Generative Neural Machine Translation

Xing Wang

2018-06-19

- Motivations
- Model
- Experiments

- Motivations

- Model

- Experiments

- VNMT: the latent representation is dependent on the source sentence, it can be argued that it doesn’t serve a different purpose to the encoder hidden states of the baseline model.

$$p(\mathbf{y}|\mathbf{x}) = \int_z p(\mathbf{y}, z|\mathbf{x}) d_z = \int_z p(\mathbf{y}|z, \mathbf{x}) p(z|\mathbf{x}) d_z$$

- Variational Neural Machine Translation (VNMT) [Zhang et al., 2016] attempts to achieve this by augmenting a baseline model with a latent variable intended to represent the underlying semantics of the source sentence.

- GNMT: models the joint distribution of the target sentence and the source sentence. To do this, it uses the latent variable as a language agnostic representation of the sentence, which generates text in both the source and target languages.

$$p_\theta(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{y}|\mathbf{z}, \mathbf{x})$$

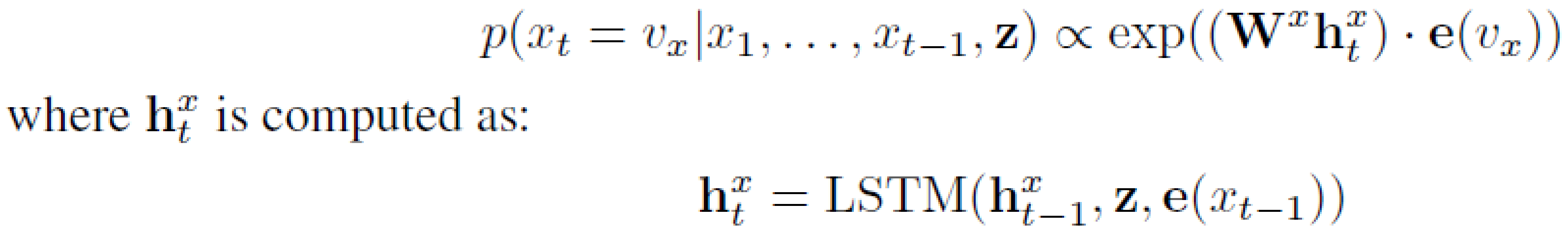
- This architecture means that z models the commonality between the source and target sentences, which is the semantic meaning. We use a Gaussian prior: p(z) = N(0; I).

