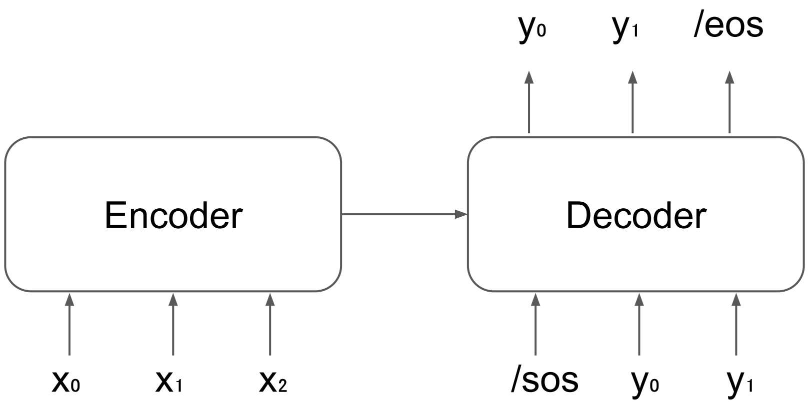
Non-Autoregressive Island in Autoregressive World

Mikhail Arkhipov
MIPT

Autoregressive Approaches

- Use probability chain rule to factorize joint distribution
- Intrinsically sequential
- Exposure Bias due to Teacher Forcing
- Best metrics among other generative models

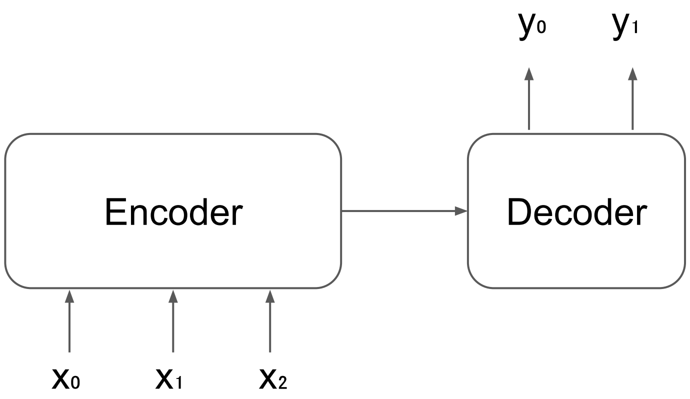


$$p(\mathbf{y}|\mathbf{x}) = \prod_i p(y_i|\mathbf{y}_{<i}, \mathbf{x}) = p(y_0|\mathbf{x})p(y_1|y_0, \mathbf{x})p(y_2|y_0, y_1, \mathbf{x})\ldots$$

[Gu et al. 18] https://arxiv.org/pdf/1711.02281.pdf

Non-Autoregressive Approaches

- Assume that target probability factorizes
- Intrinsically parallel
- Lower quality compared to autoregressive counterparts



$$p(\mathbf{y}|\mathbf{x}) = \prod_i p(y_i|\mathbf{x}) = p(y_0|\mathbf{x})p(y_1|\mathbf{x})p(y_2|\mathbf{x})\ldots$$

[Gu et al. 18] https://arxiv.org/pdf/1711.02281.pdf

Noisy Parallel Approximate Decoding for Conditional Recurrent Language Model

$$\mathbf{h}_t = \phi\left(\mathbf{h}_{t-1} + \epsilon_t, \mathbf{E}\left[x_t\right], f(Y, t)\right),$$

$$\epsilon_t \sim \mathcal{N}(\mathbf{0}, \sigma_t^2 \mathbf{I}).$$

$$\sigma_t = \frac{\sigma_0}{t},$$

[Cho 16] https://arxiv.org/abs/1605.03835

Noisy Parallel Approximate Decoding for Conditional Recurrent Language Model

Strategy	# Parallels	Valid		Test-1	
		NLL↓	BLEU↑	NLL↓	BLEU↑
Greedy	1	27.879	15.5	26.4928	16.66
NPAD	5	21.5984	16.09	24.3863	17.51
NPAD	10	21.054	16.33	23.6942	17.81
NPAD	<u>50</u>	**20.4463**	**16.71**	<u>23.0111</u>	<u>18.03</u>

