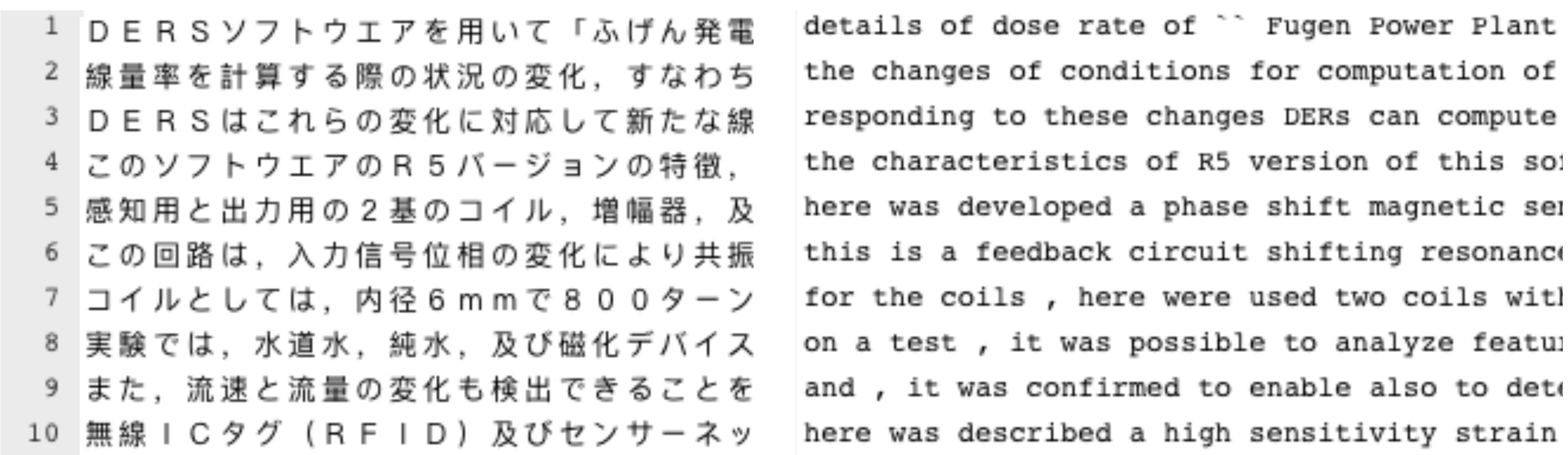
Non-autoregressive Neural Machine Translation Based on Latent-Variable Models
潜在変数モデルに基づいた非自己回帰型ニューラル機械翻訳

Raphael Shu (朱 中元)

東京大学

Neural Machine Translation

- NMT is conditional probabilisitic model


ＤＥＲＳソフトウエアを用いて「ふげん発電
線量率を計算する際の状況の変化，すなわち
ＤＥＲＳはこれらの変化に対応して新たな線
このソフトウエアのＲ５バージョンの特徴，
感知用と出力用の２基のコイル，増幅器，及
この回路は，入力信号位相の変化により共振
コイルとしては，内径６ｍｍで８００ターン
実験では，水道水，純水，及び磁化デバイス
また，流速と流量の変化も検出できることを
無線ＩＣタグ（ＲＦＩＤ）及びセンサーネッ

details of dose rate of `` Fugen Power Plant
the changes of conditions for computation of
responding to these changes DERs can compute
the characteristics of R5 version of this so
here was developed a phase shift magnetic se
this is a feedback circuit shifting resonanc
for the coils , here were used two coils wit
on a test , it was possible to analyze featu
and , it was confirmed to enable also to det
here was described a high sensitivity strain

