Neural Machine Translation with V-AE

Baosong Yang

2018.03.12

Contents

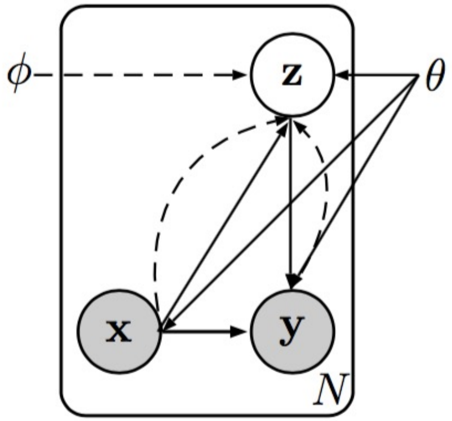
Variational Neural Machine Translation

Motivations

- The semantic representations are learned in an implicit way.
- Far from being sufficient for capturing all semantic details and dependencies.

- **Solution:** Incorporates the latent random variable z into NMT

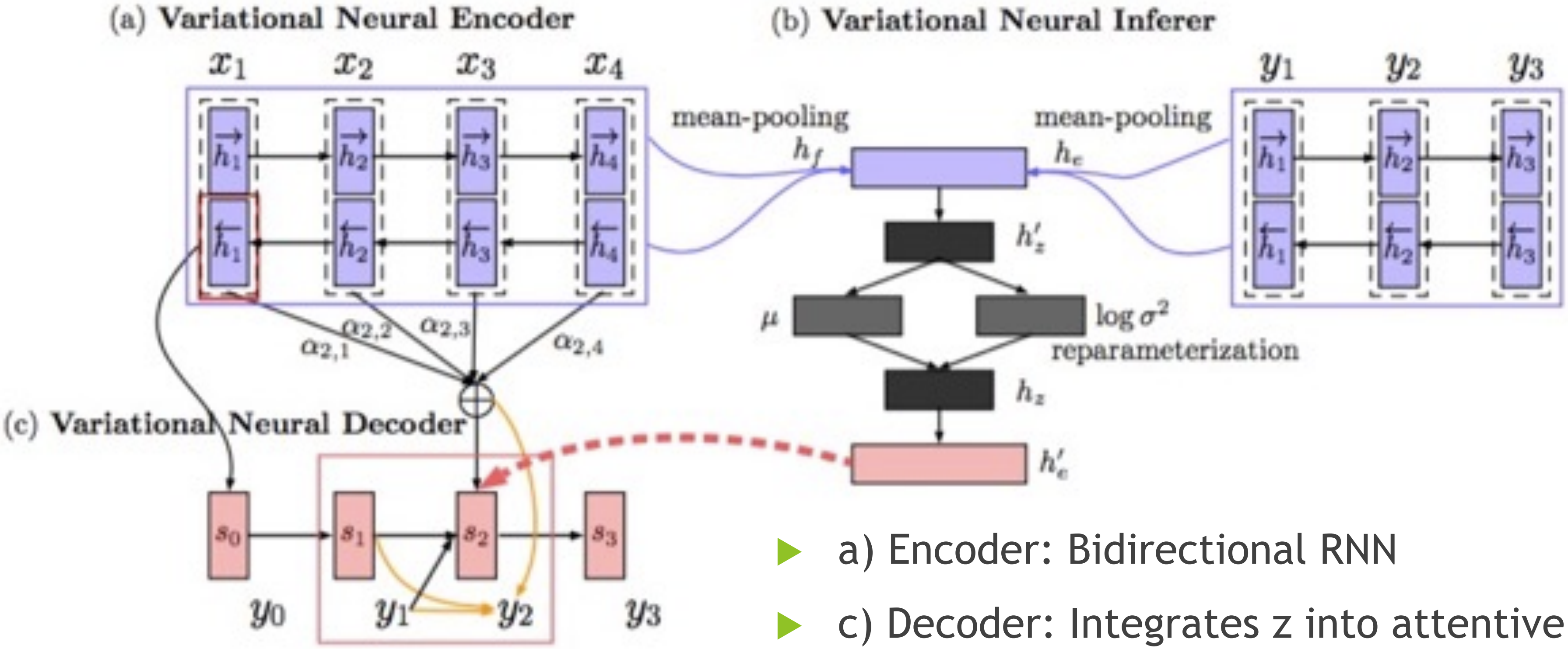
$$p(\mathbf{y}|\mathbf{x}) = \int_z p(\mathbf{y}, z|\mathbf{x})d_z = \int_z p(\mathbf{y}|z, \mathbf{x})p(z|\mathbf{x})d_z$$