- Motivations

- Model

- Experiments

- Generative process
- Source sentence

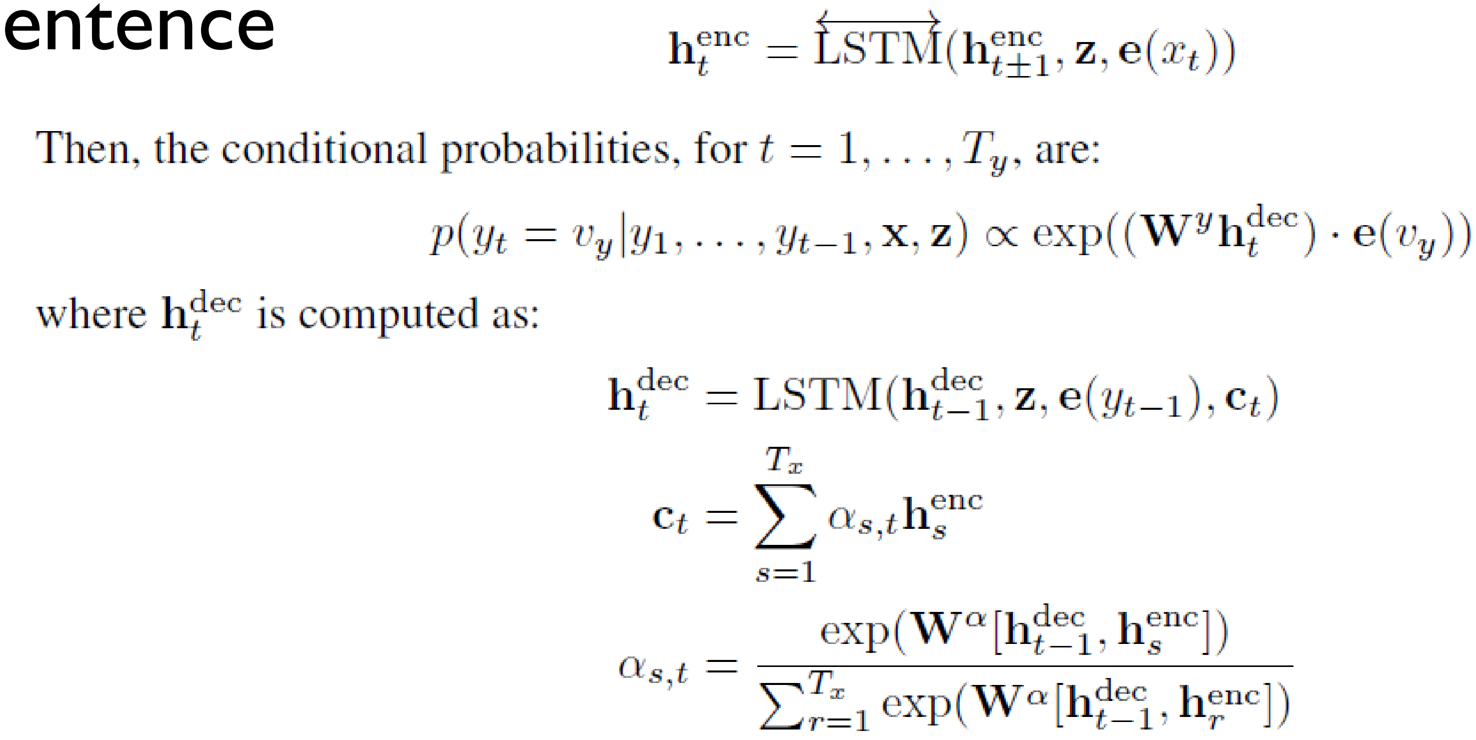


$$p(x_t = v_x | x_1, \ldots, x_{t-1}, \mathbf{z}) \propto \exp((\mathbf{W}^x \mathbf{h}_t^x) \cdot \mathbf{e}(v_x))$$

where $\mathbf{h}_t^x$ is computed as:

$$\mathbf{h}_t^x = \mathrm{LSTM}(\mathbf{h}_{t-1}^x, \mathbf{z}, \mathbf{e}(x_{t-1}))$$

- Generative process
- Target sentence



$$\mathbf{h}_t^{\text{enc}} = \overleftrightarrow{\text{LSTM}}(\mathbf{h}_{t\pm1}^{\text{enc}}, \mathbf{z}, \mathbf{e}(x_t))$$

Then, the conditional probabilities, for $t = 1, \dots, T_y$, are:

$$p(y_t = v_y | y_1, \dots, y_{t-1}, \mathbf{x}, \mathbf{z}) \propto \exp((\mathbf{W}^y \mathbf{h}_t^{\text{dec}}) \cdot \mathbf{e}(v_y))$$

where $\mathbf{h}_t^{\text{dec}}$ is computed as:

$$\mathbf{h}_t^{\text{dec}} = \text{LSTM}(\mathbf{h}_{t-1}^{\text{dec}}, \mathbf{z}, \mathbf{e}(y_{t-1}), \mathbf{c}_t)$$

$$\mathbf{c}_t = \sum_{s=1}^{T_x} \alpha_{s,t} \mathbf{h}_s^{\text{enc}}$$

$$\alpha_{s,t} = \frac{\exp(\mathbf{W}^\alpha [\mathbf{h}_{t-1}^{\text{dec}}, \mathbf{h}_s^{\text{enc}}])}{\sum_{r=1}^{T_x} \exp(\mathbf{W}^\alpha [\mathbf{h}_{t-1}^{\text{dec}}, \mathbf{h}_r^{\text{enc}}])}$$

- Training

$$q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x},\mathbf{y}), \boldsymbol{\Sigma}_\phi(\mathbf{x},\mathbf{y}))$$

$$\log p(\mathbf{x},\mathbf{y}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\right] \equiv \mathcal{L}(\mathbf{x},\mathbf{y})$$

$$\mathcal{L}(\mathbf{x},\mathbf{y}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p(\mathbf{x},\mathbf{y}|\mathbf{z})\right] - D_{\mathrm{KL}}\left[q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y}) \,||\, p(\mathbf{z})\right]$$

- Training
- We first encode the source and target sentences using a bidirectional LSTM.