Parallel corpus

training →

MT Model

$p(Y|X)$

- Translation in NMT
  - Find an output to maximize the probability

そこのどんつきまでガッと行ったら右やで

X

what is argmax p(Y|X=そこの...) ? →

Go straight and turn right

Y

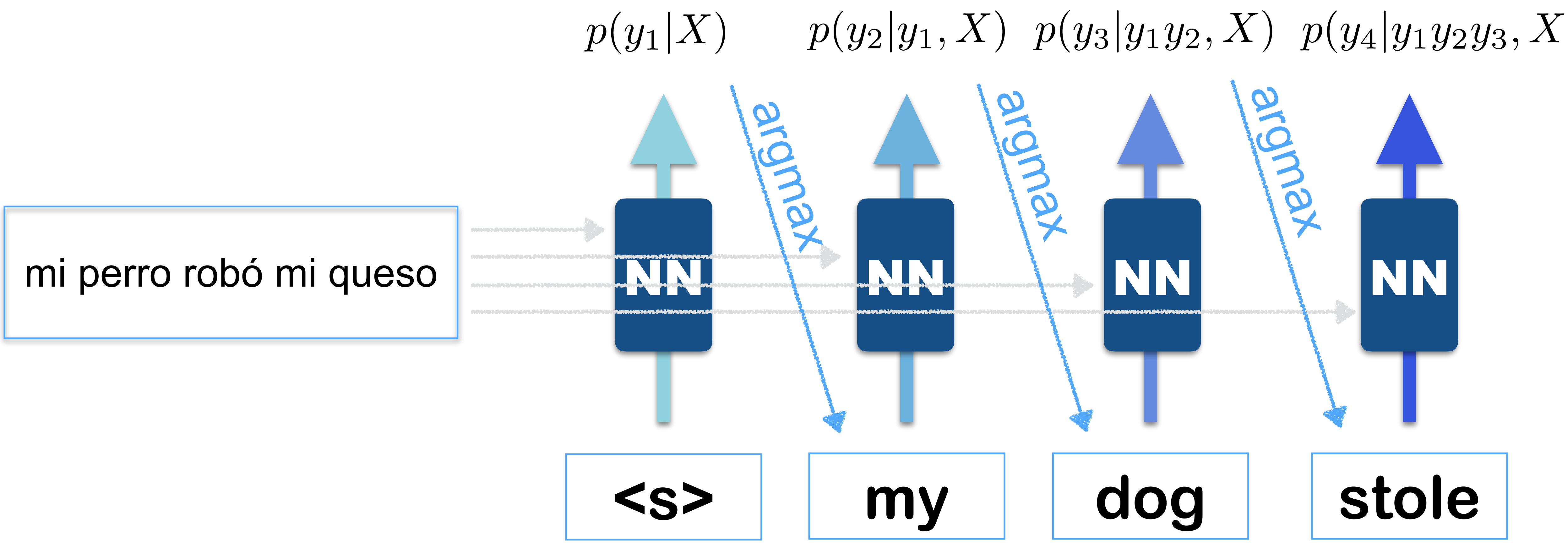
Autoregressive modeling

- Most of current application-level NMT models are based on autoregressive modeling
  - Breaks down p(Y|X) to word probabilities using chain rule
  - In each step, the model predicts the next word

$$
\begin{aligned}
&\texttt{p(my dog stole my cheese | mi perro robó mi queso)} \\
&\quad = \texttt{p(my | mi perro robó mi queso)} \\
&\qquad \texttt{p(dog | my, mi perro robó mi queso)} \\
&\qquad \texttt{p(stole | my dog, mi perro robó mi queso)} \\
&\qquad \texttt{p(my | my dog stole, mi perro robó mi queso)} \\
&\qquad \texttt{p(cheese | my dog stole my, mi perro robó mi queso)}
\end{aligned}
$$

Obtain Translations

- Approximating the global argmax with search algorithms
  - Greedy search and beam search

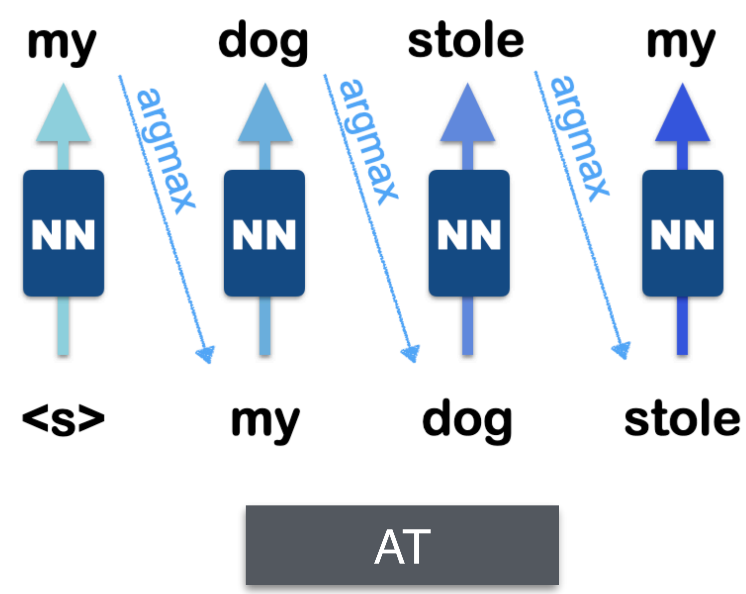


Problems

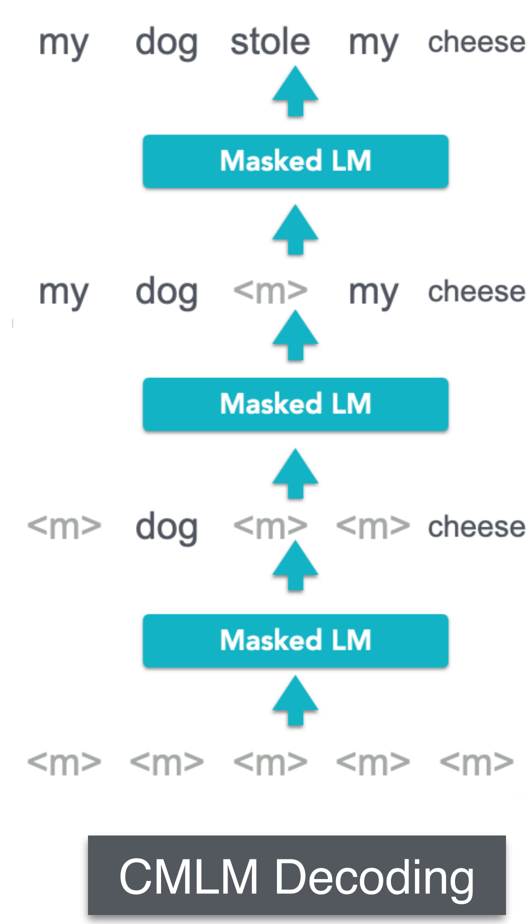
- <u>Low parallelizability</u>
- Worse if the model is bigger and deeper
- Require search algorithm to approximate the argmax

