Unsupervised MT

Huayang Li

2019/05/29

Contents

- Introduction
- Style Transfer
- Unsupervised MT
- Conclusion

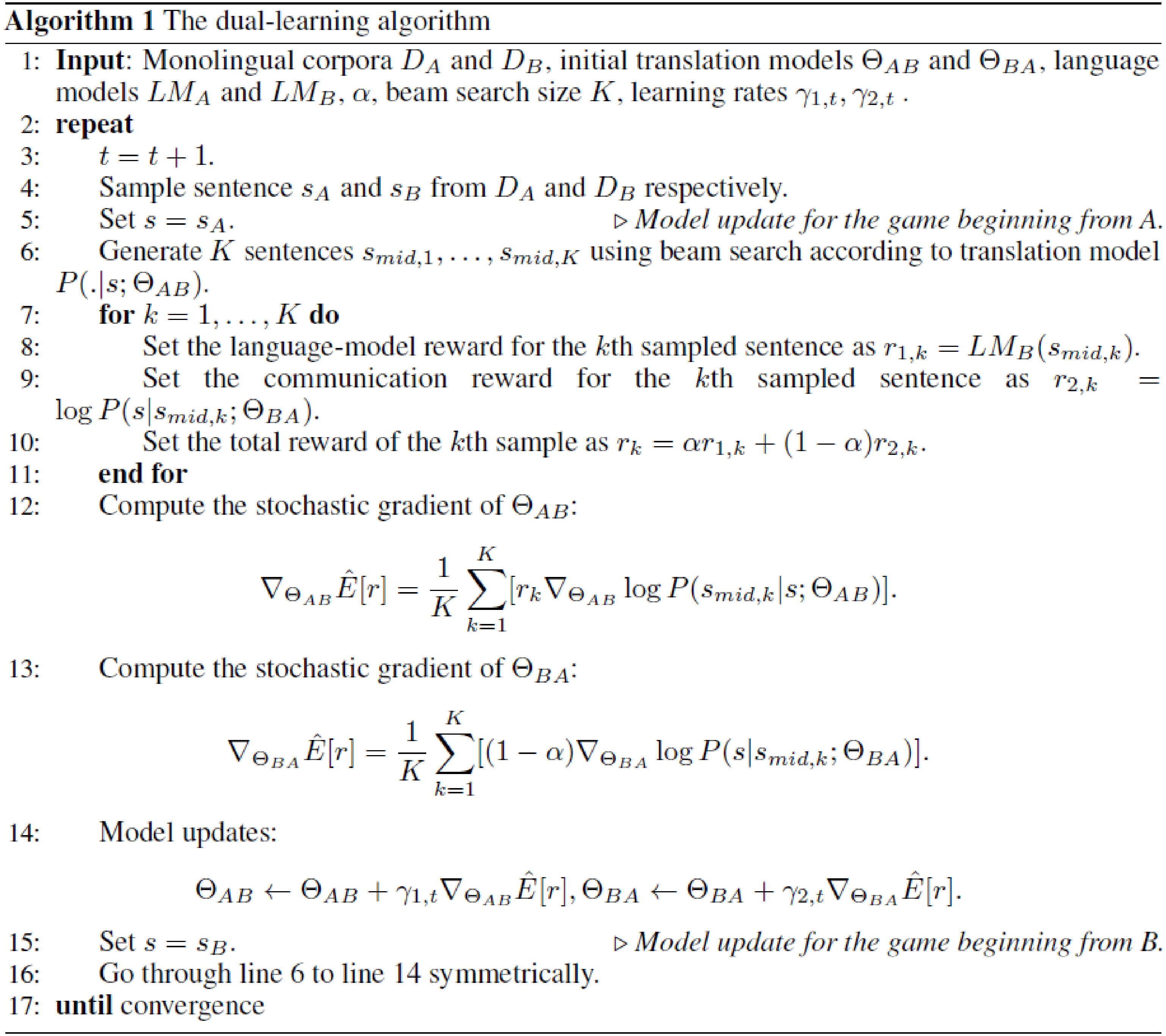
Introduction

- Improving Neural Machine Translation Models with Monolingual Data, ACL2016

- Dual Learning for Machine Translation, NIPS2016

Dual Learning for Machine Translation

- Two agents A and B
- A send a msg to a noisy channel which translate A to B
- B check whether the received msg is nature in language B
- B send this msg back to A through another noisy channel
- A check whether the msg received is consistent with original msg
- Repeat the previous game



Algorithm 1 The dual-learning algorithm

```
 1: Input: Monolingual corpora $D_A$ and $D_B$, initial translation models $\Theta_{AB}$ and $\Theta_{BA}$, language
    models $LM_A$ and $LM_B$, $\alpha$, beam search size $K$, learning rates $\gamma_{1,t}, \gamma_{2,t}$ .
 2: repeat
 3:     $t = t + 1$.
 4:     Sample sentence $s_A$ and $s_B$ from $D_A$ and $D_B$ respectively.
 5:     Set $s = s_A$.                          ▷ Model update for the game beginning from A.
 6:     Generate $K$ sentences $s_{mid,1}, \dots, s_{mid,K}$ using beam search according to translation model
    $P(.|s; \Theta_{AB})$.
 7:     for $k = 1, \dots, K$ do
 8:         Set the language-model reward for the $k$th sampled sentence as $r_{1,k} = LM_B(s_{mid,k})$.
 9:         Set the communication reward for the $k$th sampled sentence as $r_{2,k} =$
    $\log P(s|s_{mid,k}; \Theta_{BA})$.
10:         Set the total reward of the $k$th sample as $r_k = \alpha r_{1,k} + (1 - \alpha) r_{2,k}$.
11:     end for
12:     Compute the stochastic gradient of $\Theta_{AB}$:
            $$\nabla_{\Theta_{AB}} \hat{E}[r] = \frac{1}{K} \sum_{k=1}^{K} [r_k \nabla_{\Theta_{AB}} \log P(s_{mid,k}|s; \Theta_{AB})].$$
13:     Compute the stochastic gradient of $\Theta_{BA}$:
            $$\nabla_{\Theta_{BA}} \hat{E}[r] = \frac{1}{K} \sum_{k=1}^{K} [(1 - \alpha) \nabla_{\Theta_{BA}} \log P(s|s_{mid,k}; \Theta_{BA})].$$
14:     Model updates:
            $$\Theta_{AB} \leftarrow \Theta_{AB} + \gamma_{1,t} \nabla_{\Theta_{AB}} \hat{E}[r], \Theta_{BA} \leftarrow \Theta_{BA} + \gamma_{2,t} \nabla_{\Theta_{BA}} \hat{E}[r].$$
15:     Set $s = s_B$.                          ▷ Model update for the game beginning from B.
16:     Go through line 6 to line 14 symmetrically.
17: until convergence
```

Result of Dual Learning

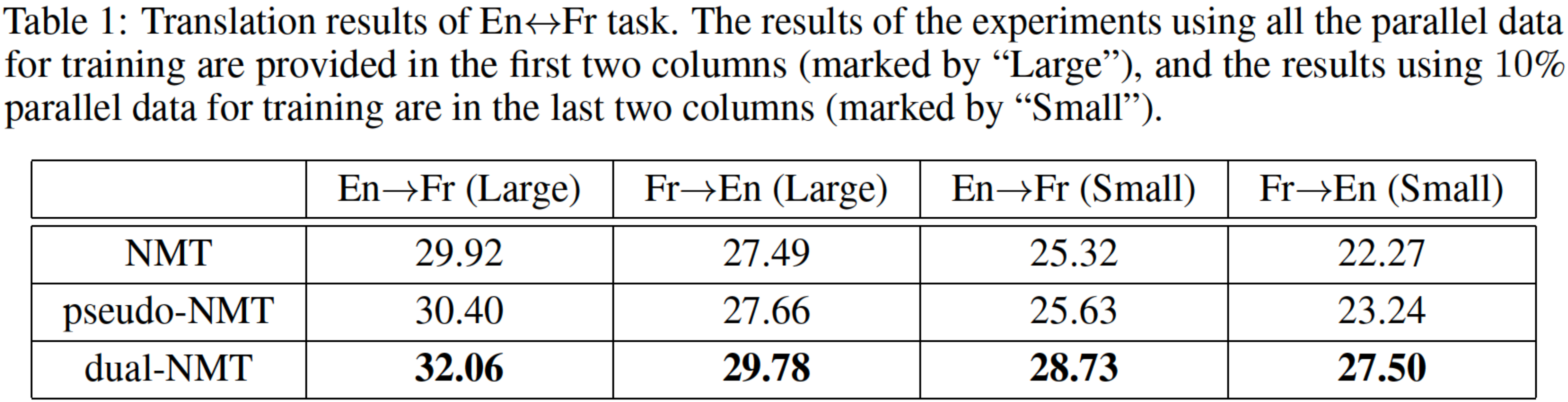


Table 1: Translation results of En↔Fr task. The results of the experiments using all the parallel data for training are provided in the first two columns (marked by "Large"), and the results using 10% parallel data for training are in the last two columns (marked by "Small").