[Cho 16] https://arxiv.org/abs/1605.03835

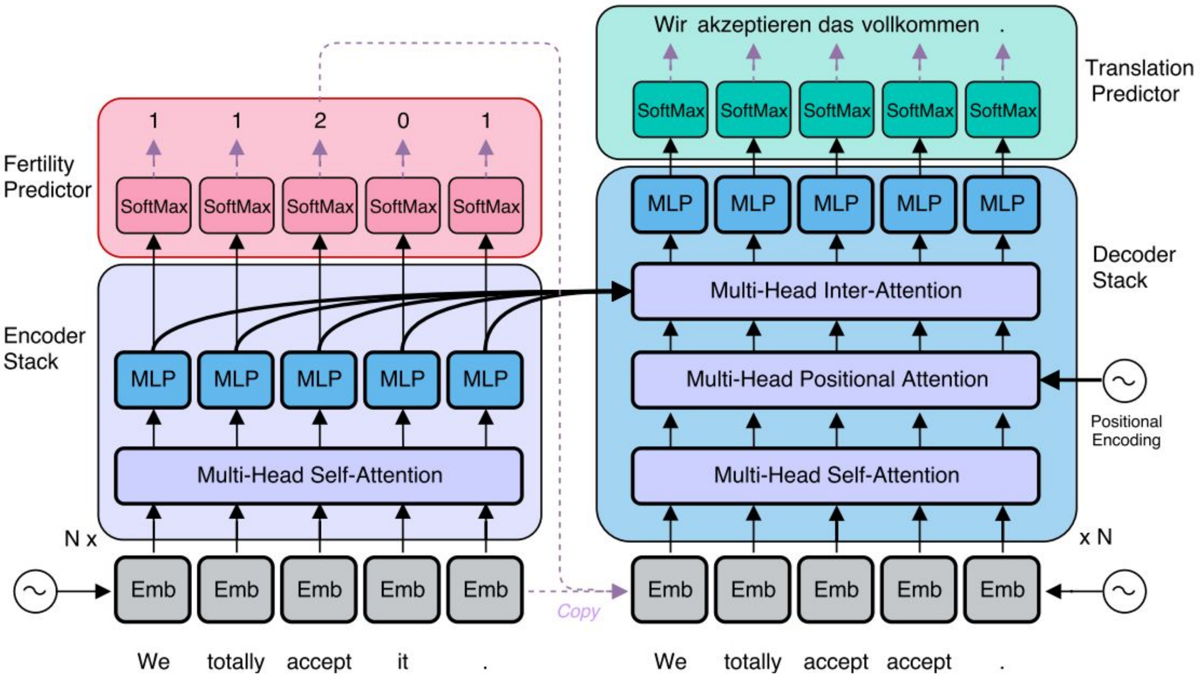
Noisy Parallel Approximate Decoding for Conditional Recurrent Language Model

Strategy	σ_0	Valid NLL↓	Valid BLEU↑	Test-1 NLL↓	Test-1 BLEU↑
Greedy	-	27.879	15.5	26.4928	16.66
Sto. Sampling	-	22.9818	15.64	26.2536	16.76
NPAD	0.1	21.125	16.06	23.8542	17.48
NPAD	0.2	20.6353	16.37	23.2631	17.86
NPAD	0.3	**20.4463**	**16.71**	23.0111	18.03
NPAD	0.5	20.7648	16.48	23.3056	18.13

Table 1: Effect of noise injection. For both stochastic sampling and NPAD, we used 50 parallel samplers. For NPAD, we used the greedy decoding as an inner-decoding strategy.

[Cho 16] https://arxiv.org/abs/1605.03835

Non-Autoregressive Neural Machine Translation



[Gu et al. 18] https://arxiv.org/pdf/1711.02281.pdf

Non-Autoregressive Neural Machine Translation

$$p_{\mathcal{NA}}(Y|X;\theta) = \sum_{f_1,\ldots,f_{T'} \in \mathcal{F}} \left(\prod_{t'=1}^{T'} p_F(f_{t'}|x_{1:T'};\theta) \cdot \prod_{t=1}^{T} p(y_t|x_1\{f_1\},..,x_{T'}\{f_{T'}\};\theta) \right)$$

$$\mathcal{L}_{\text{FT}} = \lambda \left(\underbrace{\mathop{\mathbb{E}}_{f_{1:T'} \sim p_F} \left(\mathcal{L}_{\text{RKL}}\left(f_{1:T'}\right) - \mathcal{L}_{\text{RKL}}\left(\bar{f}_{1:T'}\right) \right)}_{\mathcal{L}_{\text{RL}}} + \underbrace{\mathop{\mathbb{E}}_{f_{1:T'} \sim q} \left(\mathcal{L}_{\text{RKL}}\left(f_{1:T'}\right) \right)}_{\mathcal{L}_{\text{BP}}} \right) + (1-\lambda)\mathcal{L}_{\text{KD}}$$

[Gu et al. 18] https://arxiv.org/pdf/1711.02281.pdf

Non-Autoregressive Neural Machine Translation

Models	WMT14		WMT16		IWSLT16		
	En→De	De→En	En→Ro	Ro→En	En→De	Latency / Speedup	
NAT	17.35	20.62	26.22	27.83	25.20	39 ms	15.6×
NAT (+FT)	17.69	21.47	27.29	29.06	26.52	39 ms	15.6×
NAT (+FT + NPD $s = 10$)	18.66	22.41	29.02	30.76	27.44	79 ms	7.68×
NAT (+FT + NPD $s = 100$)	19.17	23.20	29.79	**31.44**	28.16	257 ms	2.36×
Autoregressive ($b = 1$)	22.71	26.39	31.35	31.03	28.89	408 ms	1.49×
Autoregressive ($b = 4$)	23.45	27.02	31.91	31.76	29.70	607 ms	1.00×

[Gu et al. 18] https://arxiv.org/pdf/1711.02281.pdf

Mask-Predict: Parallel Decoding of Conditional Masked Language Models

src	Der Abzug der franzsischen Kampftruppen wurde am 20. November abgeschlossen .
$t = 0$	The departure of the French combat completed completed on 20 November .
$t = 1$	The departure of French combat troops was completed on 20 November .
$t = 2$	The withdrawal of French combat troops was completed on November 20th .