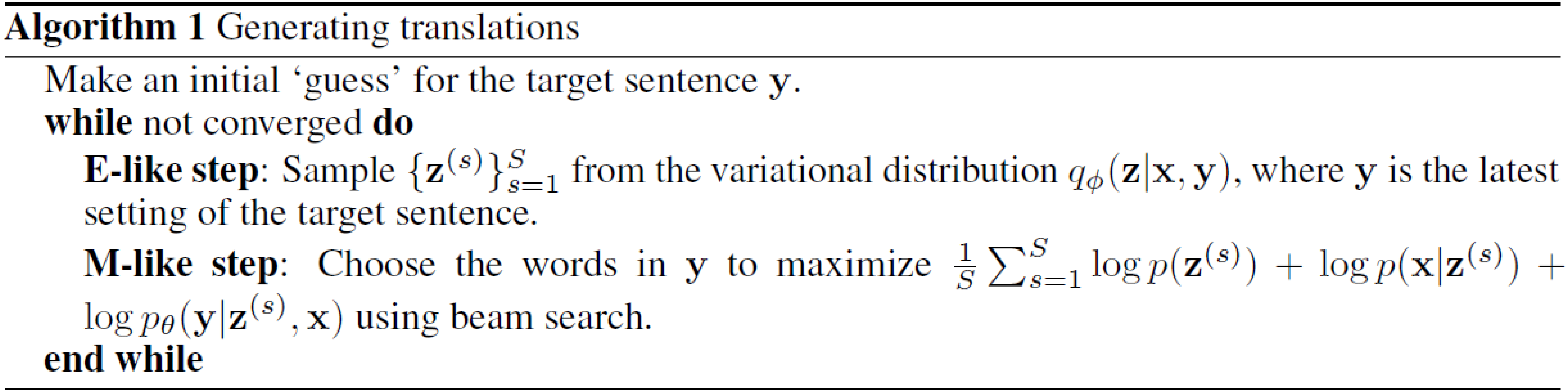
$$\mathbf{h}_t^{\mathrm{inf},i} = \overleftrightarrow{\mathrm{LSTM}}(\mathbf{h}_{t\pm1}^{\mathrm{inf},i}, \mathbf{e}(i_t)) \quad \text{where } i_t = \begin{cases} x_t & \text{if } i = x \\ y_t & \text{if } i = y \end{cases}$$

- We then concatenate the averages of the two sets of hidden states, and use this vector to compute the mean and variance of the Gaussian distribution.

$$\mathbf{h}^{\mathrm{inf}} = \left[\frac{1}{T_x} \sum_{t=1}^{T_x} \mathbf{h}_t^{\mathrm{inf},x}, \frac{1}{T_y} \sum_{t=1}^{T_y} \mathbf{h}_t^{\mathrm{inf},y} \right]$$

$$q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y}) = \mathcal{N}(\mathbf{W}^\mu \mathbf{h}^{\mathrm{inf}},\ \mathrm{diag}(\exp(\mathbf{W}^\Sigma \mathbf{h}^{\mathrm{inf}})))$$