Non-autoregressive Machine Translation

- Predict all output tokens in one forward pass
- Can be fully parallelized on GPU

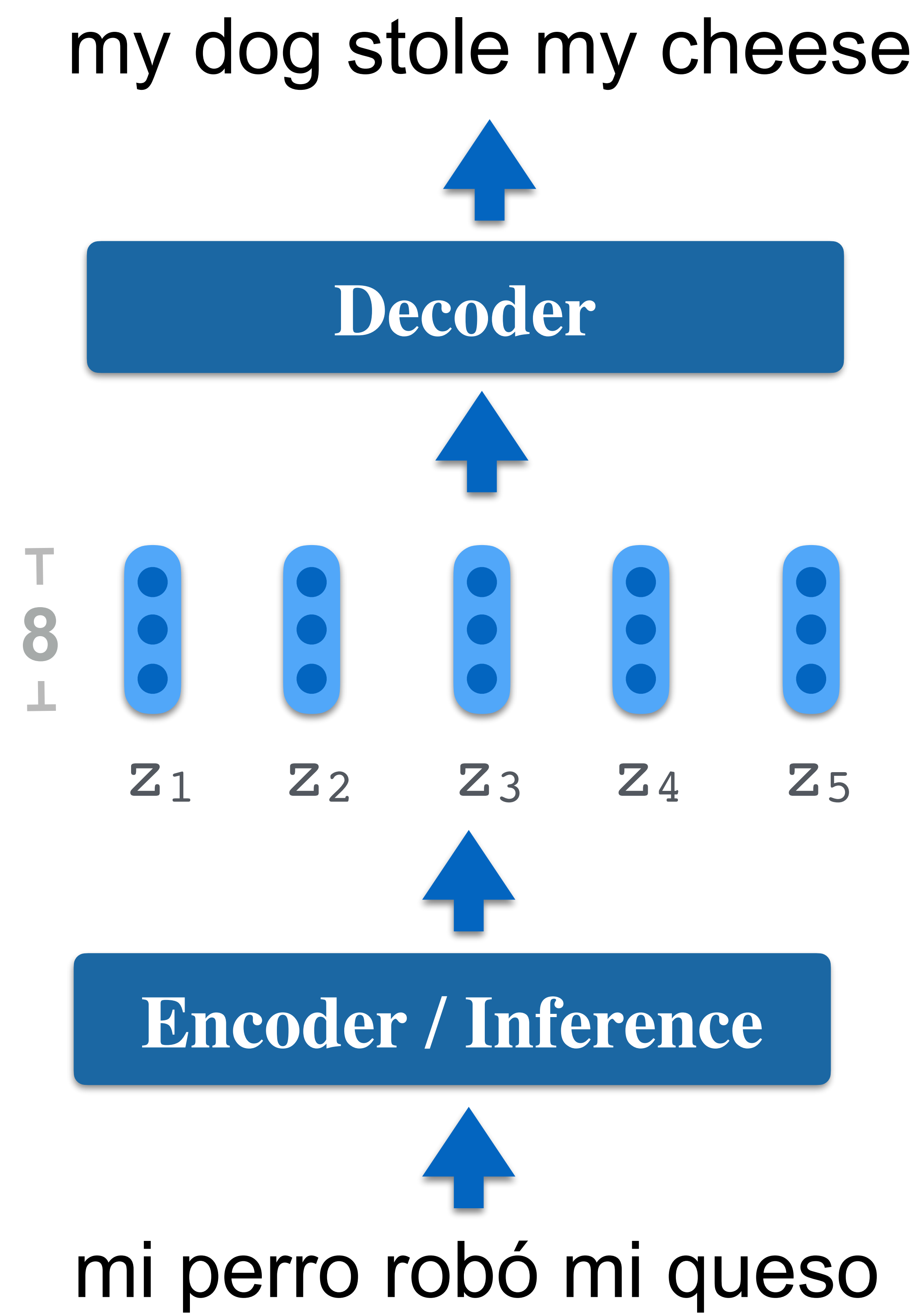


- CMLM (conditional masked language model) [Ghazvinine et al., 2018]
  - Predict the sequence and mask tokens with low confidence
  - Perform such token refinement by multiple iterations
- Drawback of token-based refinement models
  - Token prediction is time-consuming