Unobserved tokens

$$y_i^{(t)} = \arg\max_w P(y_i = w | X, Y_{obs}^{(t)})$$

$$p_i^{(t)} = \max_w P(y_i = w | X, Y_{obs}^{(t)})$$

Observed tokens

$$y_i^{(t)} = y_i^{(t-1)}$$

$$p_i^{(t)} = p_i^{(t-1)}$$

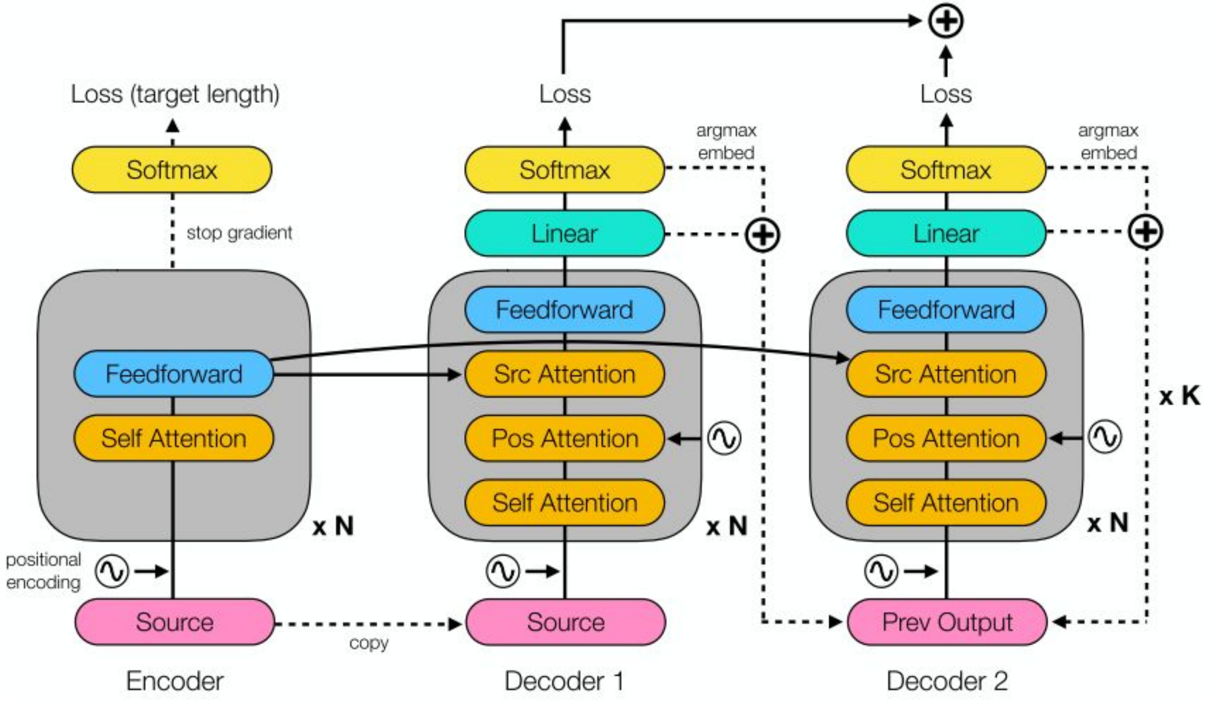
[Ghazvininejad et al. 19] Constant-time machine translation with conditional masked language models

Mask-Predict: Parallel Decoding of Conditional Masked Language Models

Model	Dimensions (Model/Hidden)	Iterations	WMT'14 EN-DE	WMT'14 DE-EN	WMT'16 EN-RO	WMT'16 RO-EN
NAT w/ Fertility (Gu et al., 2018)	512/512	1	19.17	23.20	29.79	31.44
CTC Loss (Libovický and Helcl, 2018)	512/4096	1	17.68	19.80	19.93	24.71
Iterative Refinement (Lee et al., 2018)	512/512	1	13.91	16.77	24.45	25.73
	512/512	10	21.61	25.48	29.32	30.19
(Dynamic #Iterations)	512/512	?	21.54	25.43	29.66	30.30
Small CMLM with Mask-Predict	512/512	1	15.06	19.26	20.12	20.36
	512/512	4	**24.17**	**28.55**	**30.00**	30.43
	512/512	10	**25.51**	**29.47**	**31.65**	**32.27**
Base CMLM with Mask-Predict	512/2048	1	18.05	21.83	27.32	28.20
	512/2048	4	**25.94**	**29.90**	**32.53**	**33.23**
	512/2048	10	**27.03**	**30.53**	**33.08**	**33.31**
Base Transformer (Vaswani et al., 2017)	512/2048	N	27.30	——	——	——
Base Transformer (Our Implementation)	512/2048	N	27.74	31.09	34.28	33.99
Base Transformer (+Distillation)	512/2048	N	27.86	31.07	——	——
Large Transformer (Vaswani et al., 2017)	1024/4096	N	28.40	——	——	——
Large Transformer (Our Implementation)	1024/4096	N	28.60	31.71	——	——

[Ghazvininejad et al. 19] Constant-time machine translation with conditional masked language models

Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement



[Lee, Mansimov, Cho 18] [Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement]

Training objective

$$J_{\mathrm{LVM}}(\theta) = -\sum_{l=0}^{L+1} \left(\sum_{t=1}^{T} \log p_\theta(y_t^* | \hat{Y}^{l-1}, X) \right),$$

$$J_{\mathrm{DAE}}(\theta) = -\sum_{t=1}^{T} \log p_\theta(y_t^* | \tilde{Y}, X).$$

$$\begin{aligned} J(\theta) = -\sum_{l=0}^{L+1} \Bigg(& \alpha_l \sum_{t=1}^{T} \log p_\theta(y_t^* | \hat{Y}^{l-1}, X) \qquad (4) \\ & + (1-\alpha_l) \sum_{t=1}^{T} \log p_\theta(y_t^* | \tilde{Y}, X) \Bigg), \end{aligned}$$