- Solid lines:
  - Generative model $p_\theta(\mathbf{z}|\mathbf{x})p_\theta(\mathbf{y}|\mathbf{z}, \mathbf{x})$
- Dashed lines:
  - Variational approxiamation $q_\phi(\mathbf{z}|\mathbf{x})$
  - Intractable posterior $p(\mathbf{z}|\mathbf{x}, \mathbf{y})$

Variational Neural Machine Translation

Model



- a) Encoder: Bidirectional RNN
- c) Decoder: Integrates z into attentive decoder

Variational Neural Machine Translation

Variational Neural Inferer

- Infers the representation of z according to
  - 1） the learned source representations; $p_\theta(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu'(\mathbf{x}), \sigma'(\mathbf{x})^2\mathbf{I})$
  - 2） target ones. $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y}) = \mathcal{N}(\mathbf{z}; \mu(\mathbf{x},\mathbf{y}), \sigma(\mathbf{x},\mathbf{y})^2\mathbf{I})$
- Obtain z:
  - **Reparameterization** reparameterizes z as a function of μ and σ, rather than using the standard sampling method.

$$\mathbf{h}_z = \mu + \sigma \odot \epsilon, \epsilon \sim \mathcal{N}(0, \mathbf{I})$$

- Variational Lower Bound -> Training: Monte Carlo (L = 1)

$$\mathcal{L}_{\text{VNMT}}(\theta, \phi; \mathbf{x}, \mathbf{y}) = -\text{KL}(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})||p_\theta(\mathbf{z}|\mathbf{x})) + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}[\log p_\theta(\mathbf{y}|\mathbf{z},\mathbf{x})]$$

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}[\cdot] \simeq \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{h}_z^{(l)})$$

Variational Neural Machine Translation

Experiments

- 2.9M LDC Zh-En

System	MT05	MT02	MT03	MT04	MT06	MT08	AVG
Moses	33.68	34.19	34.39	35.34	29.20	22.94	31.21
GroundHog	31.38	33.32	32.59	35.05	29.80	22.82	30.72
VNMT w/o KL	31.40	33.50	32.92	34.95	28.74	22.07	30.44
VNMT	32.25	**34.50**$^{++}$	33.78^{++}	**36.72**$^{\Uparrow++}$	**30.92**$^{\Uparrow++}$	**24.41**$^{\uparrow++}$	**32.07**

- 4.5M WMT14 En-De (Evaluated on newstest14)

System	Architecture	BLEU
	Existing end-to-end NMT systems	
Jean et al. (2015)	RNNSearch	16.46
Jean et al. (2015)	RNNSearch + unk replace	18.97
Jean et al. (2015)	RNNsearch + unk replace + large vocab	19.40
Luong et al. (2015a)	LSTM with 4 layers + dropout + local att. + unk replace	20.90
	Our end-to-end NMT systems	
this work	RNNSearch	16.40
	VNMT	17.13^{++}
	VNMT + unk replace	19.58^{++}