	En→Fr (Large)	Fr→En (Large)	En→Fr (Small)	Fr→En (Small)
NMT	29.92	27.49	25.32	22.27
pseudo-NMT	30.40	27.66	25.63	23.24
dual-NMT	**32.06**	**29.78**	**28.73**	**27.50**

Style Transfer

- Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV2017
- Style Transfer from Non-Parallel Text by Cross-Alignment, NIPS2017

Cycle GAN

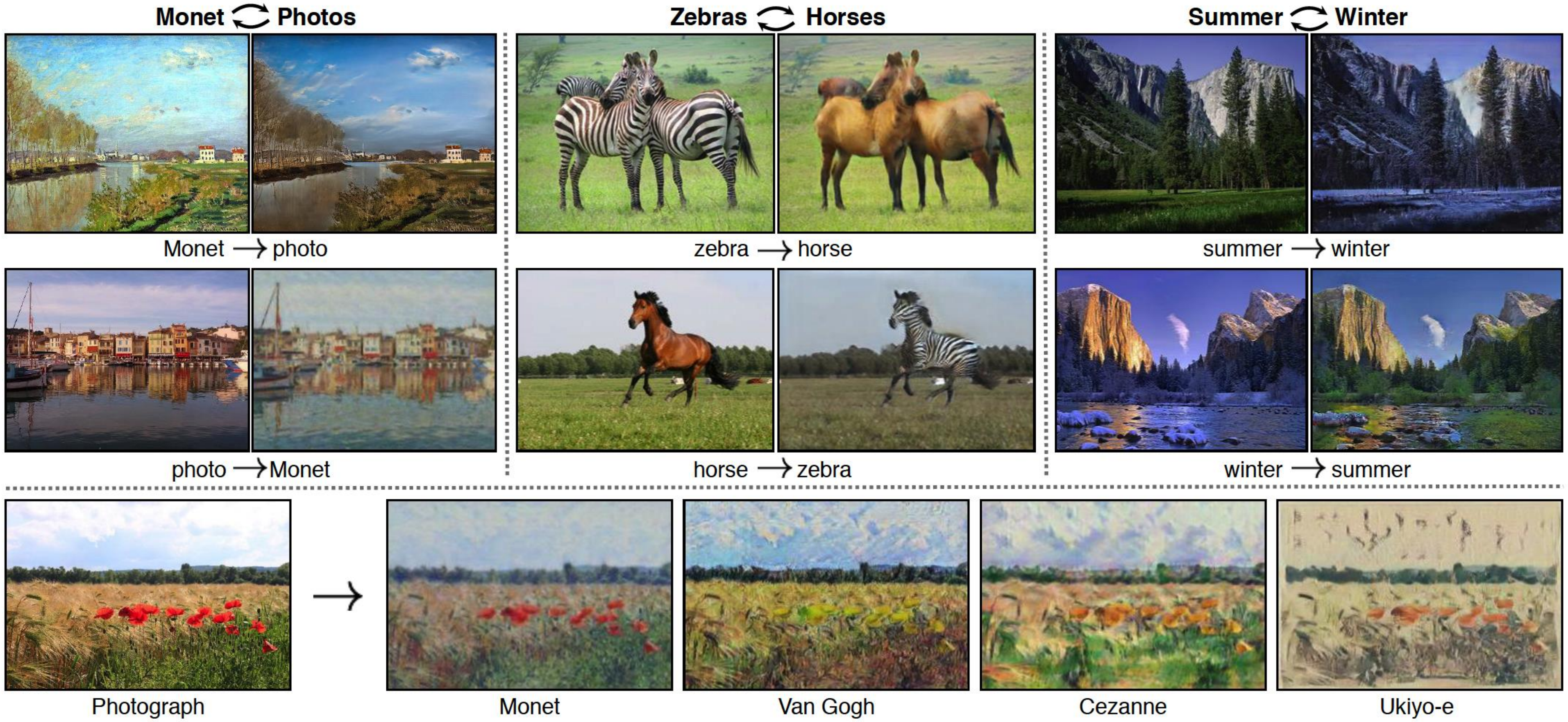


Figure 1: Given any two unordered image collections X and Y, our algorithm learns to automatically "translate" an image from one into the other and vice versa: *(left)* Monet paintings and landscape photos from Flickr; *(center)* zebras and horses from ImageNet; *(right)* summer and winter Yosemite photos from Flickr. Example application *(bottom)*: using a collection of paintings of famous artists, our method learns to render natural photographs into the respective styles.

Cycle GAN

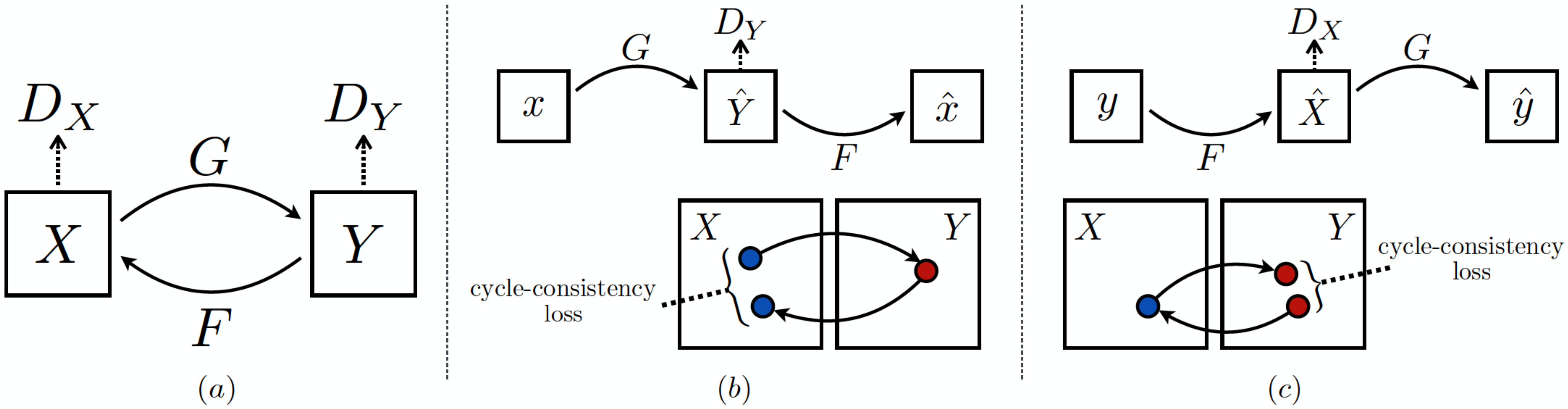
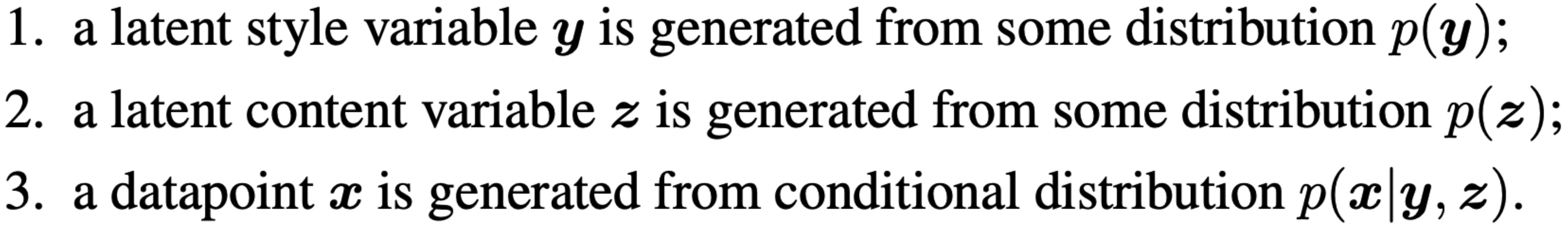


Figure 3: (a) Our model contains two mapping functions $G : X \rightarrow Y$ and $F : Y \rightarrow X$, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y, and vice versa for D_X and F. To further regularize the mappings, we introduce two *cycle consistency losses* that capture the intuition that if we translate from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss: $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$, and (c) backward cycle-consistency loss: $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$

Style Transfer from Non-Parallel Text

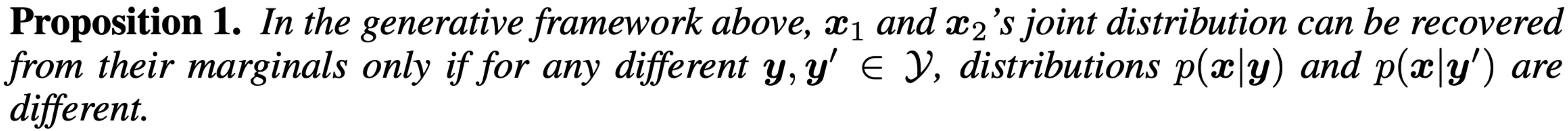
- Give us a proposition of unsupervised style transfer in NLP
- Introduce the reconstruction loss, aligned auto-encoder, and cross-aligned auto-encoder

The data generating process

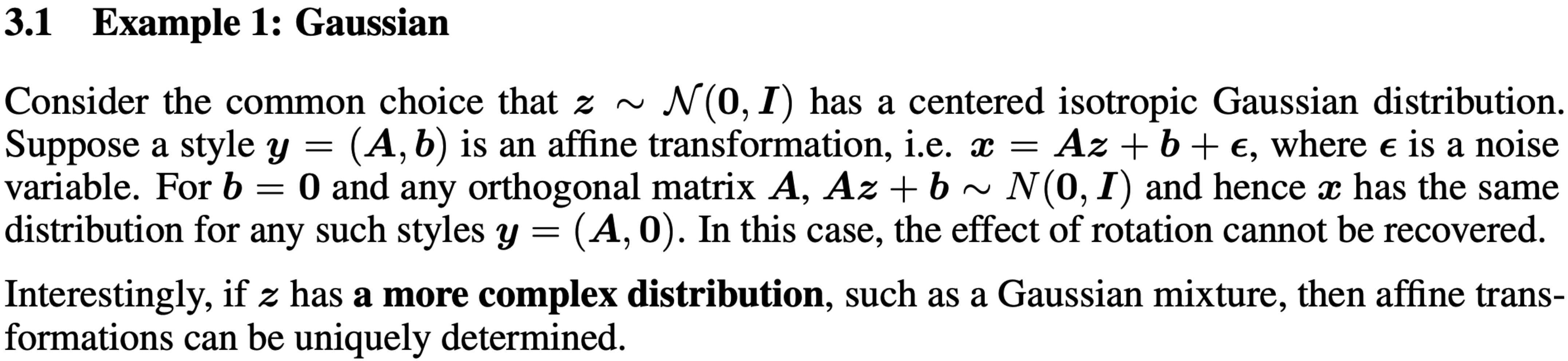


1. a latent style variable $\boldsymbol{y}$ is generated from some distribution $p(\boldsymbol{y})$;
2. a latent content variable $\boldsymbol{z}$ is generated from some distribution $p(\boldsymbol{z})$;
3. a datapoint $\boldsymbol{x}$ is generated from conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{z})$.