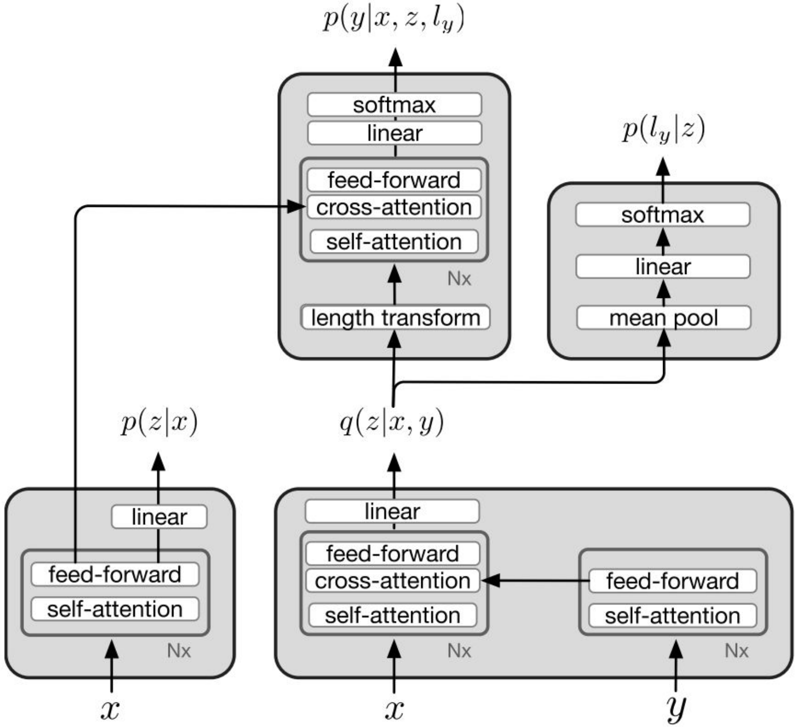
[Lee, Mansimov, Cho 18] Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement

Results

		IWSLT'16 En-De				WMT'16 En-Ro				WMT'14 En-De				MS COCO		
		En→	De→	GPU	CPU	En→	Ro→	GPU	CPU	En→	De→	GPU	CPU	BLEU	GPU	CPU
AR	$b = 1$	28.64	34.11	70.3	32.2	31.93	31.55	55.6	15.7	23.77	28.15	54.0	15.8	23.47	4.3	2.1
	$b = 4$	28.98	34.81	63.8	14.6	32.40	32.06	43.3	7.3	24.57	28.47	44.9	7.0	24.78	3.6	1.0
NAT	FT	26.52	–	–	–	27.29	29.06	–	–	17.69	21.47	–	–	–	–	–
	FT+NPD	28.16	–	–	–	29.79	31.44	–	–	19.17	23.30	–	–	–	–	–
Our Model	$i_{dec} = 1$	22.20	27.68	573.0	213.2	24.45	25.73	694.2	98.6	13.91	16.77	511.4	83.3	20.12	17.1	8.9
	$i_{dec} = 2$	24.82	30.23	423.8	110.9	27.10	28.15	332.7	62.8	16.95	20.39	393.6	49.6	20.88	12.0	5.7
	$i_{dec} = 5$	26.58	31.85	189.7	52.8	28.86	29.72	194.4	29.0	20.26	23.86	139.7	23.1	21.12	6.2	2.8
	$i_{dec} = 10$	27.11	32.31	98.8	24.1	29.32	30.19	93.1	14.8	21.61	25.48	90.4	12.3	21.24	2.0	1.2
	Adaptive	27.01	32.43	125.9	29.3	29.66	30.30	118.3	16.5	21.54	25.43	107.2	20.3	21.12	10.8	4.8

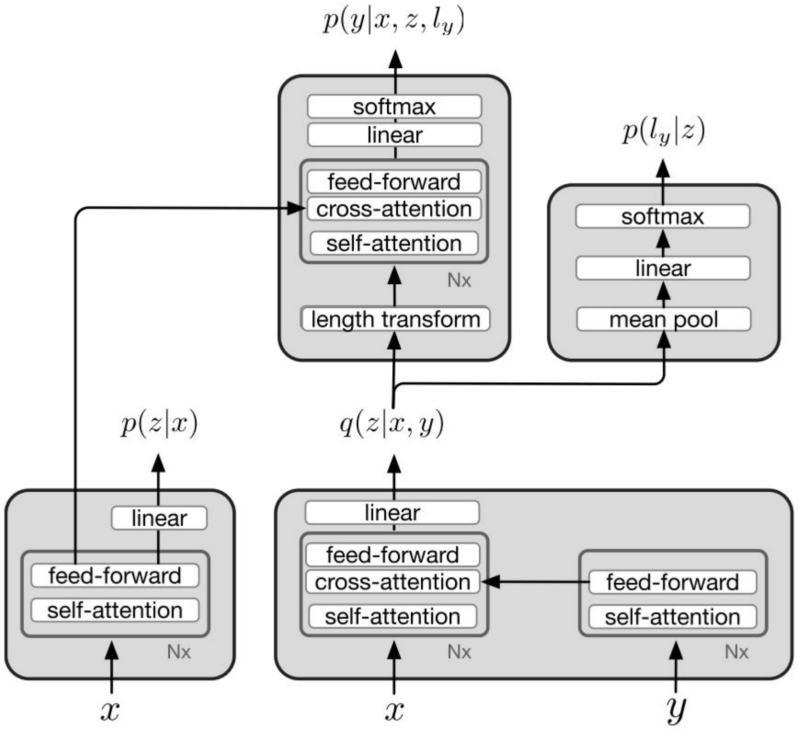
[Lee, Mansimov, Cho 18] Deterministic Non-Autoregressive Neural Sequence Modeling by Iterative Refinement

