89**†
2,013,142	32.44	31.40†	**33.21**†	**33.21**	32.61	33.26†	**34.03***†

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	13.55	14.02†	14.75†	**14.91**†	15.14†	15.28†	**15.96***†
250,000	20.11	20.18	**20.92***†	20.47	20.74†	20.76†	**20.95**†
500,000	23.09	23.46	**24.14**†	24.11†	23.55†	24.14†	**24.64***†
1,000,000	26.17	25.95	**26.29**	26.21	26.27	26.19	**26.59***†
4,562,102	29.71	28.73†	30.00†	**30.07**	29.17†	29.13†	**30.00***

- the parallel training data has a minor domain mismatch from the monolingual training data (parliament proceedings vs. news)

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	15.99	15.36†	16.44	**16.89**†	15.90	17.55†	**17.56**†
250,000	21.72	22.05	**22.16**	22.08	22.05	23.04†	**23.38**†
500,000	26.28	25.48†	**26.51**	26.29	26.13	**27.09**†	26.76†
1,000,000	29.75	29.54	29.59	**29.83**	29.39	30.6†	**30.89**†
2,013,142	32.44	31.40†	**33.21**†	**33.21**	32.61	33.26†	**34.03***†

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	13.55	14.02†	14.75†	**14.91**†	15.14†	15.28†	**15.96***†
250,000	20.11	20.18	**20.92***†	20.47	20.74†	20.76†	**20.95**†
500,000	23.09	23.46	**24.14**†	24.11†	23.55†	24.14†	**24.64***†
1,000,000	26.17	25.95	**26.29**	26.21	26.27	26.19	**26.59***†
4,562,102	29.71	28.73†	30.00†	**30.07**	29.17†	29.13†	**30.00***

- dual embedding method is more robust to domain variances between monolingual data and bi-text.

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	15.99	15.36†	16.44	**16.89**†	15.90	17.55†	**17.56**†
250,000	21.72	22.05	**22.16**	22.08	22.05	23.04†	**23.38**†
500,000	26.28	25.48†	**26.51**	26.29	26.13	**27.09**†	26.76†
1,000,000	29.75	29.54	29.59	**29.83**	29.39	30.6†	**30.89**†
2,013,142	32.44	31.40†	**33.21**†	**33.21**	32.61	33.26†	**34.03***†

Sentence Pairs	Baseline	small embedding			extended embedding		
		fixed	update	dual	fixed	update	dual
100,000	13.55	14.02†	14.75†	**14.91**†	15.14†	15.28†	**15.96***†
250,000	20.11	20.18	**20.92***†	20.47	20.74†	20.76†	**20.95**†
500,000	23.09	23.46	**24.14**†	24.11†	23.55†	24.14†	**24.64***†
1,000,000	26.17	25.95	**26.29**	26.21	26.27	26.19	**26.59***†
4,562,102	29.71	28.73†	30.00†	**30.07**	29.17†	29.13†	**30.00***

- extra monolingual information is most useful under low-resource settings.
- dual embedding model is able to get the best of both worlds as expected.

How

- Qualitative Analysis

- In the case of extended embedding, however, we observed an specific improvement of the translation adequacy for rare words (mainly named entities) in the training data, and the usage of dual embedding model often brings further improvements in that aspect.

How

- are the singleton words appeared in the sentence translated correctly in the test output?

system	accuracy
baseline	29.10%
update	32.09%
dual	**33.58%**

Table 5: Human-evaluated singleton word translation accuracies on Chinese-English test set.

How

- Conclusion:
- the source-side embeddings should be updated during NMT training
- the source-side embeddings are more effective when bilingual training data is limited, especially when OOV rates is high. Moreover, source-side embedding incorporation is also useful under some high-resource settings when incorporated properly.
- the effect of source-side word embedding strengthens when extra monolingual data is provided for training, and the domain of the monolingual data also seems to matter.

- When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? (CMU)
- How Do Source-side Monolingual Word Embeddings Impact Neural Machine Translation? (Johns Hopkins University)
- Context-Attentive Embeddings for Improved Sentence Representations (Facebook AI Research)

CAE

- introduce a novel, straightforward yet highly effective method for **combining multiple types of word embeddings in a single model**, leading to state-of-the-art performance within the same model class on a variety of tasks.