Proposition and Example



Proposition 1. *In the generative framework above,* $\boldsymbol{x}_1$ *and* $\boldsymbol{x}_2$*'s joint distribution can be recovered from their marginals only if for any different* $\boldsymbol{y}, \boldsymbol{y}' \in \mathcal{Y}$*, distributions* $p(\boldsymbol{x}|\boldsymbol{y})$ *and* $p(\boldsymbol{x}|\boldsymbol{y}')$ *are different.*



3.1 Example 1: Gaussian

Consider the common choice that $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ has a centered isotropic Gaussian distribution. Suppose a style $\boldsymbol{y} = (\boldsymbol{A}, \boldsymbol{b})$ is an affine transformation, i.e. $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{b} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is a noise variable. For $\boldsymbol{b} = \boldsymbol{0}$ and any orthogonal matrix $\boldsymbol{A}$, $\boldsymbol{A}\boldsymbol{z} + \boldsymbol{b} \sim N(\boldsymbol{0}, \boldsymbol{I})$ and hence $\boldsymbol{x}$ has the same distribution for any such styles $\boldsymbol{y} = (\boldsymbol{A}, \boldsymbol{0})$. In this case, the effect of rotation cannot be recovered.

Interestingly, if $\boldsymbol{z}$ has **a more complex distribution**, such as a Gaussian mixture, then affine transformations can be uniquely determined.

Reconstruction Loss

$$
\begin{aligned}
p(\boldsymbol{x}_1|\boldsymbol{x}_2;\boldsymbol{y}_1,\boldsymbol{y}_2) &= \int_{\boldsymbol{z}} p(\boldsymbol{x}_1,\boldsymbol{z}|\boldsymbol{x}_2;\boldsymbol{y}_1,\boldsymbol{y}_2)d\boldsymbol{z} \\
&= \int_{\boldsymbol{z}} p(\boldsymbol{z}|\boldsymbol{x}_2,\boldsymbol{y}_2)\cdot p(\boldsymbol{x}_1|\boldsymbol{y}_1,\boldsymbol{z})d\boldsymbol{z} \\
&= \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z}|\boldsymbol{x}_2,\boldsymbol{y}_2)}[p(\boldsymbol{x}_1|\boldsymbol{y}_1,\boldsymbol{z})]
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{L}_{\text{rec}}(\boldsymbol{\theta}_E,\boldsymbol{\theta}_G) = \ & \mathbb{E}_{\boldsymbol{x}_1\sim\boldsymbol{X}_1}[-\log p_G(\boldsymbol{x}_1|\boldsymbol{y}_1,E(\boldsymbol{x}_1,\boldsymbol{y}_1))] + \\
& \mathbb{E}_{\boldsymbol{x}_2\sim\boldsymbol{X}_2}[-\log p_G(\boldsymbol{x}_2|\boldsymbol{y}_2,E(\boldsymbol{x}_2,\boldsymbol{y}_2))]
\end{aligned}
$$

Aligned Auto-Encoder

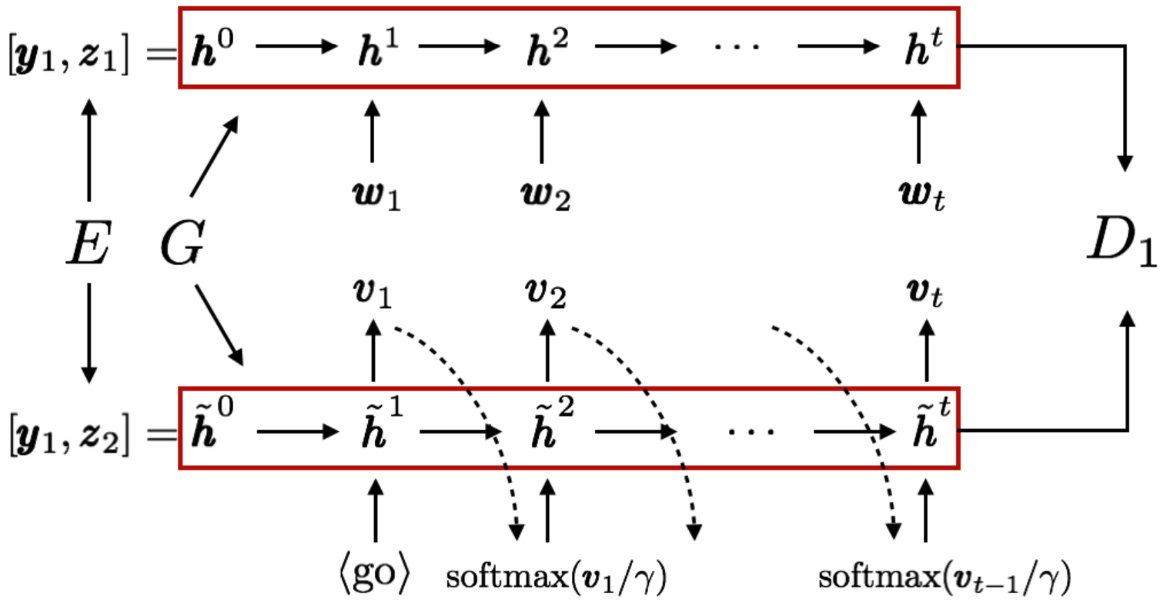
$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{rec}}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_G)$$

$$\text{s.t.} \quad E(\boldsymbol{x}_1, \boldsymbol{y}_1) \overset{\text{d}}{=} E(\boldsymbol{x}_2, \boldsymbol{y}_2) \qquad \boldsymbol{x}_1 \sim \boldsymbol{X}_1, \boldsymbol{x}_2 \sim \boldsymbol{X}_2$$

$$\mathcal{L}_{\text{adv}}(\boldsymbol{\theta}_E, \boldsymbol{\theta}_D) = \ \mathbb{E}_{\boldsymbol{x}_1 \sim \boldsymbol{X}_1}[-\log D(E(\boldsymbol{x}_1, \boldsymbol{y}_1))] \ + \mathbb{E}_{\boldsymbol{x}_2 \sim \boldsymbol{X}_2}[-\log(1 - D(E(\boldsymbol{x}_2, \boldsymbol{y}_2)))]$$

$$\min_{E,G} \max_{D} \mathcal{L}_{\text{rec}} - \lambda \mathcal{L}_{\text{adv}}$$

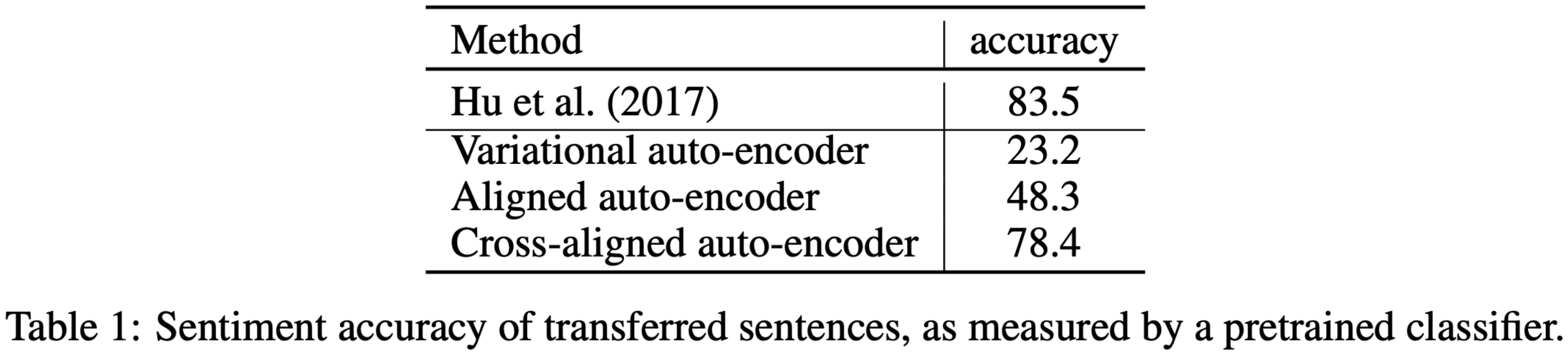
Cross-Aligned Auto-Encoder



$[\boldsymbol{y}_1, \boldsymbol{z}_1] =$
$\boldsymbol{h}^0$
$\boldsymbol{h}^1$
$\boldsymbol{h}^2$
$\cdots$
$\boldsymbol{h}^t$
$\boldsymbol{w}_1$
$\boldsymbol{w}_2$
$\boldsymbol{w}_t$
E
G
D_1
$\boldsymbol{v}_1$
$\boldsymbol{v}_2$
$\boldsymbol{v}_t$
$[\boldsymbol{y}_1, \boldsymbol{z}_2] =$
$\tilde{\boldsymbol{h}}^0$
$\tilde{\boldsymbol{h}}^1$
$\tilde{\boldsymbol{h}}^2$
$\cdots$
$\tilde{\boldsymbol{h}}^t$
$\langle$go$\rangle$
softmax$(\boldsymbol{v}_1/\gamma)$
softmax$(\boldsymbol{v}_{t-1}/\gamma)$

Figure 2: Cross-aligning between $\boldsymbol{x}_1$ and transferred $\boldsymbol{x}_2$. For $\boldsymbol{x}_1$, G is teacher-forced by its words $\boldsymbol{w}_1\boldsymbol{w}_2\cdots\boldsymbol{w}_t$. For transfered $\boldsymbol{x}_2$, G is self-fed by previous output logits. The sequence of hidden states $\boldsymbol{h}^0, \cdots, \boldsymbol{h}^t$ and $\tilde{\boldsymbol{h}}^0, \cdots, \tilde{\boldsymbol{h}}^t$ are passed to discriminator D_1 to be aligned. Note that our first variant aligned auto-encoder is a special case of this, where only $\boldsymbol{h}^0$ and $\tilde{\boldsymbol{h}}^0$, i.e. $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, are aligned.