Latent-Variable Non-Autoregressive Neural Machine Translation with Deterministic Inference Using a Delta Posterior



[Shu et al. 19] [Latent-Variable Non-Autoregressive Neural Machine Translation]

Training



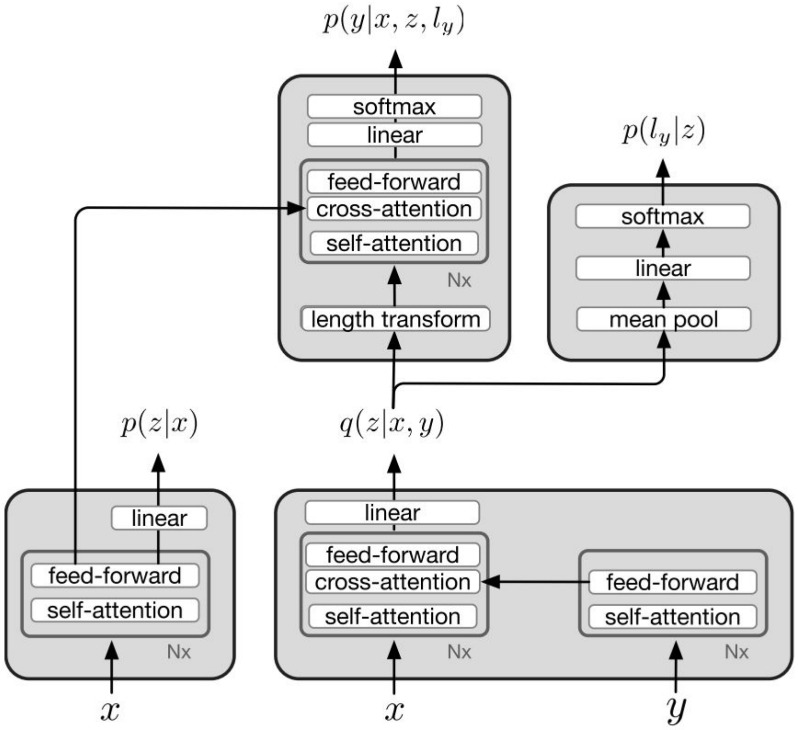
$$\log p(y|x) = \log \int p(y|z,x)p(z|x)dz$$

$$\mathcal{L}(\omega, \phi, \theta) = \mathbb{E}_{z\sim q_\phi}\left[\log p_\theta(y|x,z)\right] - \mathrm{KL}\left[q_\phi(z|x,y)||p_\omega(z|x)\right]$$

$$\mathbb{E}_{z\sim q_\phi}\left[\sum_{i=1}^{|y|}\log p_\theta(y_i|x,z,l_y) + \log p_\theta(l_y|z)\right] - \sum_{k=1}^{|x|}\mathrm{KL}\left[q_\phi(z_k|x,y)||p_\omega(z_k|x)\right].$$

[Shu et al. 19] [Latent-Variable Non-Autoregressive Neural Machine Translation]

Inference



Algorithm 1 Deterministic Iterative Inference

Inputs:

x : source sentence

T : maximum step

$\mu_0 = \mathbb{E}_{p_\omega(z|x)}[z]$

$y_0 = \text{argmax}_y \log p_\theta(y|x, z = \mu_0)$

for $t \leftarrow 1$ to T **do**

$\mu_t = \mathbb{E}_{q_\phi(z|x, y_{t-1})}[z]$

$y_t = \text{argmax}_y \log p_\theta(y|x, z = \mu_t)$

if $y_t = y_{t-1}$ **then**

break

output y_t

[Shu et al. 19] Latent-Variable Non-Autoregressive Neural Machine Translation

Posterior Collapse. KL -> 0

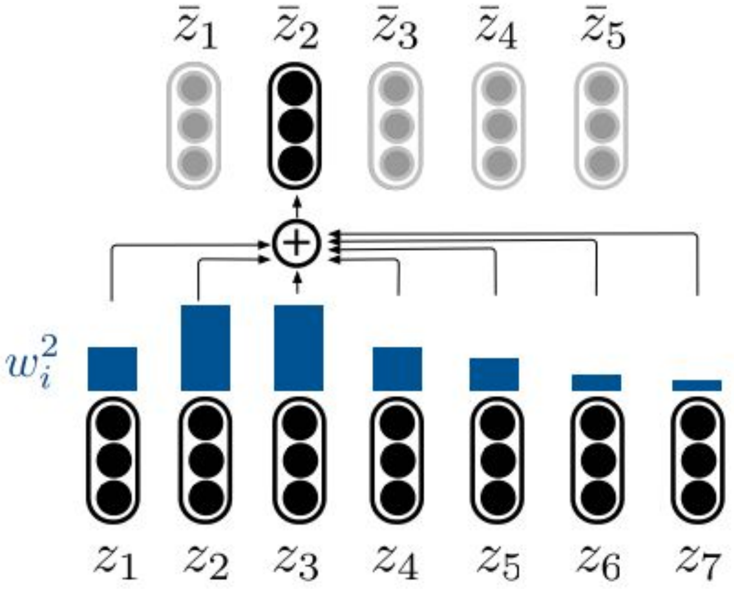
$$q_\phi(z|x) \simeq q_\phi(z) = \mathcal{N}(a, b)$$

$$\sum_{k=1}^{|x|} \max(b, \mathrm{KL}\left[q_\phi(z_k|x, y)||p_\omega(z_k|x)\right]),$$

$$b = \begin{cases} 1, & \text{if } s < M/2 \\ \frac{(M-s)}{M/2}, & \text{otherwise} \end{cases}$$