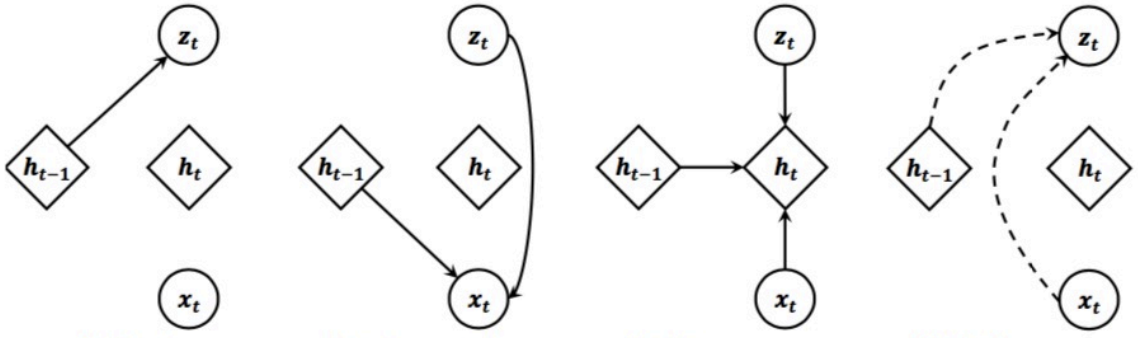
Variational Recurrent Neural Machine Translation

Motivations

- The internal transition structure of VNMT is entirely deterministic, and hence, this implementation may not be an effective way to model high variability

- **Solution:** VRNMT is adopted continuous latent random variable sequence.

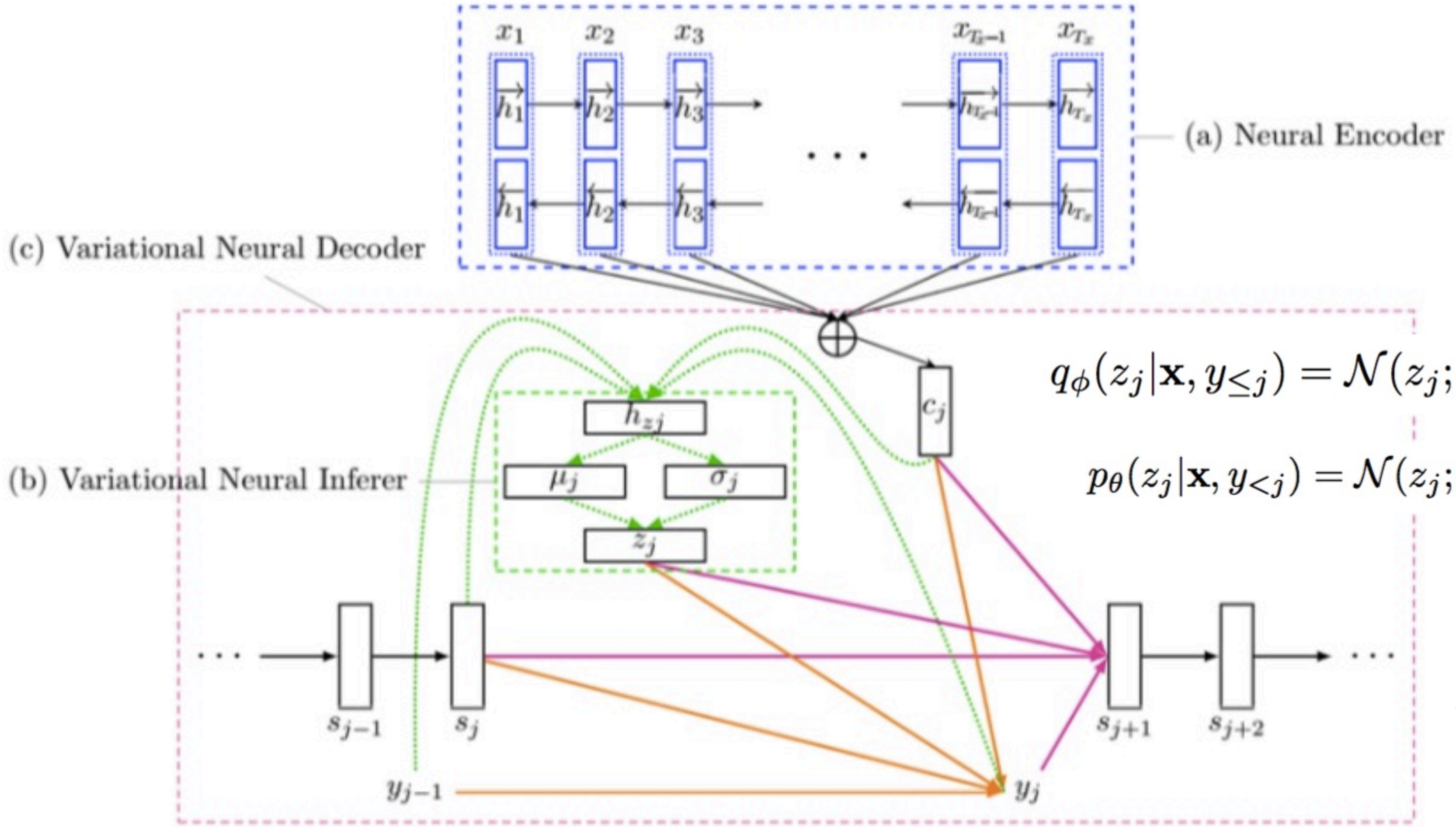
$$p(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{T_y} p(y_j|\mathbf{x}, y_{<j}) = \prod_{j=1}^{T_y} \int_{z_j} p(y_j, z_j|\mathbf{x}, y_{<j}) d_{z_j}$$
$$= \prod_{j=1}^{T_y} \int_{z_j} p(y_j|\mathbf{x}, y_{<j}, z_j) p(z_j|\mathbf{x}, y_{<j}) d_{z_j}$$