Results



Method	accuracy
Hu et al. (2017)	83.5
Variational auto-encoder	23.2
Aligned auto-encoder	48.3
Cross-aligned auto-encoder	78.4

Table 1: Sentiment accuracy of transferred sentences, as measured by a pretrained classifier.

Unsupervised MT

- UNMT
- USMT
  - Unsupervised Statistical Machine Translation, EMNLP2018
  - Phrase-Based & Neural Unsupervised Machine Translation, EMNLP2018
- USTM+UNMT
  - Phrase-Based & Neural Unsupervised Machine Translation, EMNLP2018
  - An Effective Approach to Unsupervised Machine Translation, arxiv

UNMT

- Mining parallel data by back-translation
  1. Unsupervised Neural Machine Translation, ICLR2018
  2. Unsupervised Machine Translation Using Monolingual Corpora only, ICLR2018
  3. Unsupervised Neural Machine Translation with Weight Sharing, ACL2018
  4. Phrase-Based & Neural Unsupervised Machine Translation, EMNLP2018
- Mining parallel data by information retrieval
  1. Extract and Edit: An Alternative to Back-Translation for Unsupervised Neural Machine Translation, NAACL2019

Mining Parallel Data by Back-Translation

Common Things in Two ICLR2018 Papers

- Common components:
  - Dual structure
  - Shared encoder
  - Pretrained cross-lingual embeddings
- Common Strategies of training
  - Denoising auto-encoding
  - Back-translation

Different Things in Two ICLR2018 Papers

- Besides the encoder, Paper #2 also shared the decoder
- Paper #2 introduced an adversarial loss, which forced the encoder to encode different languages to the same space
- Datasets:
    - Paper #1 trained the En <-> Fr model on News Crawl monolingual corpus with articles from 2007 to 2013
    - Paper #2 built **word level** monolingual corpora by selecting the English sentences from 15 million random pairs, and selecting the French sentences from the complementary set.

Training Strategies from Paper #2

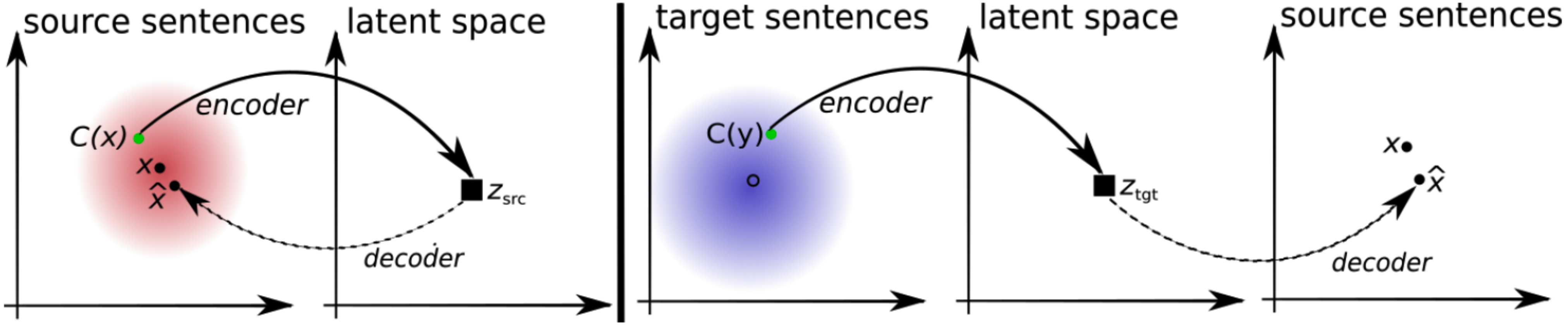


Figure 1: Toy illustration of the principles guiding the design of our objective function. Left (auto-encoding): the model is trained to reconstruct a sentence from a noisy version of it. x is the target, $C(x)$ is the noisy input, $\hat{x}$ is the reconstruction. Right (translation): the model is trained to translate a sentence in the other domain. The input is a noisy translation (in this case, from source-to-target) produced by the model itself, M, at the previous iteration (t), $y = M^{(t)}(x)$. The model is symmetric, and we repeat the same process in the other language. See text for more details.

Model of Paper #1

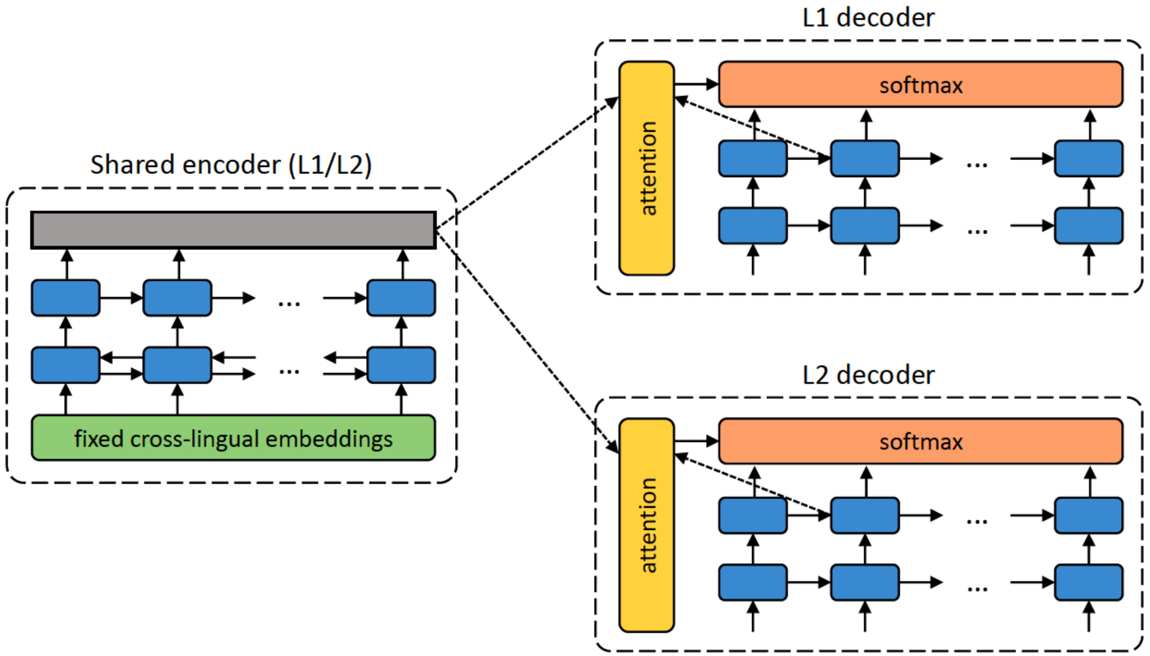


Figure 1: Architecture of the proposed system. For each sentence in language L1, the system is trained alternating two steps: *denoising*, which optimizes the probability of encoding a noised version of the sentence with the shared encoder and reconstructing it with the L1 decoder, and *on-the-fly backtranslation*, which translates the sentence in inference mode (encoding it with the shared encoder and decoding it with the L2 decoder) and then optimizes the probability of encoding this translated sentence with the shared encoder and recovering the original sentence with the L1 decoder. Training alternates between sentences in L1 and L2, with analogous steps for the latter.