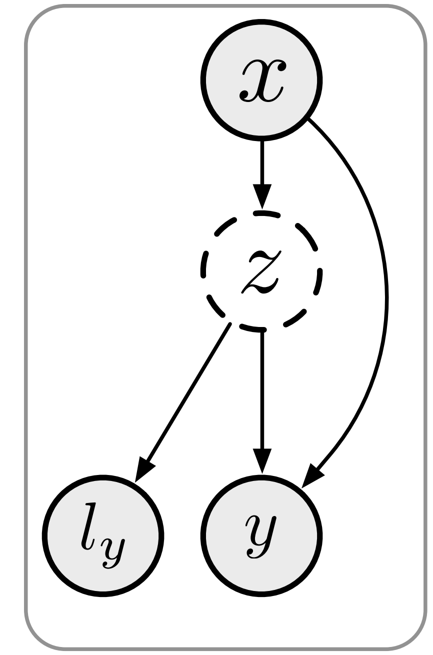


Latent-variable Non-autoregressive MT

- Core Idea:
  - Capture translation decisions with continuous latent variables
- Each source token is assigned with one latent variable
- Each latent variable is a low-dimensional vector
- Finding the best setting of latent variables with high-speed inference



Objective function



- Similar to VAE, we train our model with ELBO (evidence lower bound)

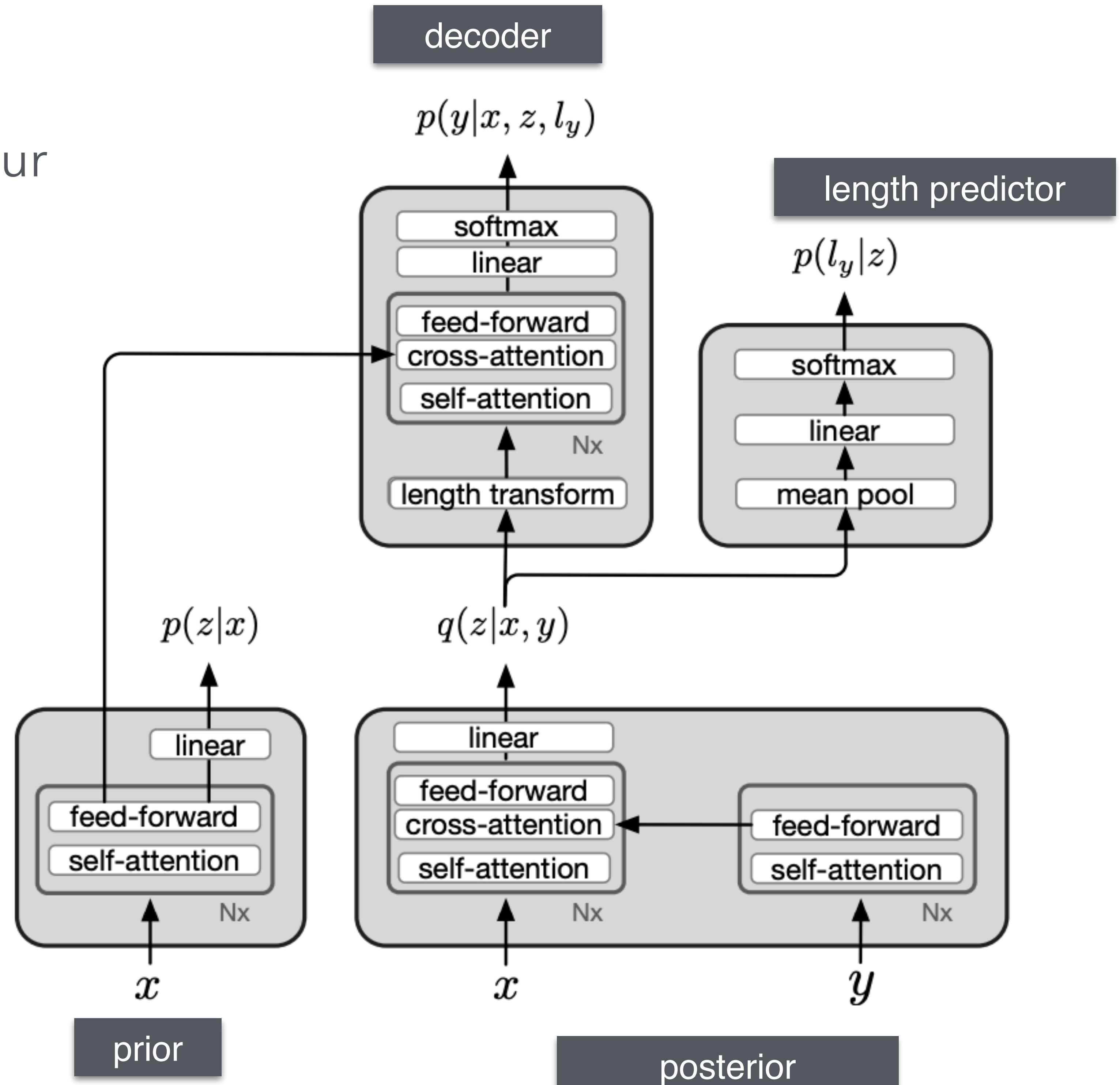
$$
\begin{aligned}
\log p(Y|X) &\geq \mathrm{ELBO}(\mathrm{X}, \mathrm{Y}; \theta, \phi, \omega) \\
&= \mathbb{E}_{Z\sim q_\phi}\Big[\underbrace{\log p_\theta(Y|X, Z, l_Y)}_{\text{decoder}} \underbrace{p_\theta(l_Y|X, Z)}_{\text{length predictor}} \Big] - \mathrm{KL}(\underbrace{q_\phi(Z|X, Y)}_{\text{posterior}} || \underbrace{p_\omega(Z|X)}_{\text{prior}})
\end{aligned}
$$