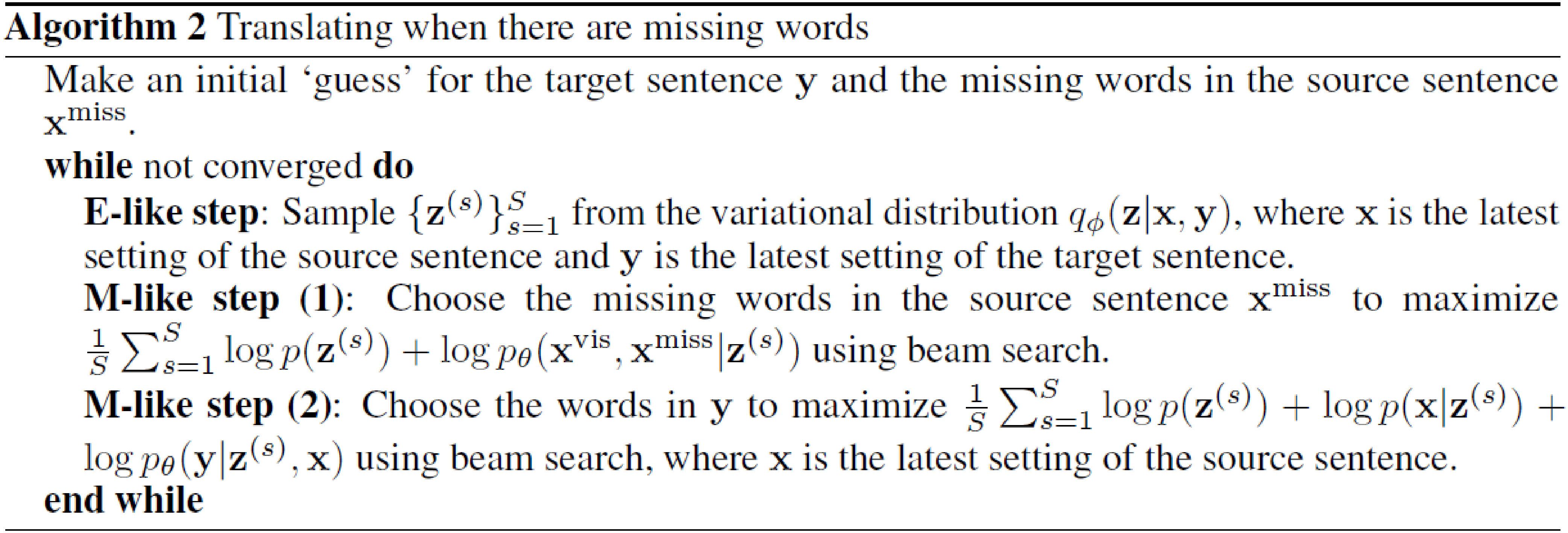
- Generating translations
- Because $\text{argmax}_y\ p(y|x) = \text{argmax}_y\ p(x, y)$, we can perform approximate maximization by using a procedure inspired by the EM algorithm



Algorithm 1 Generating translations

```
Make an initial ‘guess’ for the target sentence $\mathbf{y}$.
while not converged do
    E-like step: Sample $\{\mathbf{z}^{(s)}\}_{s=1}^{S}$ from the variational distribution $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})$, where $\mathbf{y}$ is the latest setting of the target sentence.
    M-like step: Choose the words in $\mathbf{y}$ to maximize $\frac{1}{S}\sum_{s=1}^{S} \log p(\mathbf{z}^{(s)}) + \log p(\mathbf{x}|\mathbf{z}^{(s)}) + \log p_\theta(\mathbf{y}|\mathbf{z}^{(s)}, \mathbf{x})$ using beam search.
end while
```

- Translating with missing words
- GNMT is naturally suited to performing translation when there are missing words in the source sentence, because it can use the latent representation to infer what those missing words may be.



Algorithm 2 Translating when there are missing words

```
Make an initial ‘guess’ for the target sentence $\mathbf{y}$ and the missing words in the source sentence $\mathbf{x}^{\text{miss}}$.
while not converged do
    E-like step: Sample $\{\mathbf{z}^{(s)}\}_{s=1}^{S}$ from the variational distribution $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$, where $\mathbf{x}$ is the latest setting of the source sentence and $\mathbf{y}$ is the latest setting of the target sentence.
    M-like step (1): Choose the missing words in the source sentence $\mathbf{x}^{\text{miss}}$ to maximize $\frac{1}{S}\sum_{s=1}^{S} \log p(\mathbf{z}^{(s)}) + \log p_\theta(\mathbf{x}^{\text{vis}}, \mathbf{x}^{\text{miss}}|\mathbf{z}^{(s)})$ using beam search.
    M-like step (2): Choose the words in $\mathbf{y}$ to maximize $\frac{1}{S}\sum_{s=1}^{S} \log p(\mathbf{z}^{(s)}) + \log p(\mathbf{x}|\mathbf{z}^{(s)}) + \log p_\theta(\mathbf{y}|\mathbf{z}^{(s)}, \mathbf{x})$ using beam search, where $\mathbf{x}$ is the latest setting of the source sentence.
end while
```

- Multilingual translation
- add two categorical variables to GNMT, l_x and l_y (encoded as one-hot vectors)

$$p_\theta(\mathbf{x}, \mathbf{y}, \mathbf{z} | l_x, l_y) = p(\mathbf{z}) p_\theta(\mathbf{x} | \mathbf{z}, l_x) p_\theta(\mathbf{y} | \mathbf{z}, \mathbf{x}, l_x, l_y)$$

- Semi-supervised learning
- This is simply done by setting the source and target language variables lx and ly to the same value, in which case the model must attempt to reconstruct the input sentence, rather than translate it.