Model of Paper #2

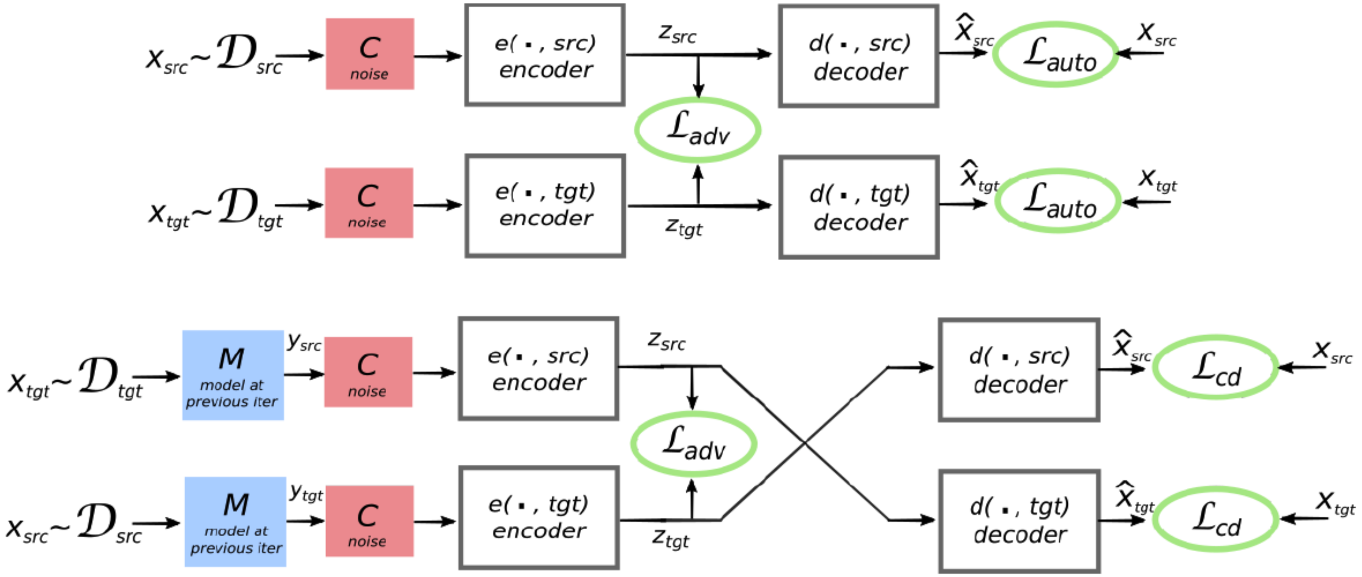
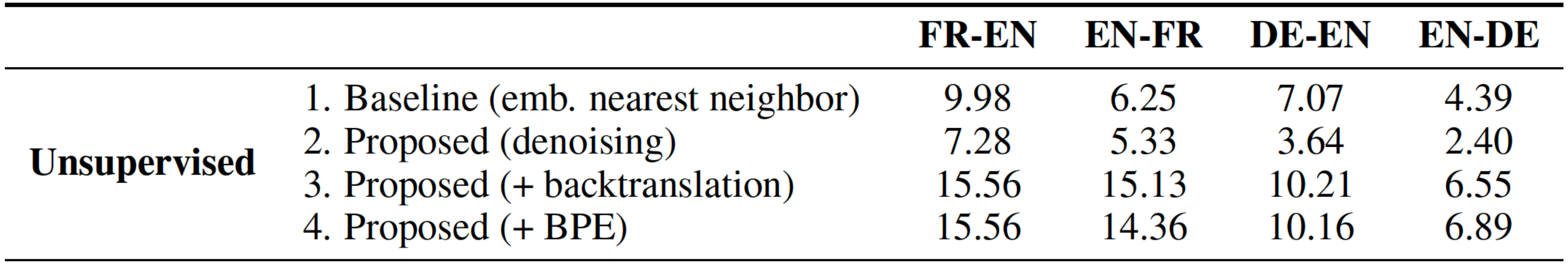
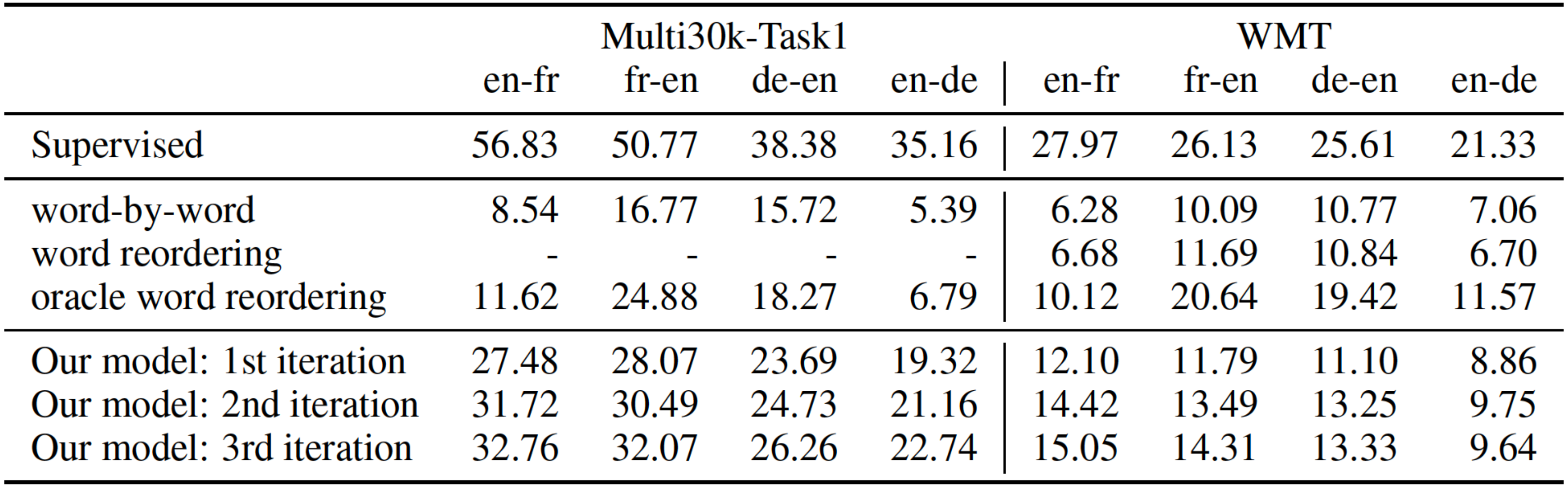


Figure 2: Illustration of the proposed architecture and training objectives. The architecture is a sequence to sequence model, with both encoder and decoder operating on two languages depending on an input language identifier that swaps lookup tables. Top (auto-encoding): the model learns to denoise sentences in each domain. Bottom (translation): like before, except that we encode from another language, using as input the translation produced by the model at the previous iteration (light blue box). The green ellipses indicate terms in the loss function.

Results of Paper #1 & #2



		FR-EN	EN-FR	DE-EN	EN-DE
Unsupervised	1. Baseline (emb. nearest neighbor)	9.98	6.25	7.07	4.39
	2. Proposed (denoising)	7.28	5.33	3.64	2.40
	3. Proposed (+ backtranslation)	15.56	15.13	10.21	6.55
	4. Proposed (+ BPE)	15.56	14.36	10.16	6.89

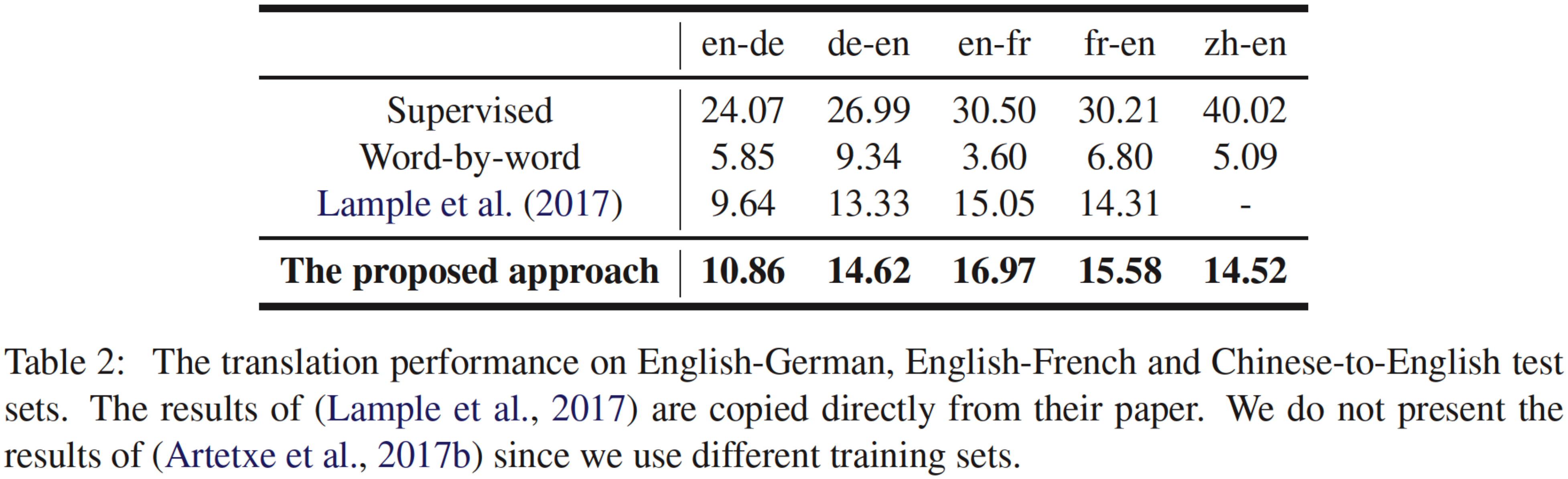


	Multi30k-Task1				WMT			
	en-fr	fr-en	de-en	en-de	en-fr	fr-en	de-en	en-de
Supervised	56.83	50.77	38.38	35.16	27.97	26.13	25.61	21.33
word-by-word	8.54	16.77	15.72	5.39	6.28	10.09	10.77	7.06
word reordering	-	-	-	-	6.68	11.69	10.84	6.70
oracle word reordering	11.62	24.88	18.27	6.79	10.12	20.64	19.42	11.57
Our model: 1st iteration	27.48	28.07	23.69	19.32	12.10	11.79	11.10	8.86
Our model: 2nd iteration	31.72	30.49	24.73	21.16	14.42	13.49	13.25	9.75
Our model: 3rd iteration	32.76	32.07	26.26	22.74	15.05	14.31	13.33	9.64

Improvements of Paper #3

- Paper #3 was based on paper #2 and made some improvements
    - Share the last few layers of encoders and the first few layers of decoders, which is claimed to capture the style and structure of different languages.
    - Introduce another adversarial loss, which is used to force the translated results and the real sentence in the same space.