(a) Prior (b) Generation (c) Recurrence (d) Inference

Variational Recurrent Neural Machine Translation

Model



$$q_\phi(z_j|\mathbf{x}, y_{\leq j}) = \mathcal{N}(z_j; \mu_j(\mathbf{x}, y_{\leq j}), \sigma_j(\mathbf{x}, y_{\leq j})^2\mathbf{I})$$

$$p_\theta(z_j|\mathbf{x}, y_{<j}) = \mathcal{N}(z_j; \mu'_j(\mathbf{x}, y_{<j}), \sigma'_j(\mathbf{x}, y_{<j})^2\mathbf{I})$$

Variational Recurrent Neural Machine Translation

Experiments

- 1.25M LDC Zh-En

System	MT03	MT04	MT05	MT06	Ave.
COVERAGE	34.49	38.34	34.91	34.25	35.50
MemDec	35.09	37.73	35.53	34.32	35.67
DeepLAU	36.16	39.81	35.91	35.98	36.97
DMAtten	**38.33**	40.11	36.71	35.29	37.61
Moses	32.93	34.76	31.31	31.05	32.51
DL4MT	36.59	39.57	35.56	35.29	36.75
VNMT	37.23	40.32	36.28	35.73	37.39
VRNMT(-TD)	36.97	40.07	36.13	35.49	37.17
VRNMT	38.08^{*}_{++}	$\mathbf{41.07}^{**}_{++}$	$\mathbf{36.82}^{**}_{++}$	$\mathbf{36.72}^{*}_{++}$	**38.17**

- 4.46M WMT14 En-De (Evaluated on newstest15)

System	BLEU
BPEChar	23.9
RecAtten	25.0
ConvEncoder	24.2
Moses	20.54
DL4MT	24.88
VNMT	25.49
VRNMT(-TD)	25.34
VRNMT	$\mathbf{25.93}^{*}_{++}$

Conclusions

- Discussions:
  - The papers improved NMT models with VAE to capture the underlying semantics of sentence pairs.
  - How to better exploit latent variables may be the future direction under this category.
  - An alternative solution for the post-edit NMT (using latent variable to model the edit vector)
- Shortages:
  - Approach 1 can not model the variants in each time step.
  - Approach 2 can not model the future context of the target.