- Motivations

- Model

- Experiments

- BLEU score

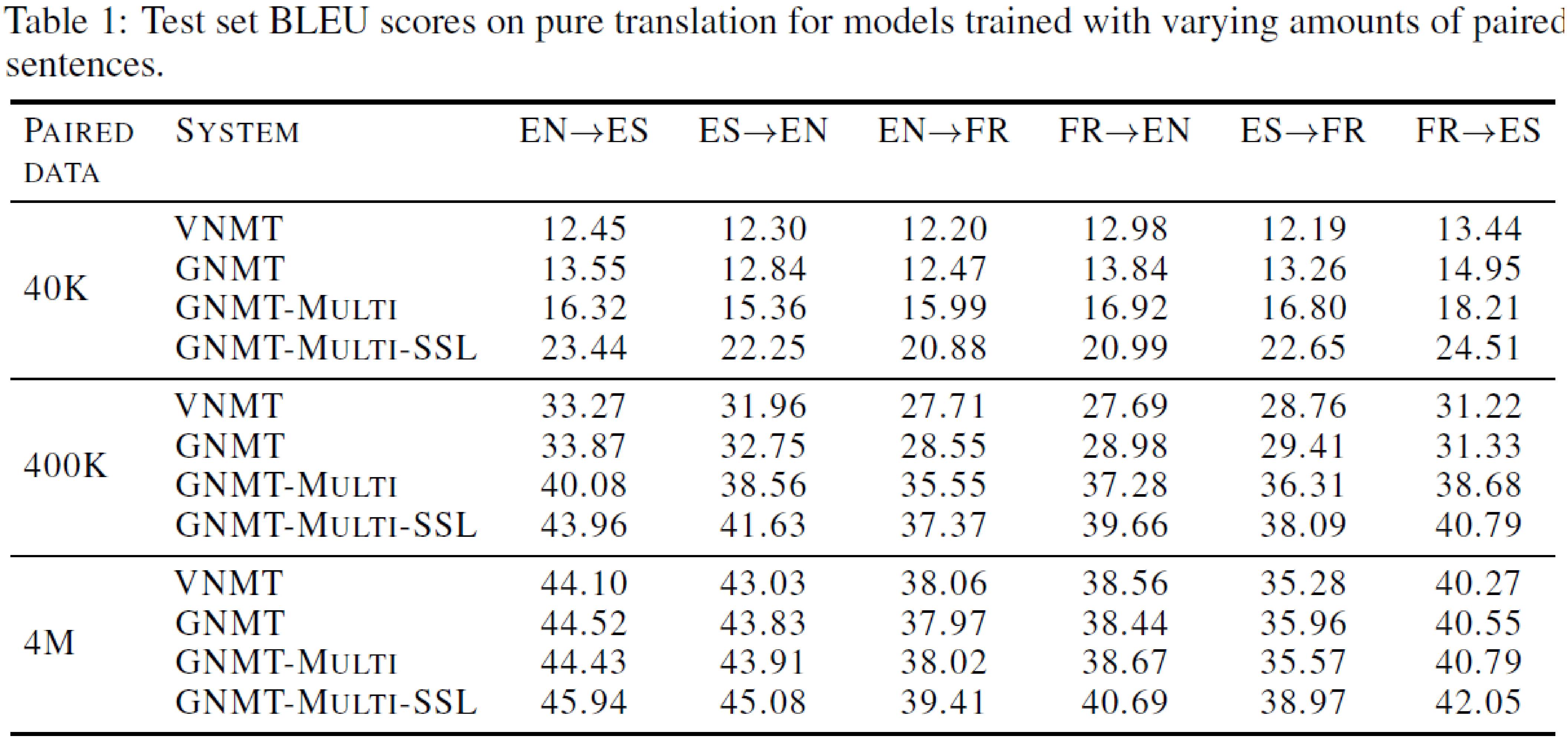


Table 1: Test set BLEU scores on pure translation for models trained with varying amounts of paired sentences.