- Training:

$$
\hat{\theta}, \hat{\phi}, \hat{\omega} = \underset{\theta,\phi,\omega}{\operatorname{argmax}} \, \mathrm{ELBO}(X, Y; \theta, \phi, \omega)
$$

Model architecture

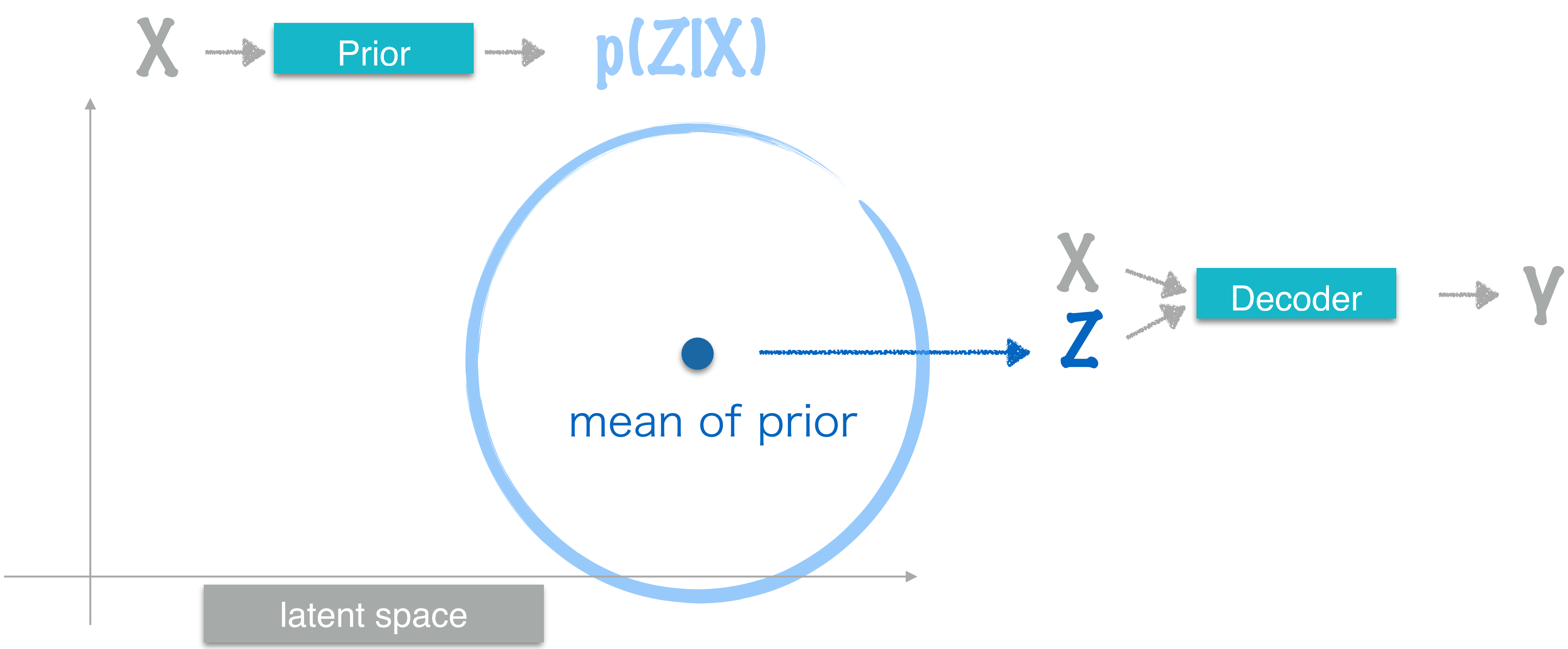
- Latent NAT parameterizes four distributions
- Reuse Transformer modules
- **Length Transform**: adjust |x| vectors to |y| vectors (skip the details here)