[Shu et al. 19] Latent-Variable Non-Autoregressive Neural Machine Translation

Tackle Target Length



$$\bar{z}_j = \sum_{k=1}^{|x|} w_k^j z_k,$$

$$w_k^j = \frac{\exp(a_k^j)}{\sum_{k'=1}^{|x|} \exp(a_{k'}^j)},$$

$$a_k^j = -\frac{1}{2\sigma^2}\left(k - \frac{|x|}{l_y} j\right)^2,$$

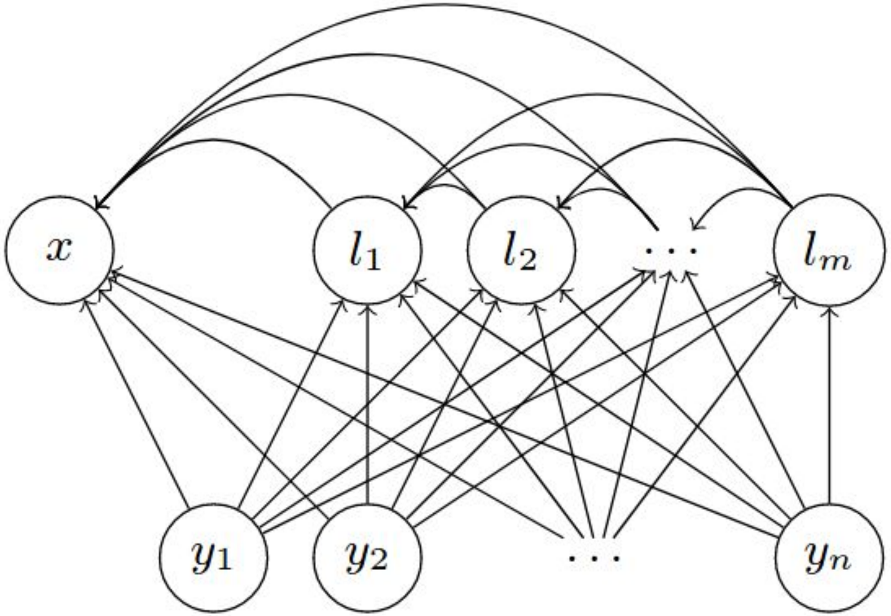
[Shu et al. 19] Latent-Variable Non-Autoregressive Neural Machine Translation

Results

	ASPEC Ja-En			WMT'14 En-De		
	BLEU(%)	speedup	wall-clock (std)	BLEU(%)	speedup	wall-clock (std)
Base Transformer, beam size=3	27.1	1x	415ms (159)	26.1	1x	602ms (274)
Base Transformer, beam size=1	24.6	1.1x	375ms (150)	25.6	1.3x	461ms (219)
Latent-Variable NAR Model	13.3	17.0x	24ms (2)	11.8	22.2x	27ms (1)
+ knowledge distillation	25.2	17.0x	24ms (2)	22.2	22.2x	27ms (1)
+ deterministic inference	27.5	8.6x	48ms (2)	24.1	12.5x	48ms (8)
+ latent search	28.3	4.8x	86ms (2)	25.1	6.8x	88ms (8)

[Shu et al. 19] Latent-Variable Non-Autoregressive Neural Machine Translation

Fast Decoding in Sequence Models Using Discrete Latent Variables



- The function $\mathrm{ae}(y, x)$ will autoencode y into a shorter sequence $l = l_1, \ldots, l_m$ of discrete latent variables using the discretization bottleneck from Section 2.
- The latent prediction model $\mathrm{lp}(x)$ (a Transformer) will autoregressively predict l based on x.
- The decoder $\mathrm{ad}(l, x)$ is a parallel model that will decode y from l and the input sequence x.

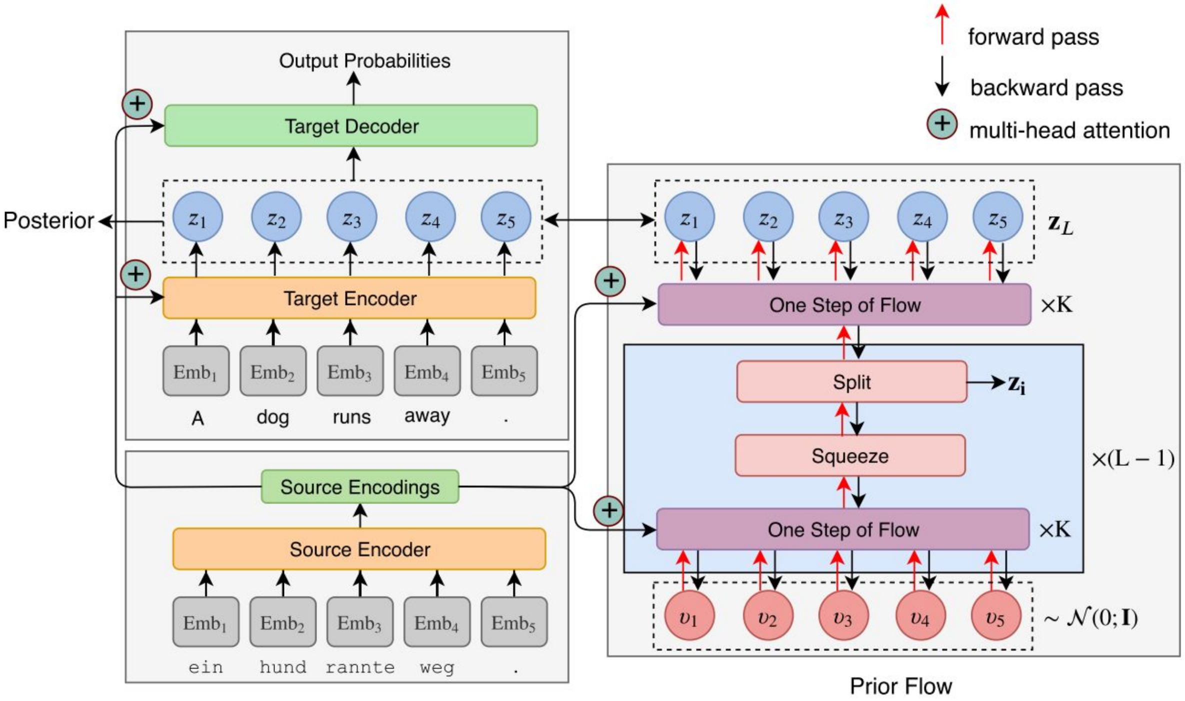
[Kaiser et al. 18] https://arxiv.org/pdf/1803.03382.pdf