CAE

- Input: context vector **h**, different types of word embedding **w_e**
- Output: context-attended embedding **w_c**

$$\alpha_w = \text{softmax}(U\ f(V\ [\mathbf{h}, \mathbf{w}_{e_1}|, \ldots, \mathbf{w}_{e_{|E|}}])),$$

$$\mathbf{w}_c = [\alpha_1 \mathbf{w}_{e_1}, \ldots, \alpha_n \mathbf{w}_{e_n}].$$

- The embeddings are centered and L2-normalized in order to ensure an equal contribution of each embedding type if the weights are identical. The input embeddings are kept fixed during training.

CAE

- Sentence Encoders

  1. BiLSTM-Max

$$\overrightarrow{\mathbf{h}_t} = \overrightarrow{\mathrm{LSTM}_t}(\mathbf{w}^1, \ldots, \mathbf{w}^t)$$
$$\overleftarrow{\mathbf{h}_t} = \overleftarrow{\mathrm{LSTM}_t}(\mathbf{w}^t, \ldots, \mathbf{w}^T)$$

$$\max(\{[\overrightarrow{\mathbf{h}_t}, \overleftarrow{\mathbf{h}_t}]\}_{t=1,\ldots,T})$$

  2. Shortcut-Stacked

the input of the i-th stacked BiLSTM layer is given as

$$\mathbf{x}_t^i = [\mathbf{w}^t, \mathbf{h}_t^1, \ldots, \mathbf{h}_t^{i-1}]$$

CAE

- Sentence Encoders

  BiLSTM-Max

  - use the context-attended embedding w_c^t , calculated using h_{t-1} as the context for the forward model and h_{t+1} for the backward model

$$\overrightarrow{\mathbf{h}_t} = \overrightarrow{\mathrm{LSTM}_t}(\mathbf{w}_c^1, \ldots, \mathbf{w}_c^t)$$
$$\overleftarrow{\mathbf{h}_t} = \overleftarrow{\mathrm{LSTM}_t}(\mathbf{w}_c^t, \ldots, \mathbf{w}_c^T)$$

$$\max(\{[\overrightarrow{\mathbf{h}_t}, \overleftarrow{\mathbf{h}_t}]\}_{t=1,\ldots,T})$$

CAE

- Entropy regularization

$$\mathcal{L} = \mathcal{L}_{task} + \lambda \sum_{i=1}^{|E|} (\alpha_i \log(\alpha_i))$$

Exepriments

Model	SNLI	MNLI
InferSent (Conneau et al., 2017)	84.5	-
NSE (Munkhdalai and Yu, 2017)	84.6	-
G-TreeLSTM (Choi et al., 2017)	86.0	-
SSSE (Nie and Bansal, 2017)	86.1	73.6
ReSan (Shen et al., 2018)	86.3	-
BiLSTM-Max G	85.4	71.4
BiLSTM-Max F	86.0	72.9
BiLSTM-Max G+F Naive	85.8	73.1
BiLSTM-Max G+F CAE	86.1	73.2
Shortcut-Stacked G	86.6	74.1
Shortcut-Stacked F	86.3	73.6
Shortcut-Stacked G+F Naive	86.3	75.1
Shortcut-Stacked G+F CAE	86.7	75.8
Shortcut-Stacked Many Naive	86.2	75.2
Shortcut-Stacked Many CAE	87.0	75.9

Table 1: Natural language inference results on the SNLI and MultiNLI (Mismatched) tasks. CAE=Context Attentive Embeddings. G=GloVe, F=FastText, Many=multiple different embeddings (see Section 3).

Exepriments

Model	SST
Const. Tree LSTM (Tai et al., 2015)	88.0
DMN (Kumar et al., 2016)	88.6
DCG (Looks et al., 2017)	89.4
NSE (Munkhdalai and Yu, 2017)	89.7
BiLSTM-Max G	88.2
BiLSTM-Max F	89.1
BiLSTM-Max G+F Naive	88.6
BiLSTM-Max G+F CAE	89.4
Shortcut-Stacked G	89.3
Shortcut-Stacked F	89.0
Shortcut-Stacked G+F Naive	89.4
Shortcut-Stacked G+F CAE	89.7

Table 2: Sentiment classification results on the binary SST task. For DCG we compare against their best single sentence model (Looks et al., 2017).

CAE

- Conclustion:

- we find that a contextattentive approach, where embeddings are selecteddepending on the context, leads to better results.

- we show that the proposed mechanism leads to better interpretability and insightful linguistic analysis.

- When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation? (CMU)
- How Do Source-side Monolingual Word Embeddings Impact Neural Machine Translation? (Johns Hopkins University)
- Context-Attentive Embeddings for Improved Sentence Representations (Facebook AI Research)