Results of Paper #3

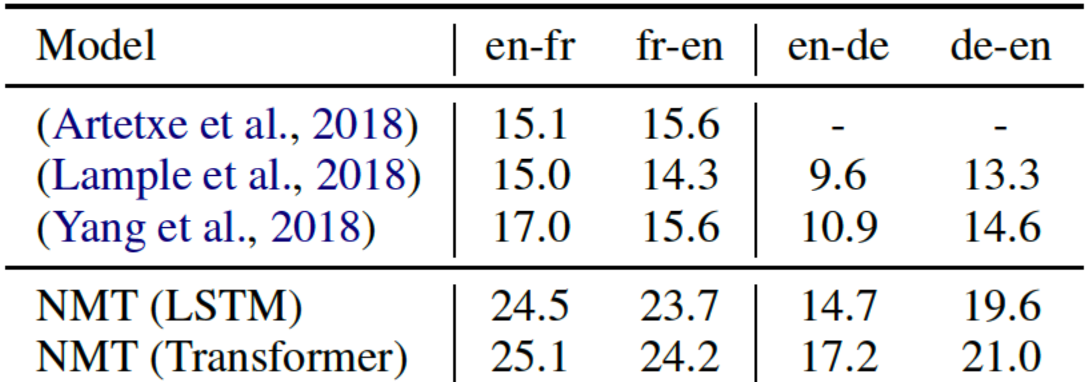


	en-de	de-en	en-fr	fr-en	zh-en
Supervised	24.07	26.99	30.50	30.21	40.02
Word-by-word	5.85	9.34	3.60	6.80	5.09
Lample et al. (2017)	9.64	13.33	15.05	14.31	-
The proposed approach	**10.86**	**14.62**	**16.97**	**15.58**	**14.52**

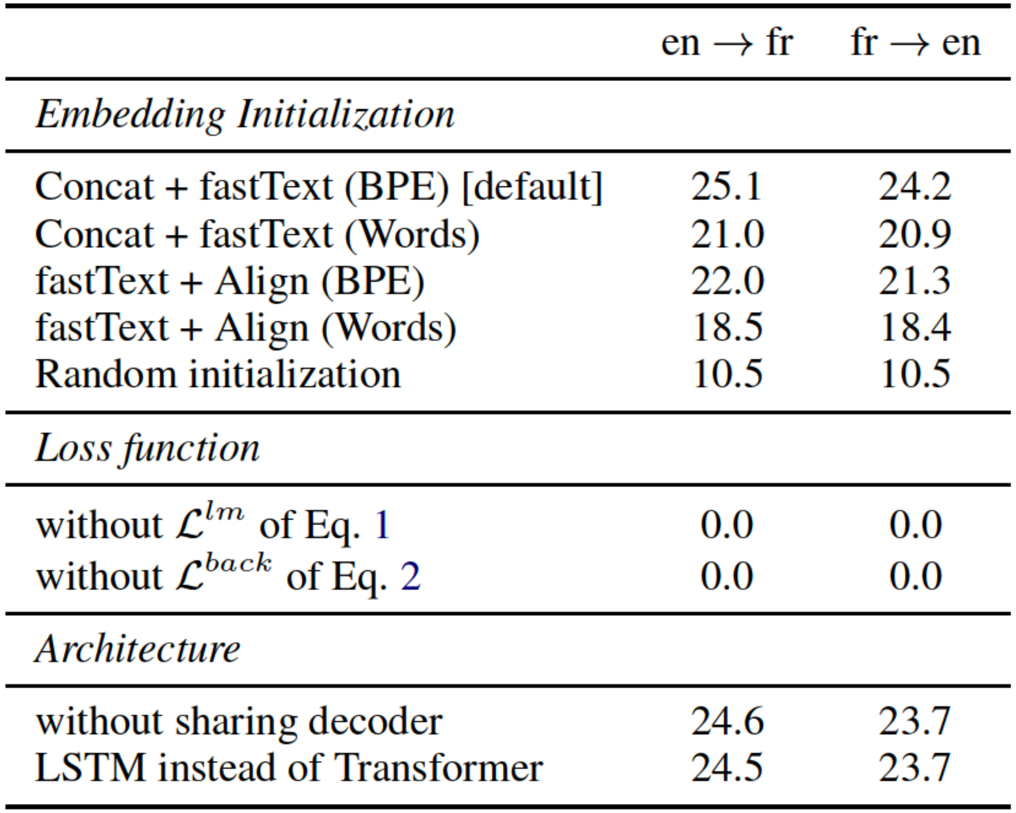
Table 2: The translation performance on English-German, English-French and Chinese-to-English test sets. The results of (Lample et al., 2017) are copied directly from their paper. We do not present the results of (Artetxe et al., 2017b) since we use different training sets.

Paper #4 Proposed A Standard UMT Framework

- Applying this framework to NMT scenario:
  - **Initialization**: jointly train the word embeddings of BPE level
  - **Language Modeling**: denoising autoencoding
  - **Back-translation**
  - **Sharing Latent Representations**: shared encoder and decoder
- Datasets
  - WMT monolingual News Crawl datasets from years 2007 through 2017



Model	en-fr	fr-en	en-de	de-en
(Artetxe et al., 2018)	15.1	15.6	-	-
(Lample et al., 2018)	15.0	14.3	9.6	13.3
(Yang et al., 2018)	17.0	15.6	10.9	14.6
NMT (LSTM)	24.5	23.7	14.7	19.6
NMT (Transformer)	25.1	24.2	17.2	21.0



	en → fr	fr → en
Embedding Initialization		
Concat + fastText (BPE) [default]	25.1	24.2
Concat + fastText (Words)	21.0	20.9
fastText + Align (BPE)	22.0	21.3
fastText + Align (Words)	18.5	18.4
Random initialization	10.5	10.5
Loss function		
without $\mathcal{L}^{lm}$ of Eq. 1	0.0	0.0
without $\mathcal{L}^{back}$ of Eq. 2	0.0	0.0
Architecture		
without sharing decoder	24.6	23.7
LSTM instead of Transformer	24.5	23.7

Table 4: **Ablation study of unsupervised NMT.** BLEU scores are computed over *newstest* 2014.

Mining Parallel Data by Information Retrieval

Extract and Edit

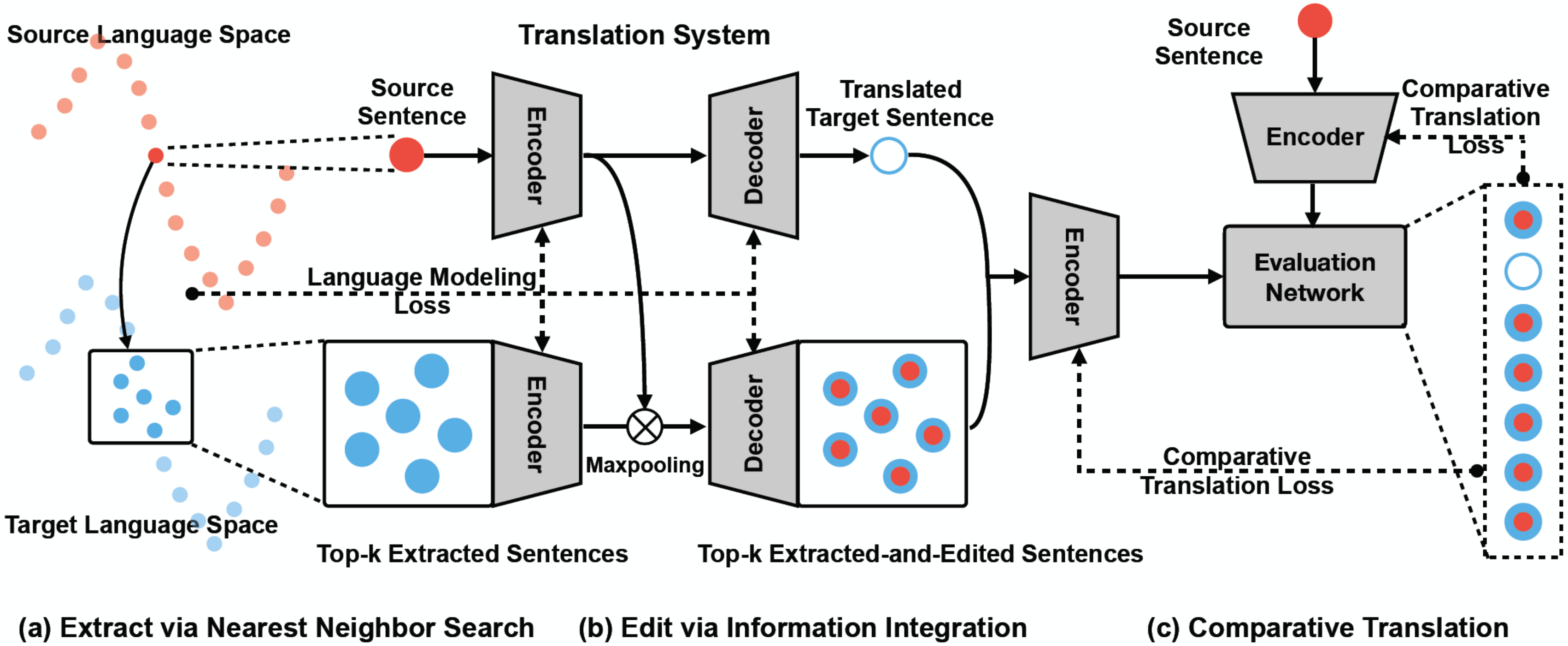

Source Language Space
Translation System
Source Sentence
Encoder
Decoder
Translated Target Sentence
Source Sentence
Encoder
Comparative Translation Loss
Language Modeling Loss
Encoder
Evaluation Network
Encoder
Maxpooling
Decoder
Comparative Translation Loss
Target Language Space
Top-k Extracted Sentences
Top-k Extracted-and-Edited Sentences
(a) Extract via Nearest Neighbor Search
(b) Edit via Information Integration
(c) Comparative Translation

Figure 2: The overview of our unsupervised NMT model based on the *extract-edit* approach. Given a source sentence, (a) the top-k potential parallel sentences of the target language are extracted via nearest neighbor search. (b) The extracted sentences are further edited with the source sentence. (c) The evaluation network evaluates the translated sentence and the extracted-and-edited sentences based on their similarities with the source sentence. Note that (1) all the encoders share the same parameters (same for decoders); (2) the decoding processes are non-differentiable, so the language modeling loss and the comparative translation loss are used to train the learning modules before and after the decoding processes, respectively.