Fast Decoding in Sequence Models Using Discrete Latent Variables

Model	**BLEU**
Baseline Transformer [1]	27.3
Baseline Transformer [2]	23.5
Baseline Transformer [2] (no beam-search)	22.7
NAT+FT (no NPD) [2]	17.7
LT without rescoring $(\frac{n}{m} = 8)$	19.8
NAT+FT (NPD rescoring 10) [2]	18.7
LT rescornig top-10 $(\frac{n}{m} = 8)$	21.0
NAT+FT (NPD rescoring 100) [2]	19.2
LT rescornig top-100 $(\frac{n}{m} = 8)$	22.5

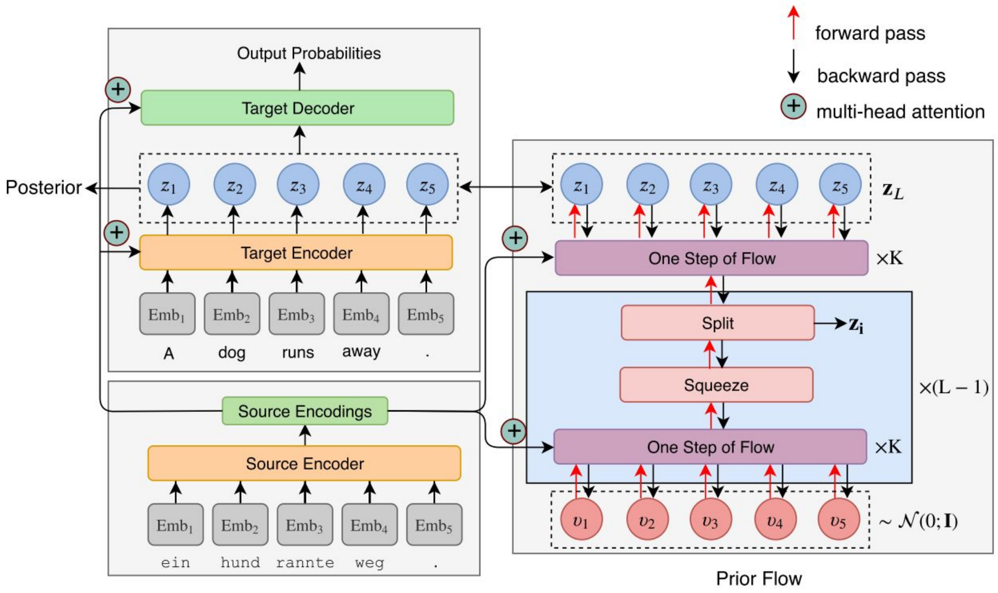
[Kaiser et al. 18] https://arxiv.org/pdf/1803.03382.pdf

FlowSeq: Non-Autoregressive Conditional Sequence Generation with Generative Flow



[Ma et al. 19] FlowSeq: Non-Autoregressive Conditional Sequence Generation with Generative Flow

Training



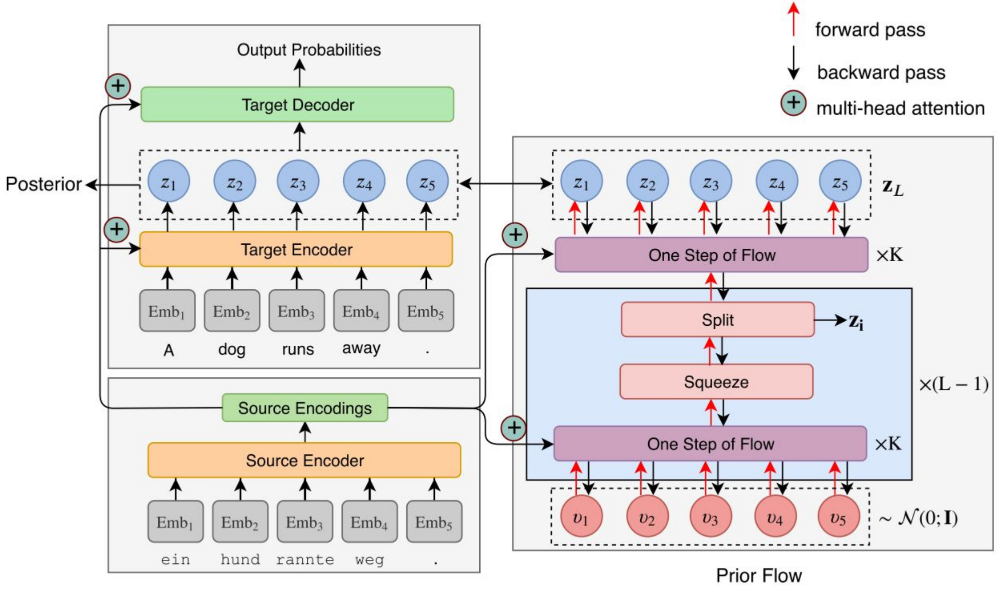
$$\log P_\theta(\mathbf{y}|\mathbf{x}) \geq \mathrm{E}_{q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x})}[\log P_\theta(\mathbf{y}|\mathbf{z},\mathbf{x})] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})).$$