Paired data	System	EN→ES	ES→EN	EN→FR	FR→EN	ES→FR	FR→ES
40K	VNMT	12.45	12.30	12.20	12.98	12.19	13.44
	GNMT	13.55	12.84	12.47	13.84	13.26	14.95
	GNMT-Multi	16.32	15.36	15.99	16.92	16.80	18.21
	GNMT-Multi-SSL	23.44	22.25	20.88	20.99	22.65	24.51
400K	VNMT	33.27	31.96	27.71	27.69	28.76	31.22
	GNMT	33.87	32.75	28.55	28.98	29.41	31.33
	GNMT-Multi	40.08	38.56	35.55	37.28	36.31	38.68
	GNMT-Multi-SSL	43.96	41.63	37.37	39.66	38.09	40.79
4M	VNMT	44.10	43.03	38.06	38.56	35.28	40.27
	GNMT	44.52	43.83	37.97	38.44	35.96	40.55
	GNMT-Multi	44.43	43.91	38.02	38.67	35.57	40.79
	GNMT-Multi-SSL	45.94	45.08	39.41	40.69	38.97	42.05

- KL divergence

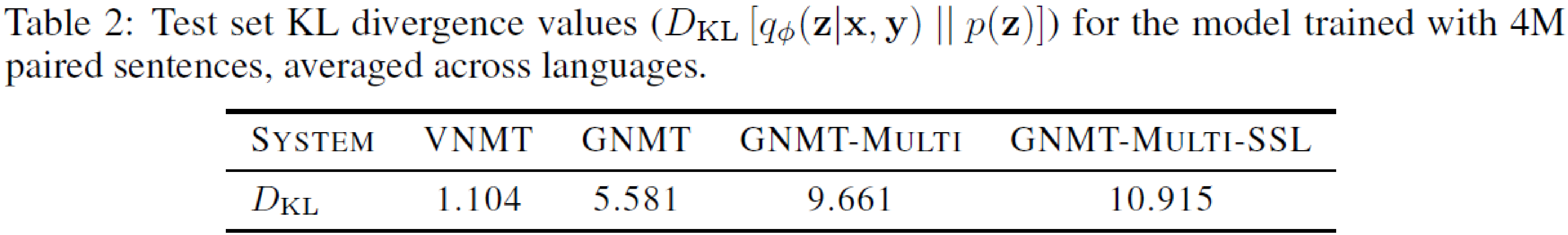


Table 2: Test set KL divergence values ($D_{\mathrm{KL}}\left[q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y}) \;||\; p(\mathbf{z})\right]$) for the model trained with 4M paired sentences, averaged across languages.

SYSTEM	VNMT	GNMT	GNMT-MULTI	GNMT-MULTI-SSL
D_{KL}	1.104	5.581	9.661	10.915

- The higher values for GNMT, GNMT-MULTI and GNMT-MULTISSL clearly indicate that these models are placing higher reliance on the latent variable than is VNMT.

- BLEU by sentence length

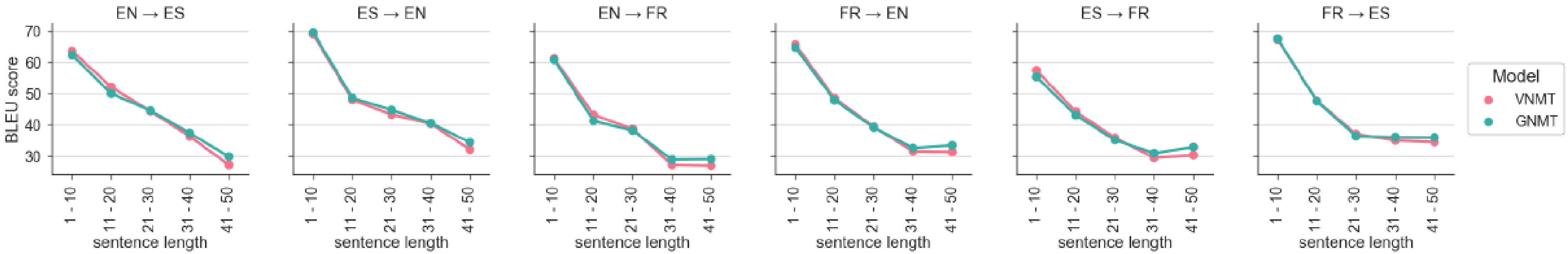


Figure 2: Test set BLEU scores on pure translation, by sentence length, evaluated using the model parameters trained with 4M paired sentences.

- The latent variable in GNMT is explicitly encouraged to model the sentence's semantics, it helps to alleviate 'coverage' issues.