Translation with Latent NAT

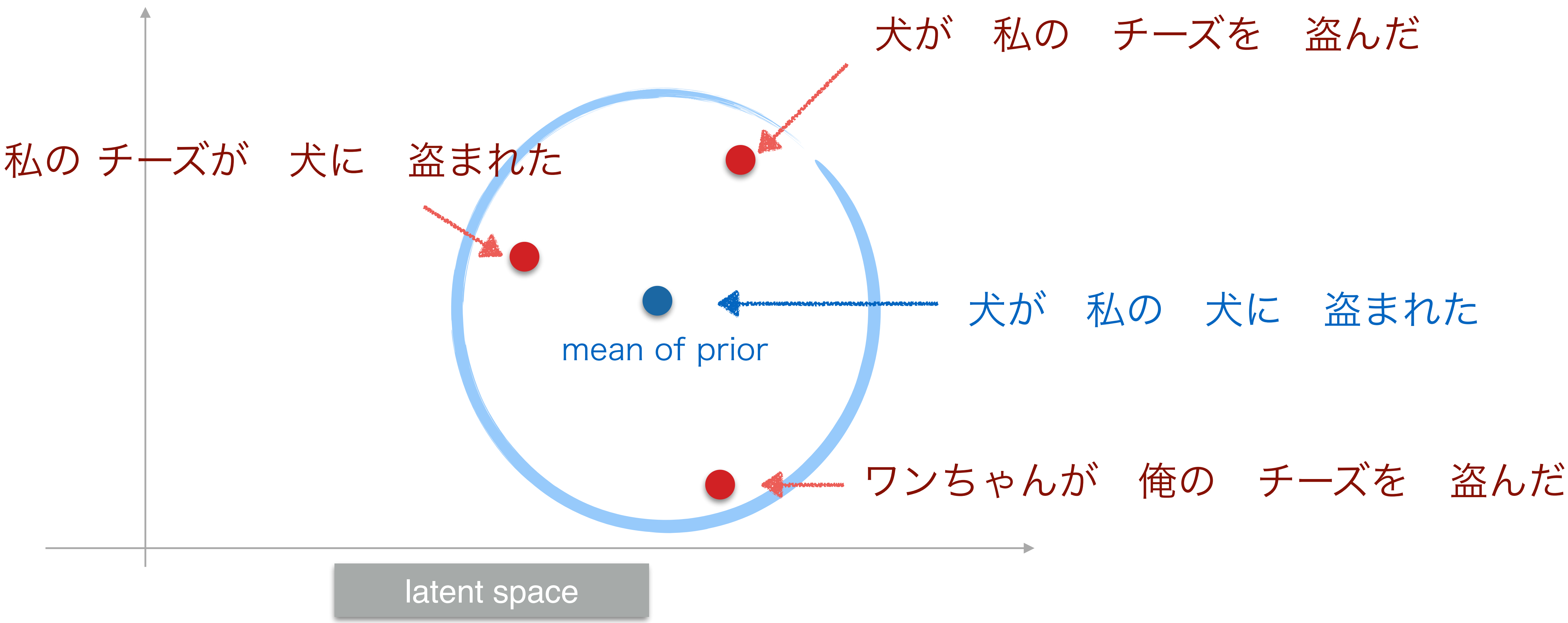
- After model training we get:

- Naive inference (decoding)