$$q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x}) = \prod_{t=1}^{T} \mathcal{N}(\mathbf{z}_t|\mu_t(\mathbf{x},\mathbf{y}), \sigma_t^2(\mathbf{x},\mathbf{y}))$$

- Zero initialization
- Token dropout

[Ma et al. 19] [FlowSeq: Non-Autoregressive Conditional Sequence Generation with Generative Flow]

Inference



- argmax decoding

$$\mathbf{z}^* = \operatorname*{argmax}_{\mathbf{z} \in \mathcal{Z}} p_\theta(\mathbf{z}|\mathbf{x})$$
$$\mathbf{y}^* = \operatorname*{argmax}_{\mathbf{y}} P_\theta(\mathbf{y}|\mathbf{z}^*, \mathbf{x})$$

- NPD decoding:
Sample length (from src encoder) and latent, rescore with AR

- Importance Weighted

$$\mathbf{z}_i \sim p_\theta(\mathbf{z}|\mathbf{x}), \forall i = 1, \ldots, N$$
$$\hat{\mathbf{y}}_i = \operatorname*{argmax}_{\mathbf{y}} P_\theta(\mathbf{y}|\mathbf{z}_i, \mathbf{x})$$
$$\mathbf{z}_i^{(k)} \sim q_\phi(\mathbf{z}|\hat{\mathbf{y}}_i, \mathbf{x}), \forall k = 1, \ldots, K$$
$$P(\hat{\mathbf{y}}_i|\mathbf{x}) \approx \frac{1}{K} \sum_{k=1}^{K} \frac{P_\theta(\hat{\mathbf{y}}_i|\mathbf{z}_i^{(k)}, \mathbf{x}) p_\theta(\mathbf{z}_i^{(k)}|\mathbf{x})}{q_\phi(\mathbf{z}_i^{(k)}|\hat{\mathbf{y}}_i, \mathbf{x})}$$

[Ma et al. 19] FlowSeq

Results

Models	WMT2014 EN-DE	WMT2014 DE-EN	WMT2016 EN-RO	WMT2016 RO-EN
Autoregressive Methods				
Transformer-base	27.30	–	–	–
Our Implementation	27.16	31.44	32.92	33.09
Raw Data				
CMLM-base (refinement 4)	22.06	–	30.89	–
CMLM-base (refinement 10)	**24.65**	–	**32.53**	–
FlowSeq-base (IWD $n = 15$)	20.20	24.63	30.61	31.50
FlowSeq-base (NPD $n = 15$)	20.81	25.76	31.38	32.01
FlowSeq-base (NPD $n = 30$)	21.15	26.04	31.74	32.45
FlowSeq-large (IWD $n = 15$)	22.94	27.16	31.08	32.03
FlowSeq-large (NPD $n = 15$)	23.14	27.71	31.97	32.46
FlowSeq-large (NPD $n = 30$)	23.64	**28.29**	32.35	**32.91**
Knowledge Distillation				
NAT-IR (refinement 10)	21.61	25.48	29.32	30.19
NAT w/ FT (NPD $n = 10$)	18.66	22.42	29.02	31.44
NAT-REG (NPD $n = 9$)	24.61	28.90	–	–
LV NAR (refinement 4)	24.20	–	–	–
CMLM-small (refinement 10)	25.51	29.47	31.65	32.27
CMLM-base (refinement 10)	**26.92**	**30.86**	**32.42**	**33.06**
FlowSeq-base (IWD $n = 15$)	22.49	27.40	30.59	31.58
FlowSeq-base (NPD $n = 15$)	23.08	28.07	31.35	32.11
FlowSeq-base (NPD $n = 30$)	23.48	28.40	31.75	32.49
FlowSeq-large (IWD $n = 15$)	24.70	29.44	31.02	31.97
FlowSeq-large (NPD $n = 15$)	25.03	30.48	31.89	32.43
FlowSeq-large (NPD $n = 30$)	25.31	30.68	32.20	32.84

[Ma et al. 19] FlowSeq

On the Discrepancy between Density Estimation and Sequence Generation

		BLEU (↑)		LL (↑)	
		Raw	Dist.	Raw	Dist.
WMT'14 En→De	Tr-S	24.54	24.94	-1.77	-2.36
	Tr-B	28.18	27.86	-1.44	-2.19
	Tr-L	<u>29.39</u>	28.29	-1.35	-2.23
	Ga-B	15.74	24.54	-1.51	-2.44
	Ga-L	17.33	**25.53**	-1.47	-2.24
	Fl-S	18.17	21.98	-1.41	-2.13
	Fl-B	18.57	21.82	**-1.23**	-2.05
	Fl-B$^{(*)}$	18.55	21.45		
	Fl-L$^{(*)}$	20.85	23.72		

[Lee et al. 20] https://arxiv.org/pdf/2002.07233.pdf

Spasibo