Extract:

$$M = \{t| \min_{1,\cdots,k} (||e_s - e_t||), t \in \mathcal{T}\}, \tag{2}$$

Edit:

$$M' = \{t'|t' = dec(\text{maxpooling}(e_s, e_t)), t \in M\}, \tag{3}$$

Evaluate

$$\alpha(t|s) = cosine(r_t, r_s) = \frac{r_t \cdot r_s}{||r_t||||r_s||}. \qquad (4)$$

$$P(t^*|s, M') = \frac{exp(\lambda\alpha(t^*|s))}{\sum_{t' \in M' \cup \{t^*\}} exp(\lambda\alpha(t'|s))}, \quad (5)$$

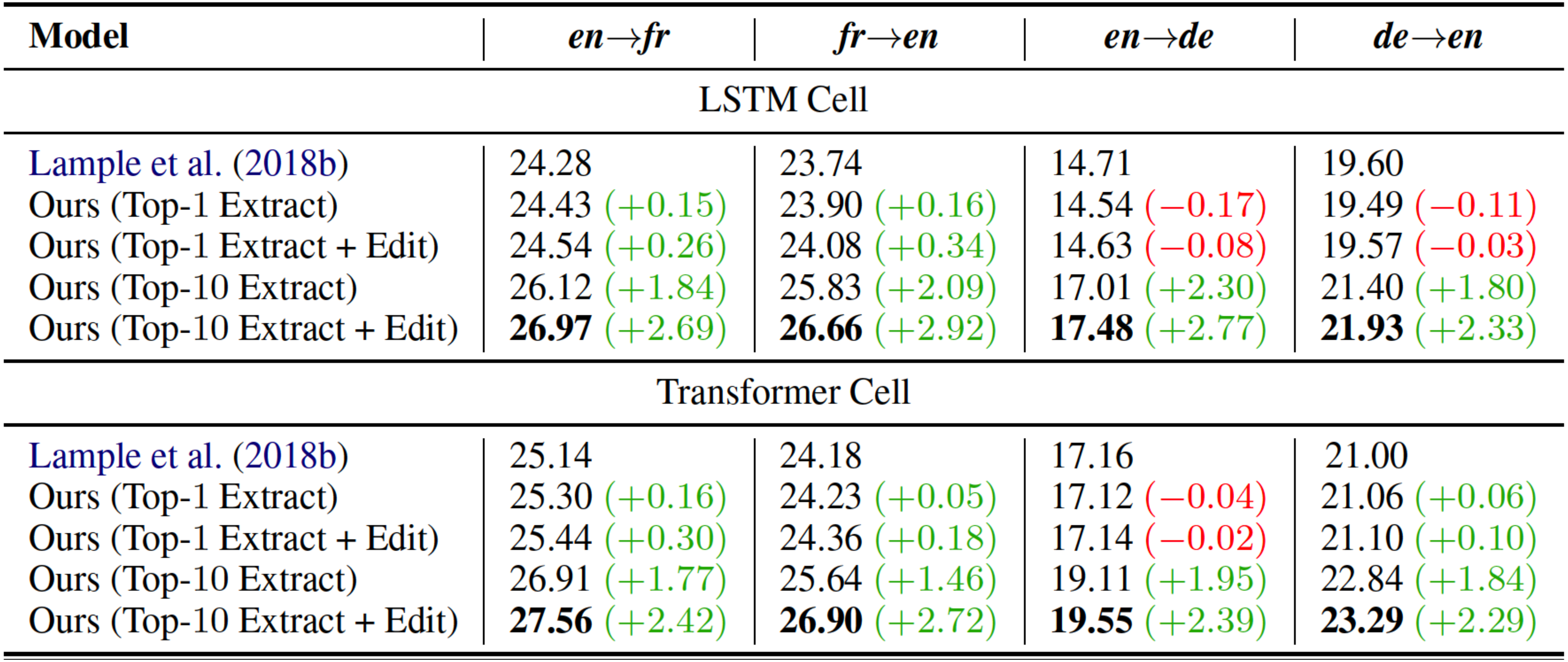
Learning

$$\mathcal{L}_{com}(\theta_{enc}|\theta_R) = -\mathbb{E}(\log P(t^* = V_{s\to t}(s)|s, M')), \tag{6}$$

$$\begin{aligned}\mathcal{L}_{s\to t}(\theta_{enc}, \theta_{dec}|\theta_R) =& \omega_{lm}\mathcal{L}_{lm}(\theta_{enc}, \theta_{dec}) + \\ & \omega_{com}\mathcal{L}_{com}(\theta_{enc}|\theta_R),\end{aligned} \tag{7}$$

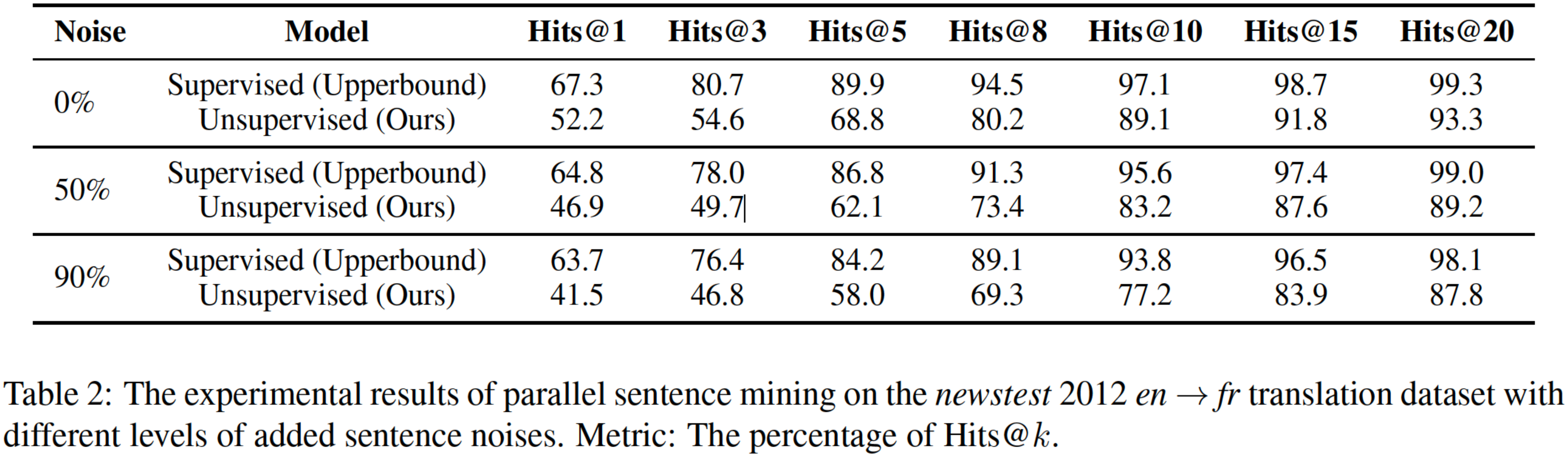
- Where L_{com} is the introduced comparative translation loss, and L_{lm} is the denoising auto-encoding loss.

Results



Model	***en→fr***	***fr→en***	***en→de***	***de→en***
		LSTM Cell		
Lample et al. (2018b)	24.28	23.74	14.71	19.60
Ours (Top-1 Extract)	24.43 (+0.15)	23.90 (+0.16)	14.54 (−0.17)	19.49 (−0.11)
Ours (Top-1 Extract + Edit)	24.54 (+0.26)	24.08 (+0.34)	14.63 (−0.08)	19.57 (−0.03)
Ours (Top-10 Extract)	26.12 (+1.84)	25.83 (+2.09)	17.01 (+2.30)	21.40 (+1.80)
Ours (Top-10 Extract + Edit)	**26.97** (+2.69)	**26.66** (+2.92)	**17.48** (+2.77)	**21.93** (+2.33)
		Transformer Cell		
Lample et al. (2018b)	25.14	24.18	17.16	21.00
Ours (Top-1 Extract)	25.30 (+0.16)	24.23 (+0.05)	17.12 (−0.04)	21.06 (+0.06)
Ours (Top-1 Extract + Edit)	25.44 (+0.30)	24.36 (+0.18)	17.14 (−0.02)	21.10 (+0.10)
Ours (Top-10 Extract)	26.91 (+1.77)	25.64 (+1.46)	19.11 (+1.95)	22.84 (+1.84)
Ours (Top-10 Extract + Edit)	**27.56** (+2.42)	**26.90** (+2.72)	**19.55** (+2.39)	**23.29** (+2.29)