- Missing word translation

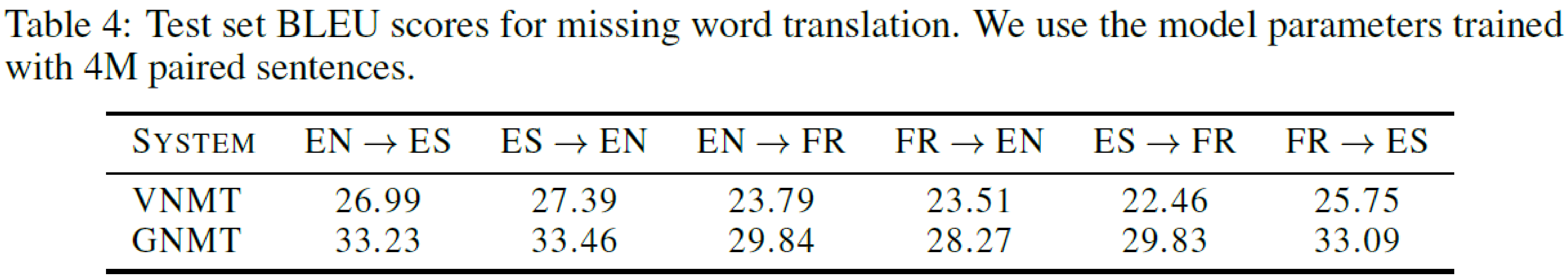


Table 4: Test set BLEU scores for missing word translation. We use the model parameters trained with 4M paired sentences.

SYSTEM	EN → ES	ES → EN	EN → FR	FR → EN	ES → FR	FR → ES
VNMT	26.99	27.39	23.79	23.51	22.46	25.75
GNMT	33.23	33.46	29.84	28.27	29.83	33.09

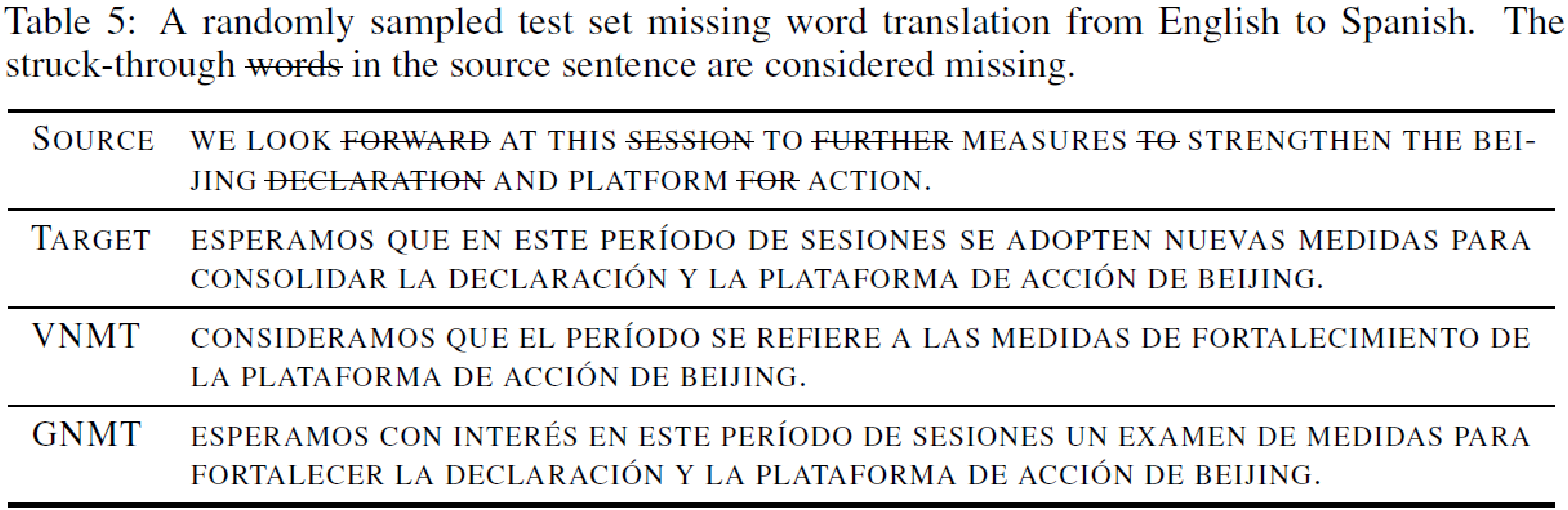


Table 5: A randomly sampled test set missing word translation from English to Spanish. The struck-through ~~words~~ in the source sentence are considered missing.