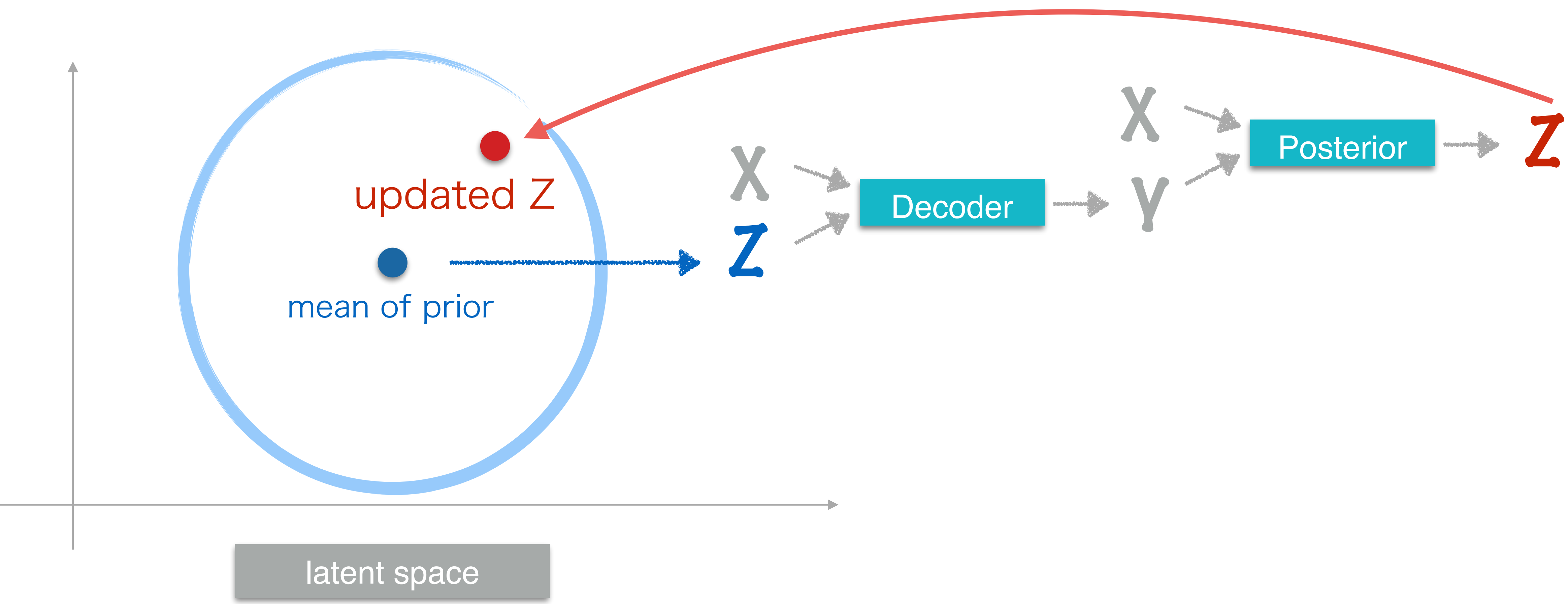
Problem of naive inference

- Problem of naive inference
  - the center of a Gaussian may not produce the best results



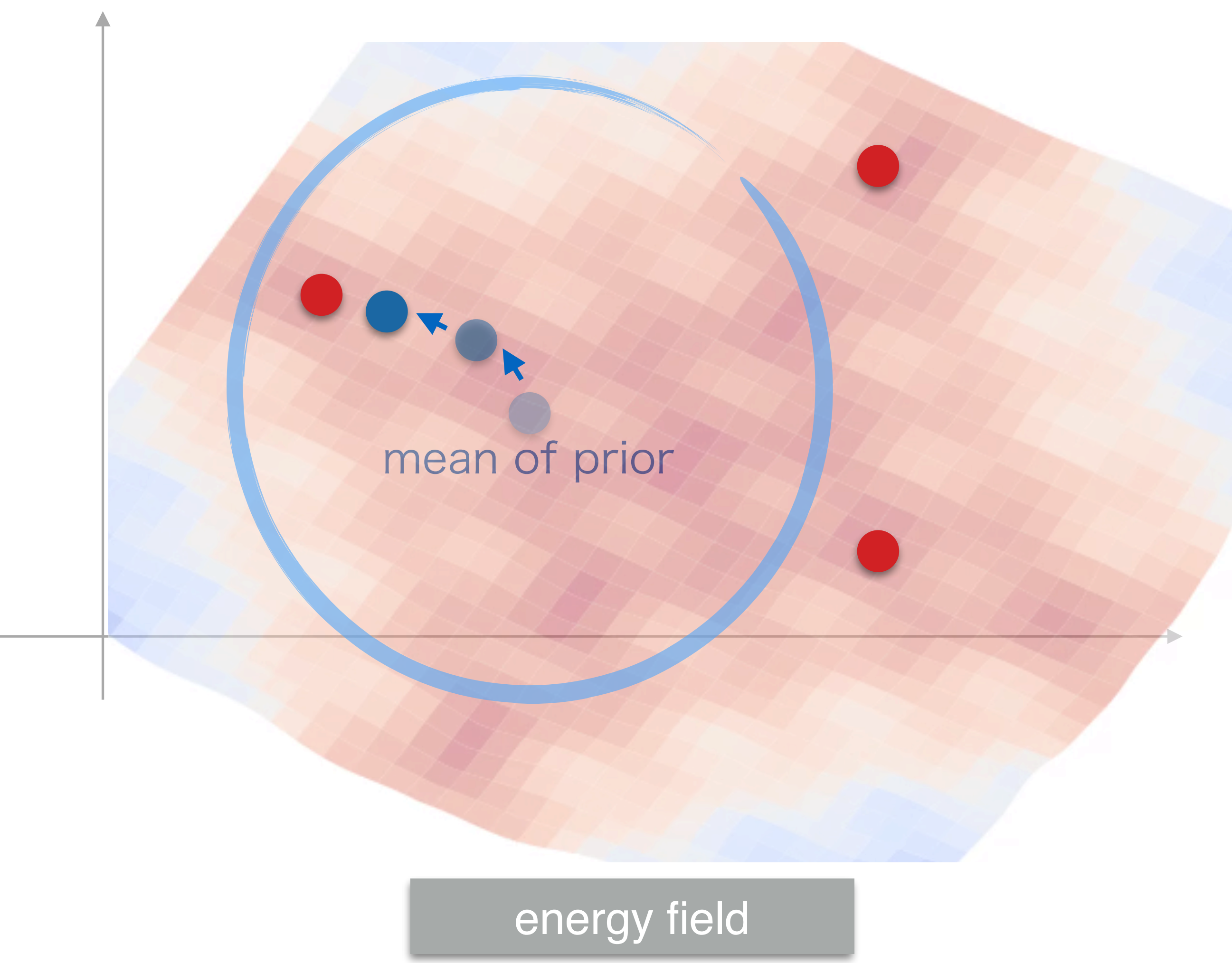
Better inference approaches (1)