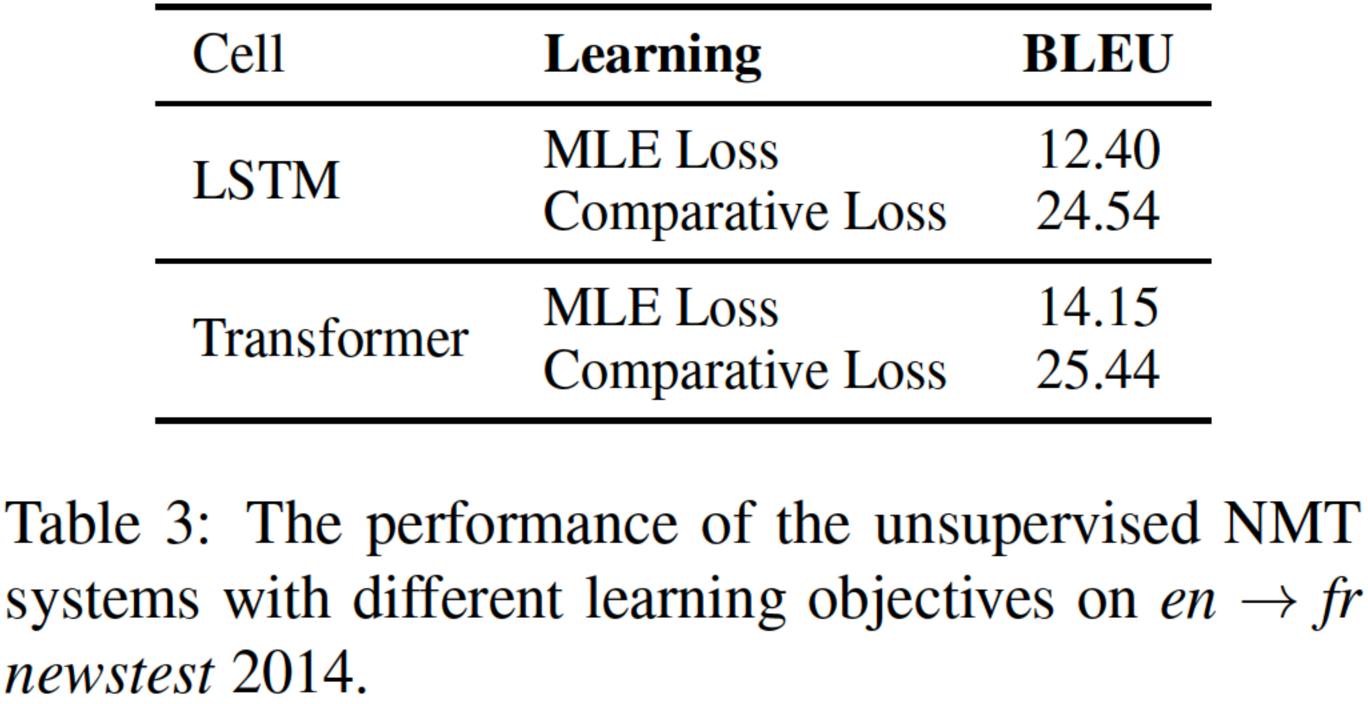
Analysis



Noise	Model	Hits@1	Hits@3	Hits@5	Hits@8	Hits@10	Hits@15	Hits@20
0%	Supervised (Upperbound)	67.3	80.7	89.9	94.5	97.1	98.7	99.3
	Unsupervised (Ours)	52.2	54.6	68.8	80.2	89.1	91.8	93.3
50%	Supervised (Upperbound)	64.8	78.0	86.8	91.3	95.6	97.4	99.0
	Unsupervised (Ours)	46.9	49.7	62.1	73.4	83.2	87.6	89.2
90%	Supervised (Upperbound)	63.7	76.4	84.2	89.1	93.8	96.5	98.1
	Unsupervised (Ours)	41.5	46.8	58.0	69.3	77.2	83.9	87.8

Table 2: The experimental results of parallel sentence mining on the *newstest* 2012 *en* → *fr* translation dataset with different levels of added sentence noises. Metric: The percentage of Hits@k.

Analysis



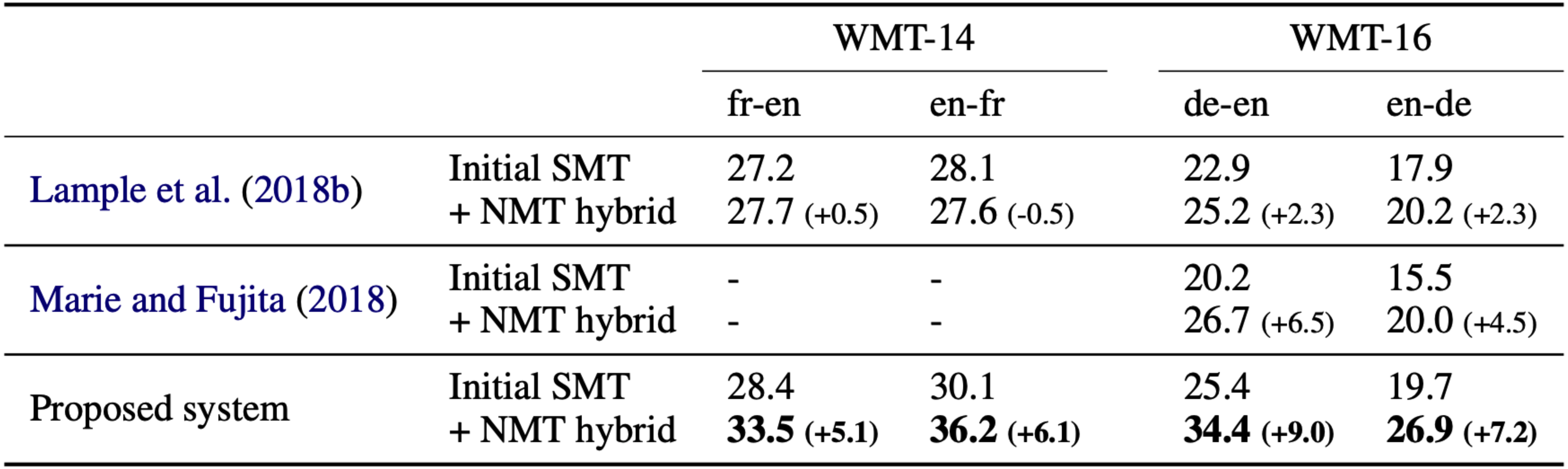
Cell	**Learning**	**BLEU**
LSTM	MLE Loss	12.40
	Comparative Loss	24.54
Transformer	MLE Loss	14.15
	Comparative Loss	25.44

Table 3: The performance of the unsupervised NMT systems with different learning objectives on *en* → *fr* *newstest* 2014.

Unsupervised MT

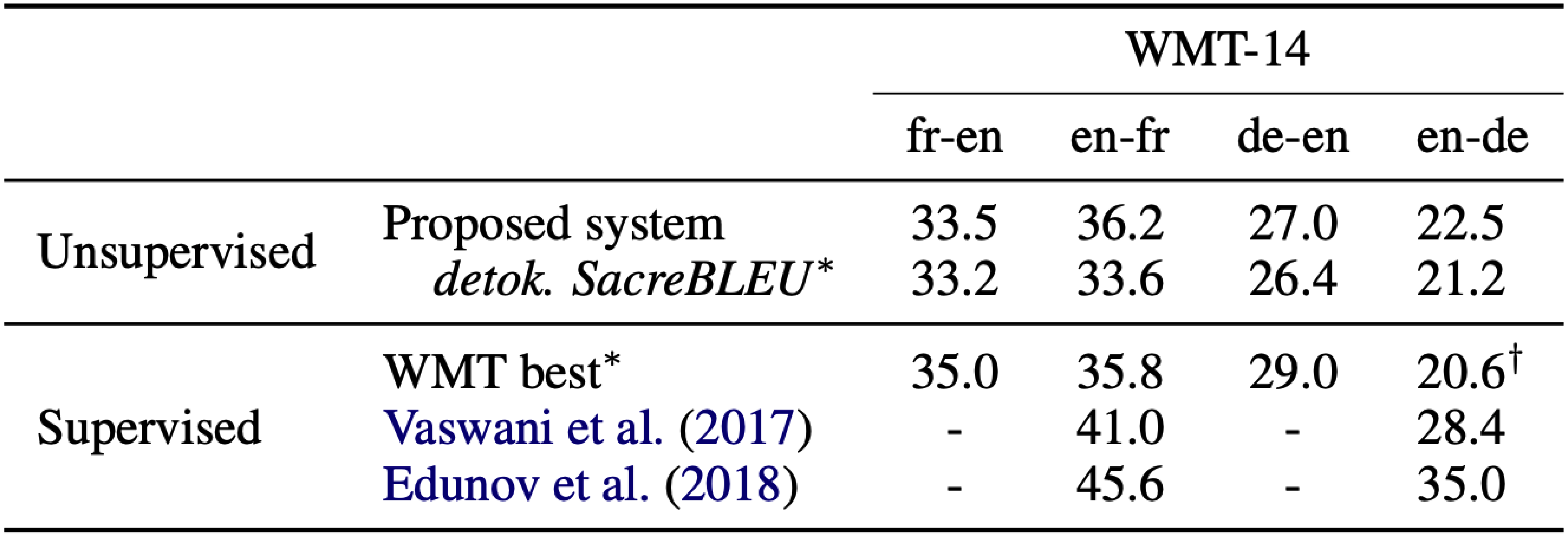
- UNMT
- USMT
  - Unsupervised Statistical Machine Translation, EMNLP2018
  - Phrase-Based & Neural Unsupervised Machine Translation, EMNLP2018
- USTM+UNMT
  - Phrase-Based & Neural Unsupervised Machine Translation, EMNLP2018
  - An Effective Approach to Unsupervised Machine Translation, arxiv

An Effective Approach to Unsupervised Machine Translation



		WMT-14		WMT-16	
		fr-en	en-fr	de-en	en-de
Lample et al. (2018b)	Initial SMT	27.2	28.1	22.9	17.9
	+ NMT hybrid	27.7 (+0.5)	27.6 (-0.5)	25.2 (+2.3)	20.2 (+2.3)
Marie and Fujita (2018)	Initial SMT	-	-	20.2	15.5
	+ NMT hybrid	-	-	26.7 (+6.5)	20.0 (+4.5)
Proposed system	Initial SMT	28.4	30.1	25.4	19.7
	+ NMT hybrid	**33.5 (+5.1)**	**36.2 (+6.1)**	**34.4 (+9.0)**	**26.9 (+7.2)**

An Effective Approach to Unsupervised Machine Translation



		WMT-14			
		fr-en	en-fr	de-en	en-de
Unsupervised	Proposed system	33.5	36.2	27.0	22.5
	*detok. SacreBLEU**	33.2	33.6	26.4	21.2
Supervised	WMT best*	35.0	35.8	29.0	20.6†
	Vaswani et al. (2017)	-	41.0	-	28.4
	Edunov et al. (2018)	-	45.6	-	35.0

Conclusion

- Is there any better way to mining parallel datasets ?
- Is there any better way to warm up the UNMT system, because the translation quality is really poor at the beginning.
  - Repeating phrases
  - Ungrammatical sentences
  - …
- Is it effective enough to chose the language of shared decoder/encoder by a simple special token ?
- Is there another choice of language model ?