SOURCE	WE LOOK ~~FORWARD~~ AT THIS ~~SESSION~~ TO ~~FURTHER~~ MEASURES ~~TO~~ STRENGTHEN THE BEIJING ~~DECLARATION~~ AND PLATFORM ~~FOR~~ ACTION.
TARGET	ESPERAMOS QUE EN ESTE PERÍODO DE SESIONES SE ADOPTEN NUEVAS MEDIDAS PARA CONSOLIDAR LA DECLARACIÓN Y LA PLATAFORMA DE ACCIÓN DE BEIJING.
VNMT	CONSIDERAMOS QUE EL PERÍODO SE REFIERE A LAS MEDIDAS DE FORTALECIMIENTO DE LA PLATAFORMA DE ACCIÓN DE BEIJING.
GNMT	ESPERAMOS CON INTERÉS EN ESTE PERÍODO DE SESIONES UN EXAMEN DE MEDIDAS PARA FORTALECER LA DECLARACIÓN Y LA PLATAFORMA DE ACCIÓN DE BEIJING.

- Unseen language pair translation

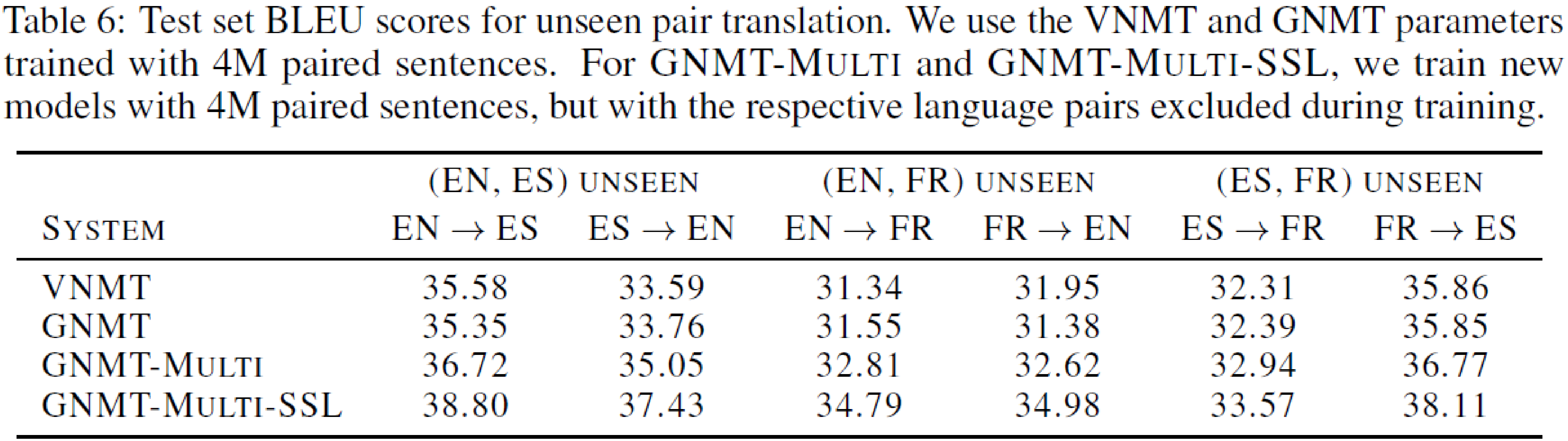


Table 6: Test set BLEU scores for unseen pair translation. We use the VNMT and GNMT parameters trained with 4M paired sentences. For GNMT-MULTI and GNMT-MULTI-SSL, we train new models with 4M paired sentences, but with the respective language pairs excluded during training.

	(EN, ES) UNSEEN		(EN, FR) UNSEEN		(ES, FR) UNSEEN	
SYSTEM	EN → ES	ES → EN	EN → FR	FR → EN	ES → FR	FR → ES
VNMT	35.58	33.59	31.34	31.95	32.31	35.86
GNMT	35.35	33.76	31.55	31.38	32.39	35.85
GNMT-MULTI	36.72	35.05	32.81	32.62	32.94	36.77
GNMT-MULTI-SSL	38.80	37.43	34.79	34.98	33.57	38.11

- GNMT models the semantic meaning of the source and target sentences:
  - Translating with missing words
  - Multilingual translation
  - Semi-supervised learning