- Delta inference (Shu et al., AAAI 2020)
  - Latent-variable updating for maximizing approximated ELBO
  - ELBO is improved after iterations with rapid convergence



Better inference approaches (2)

- Energy-based inference (Jason et al., EMNLP 2020)
  - Build an energy model $E(Z)$
  - High-quality latent vectors get low energy



- Update latent variables with the energy gradient

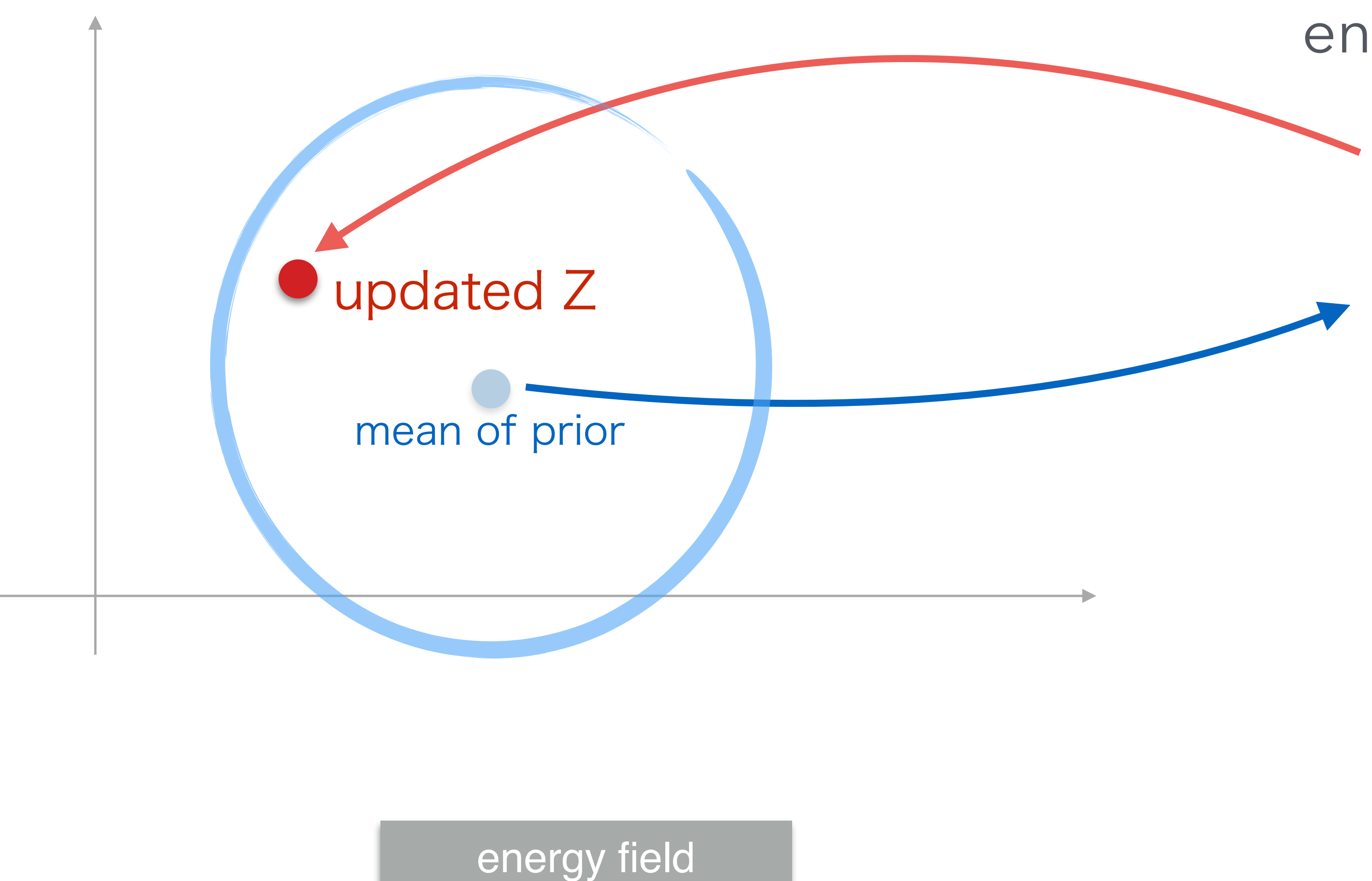
$$Z_{t+1} = Z_t - \alpha \nabla_{Z_t} E(Z_t)$$

Energy-driven gradient descent

Better inference approaches (2)

- Energy-based inference (Jason et al., EMNLP 2020)
  - Build an energy model $E(Z)$
  - High-quality latent vectors get low energy

- Update latent variables with the energy gradient



$$Z_{t+1} = Z_t - \alpha \nabla_{Z_t} E(Z_t)$$

Energy-driven gradient descent

Experiments

Experiment settings

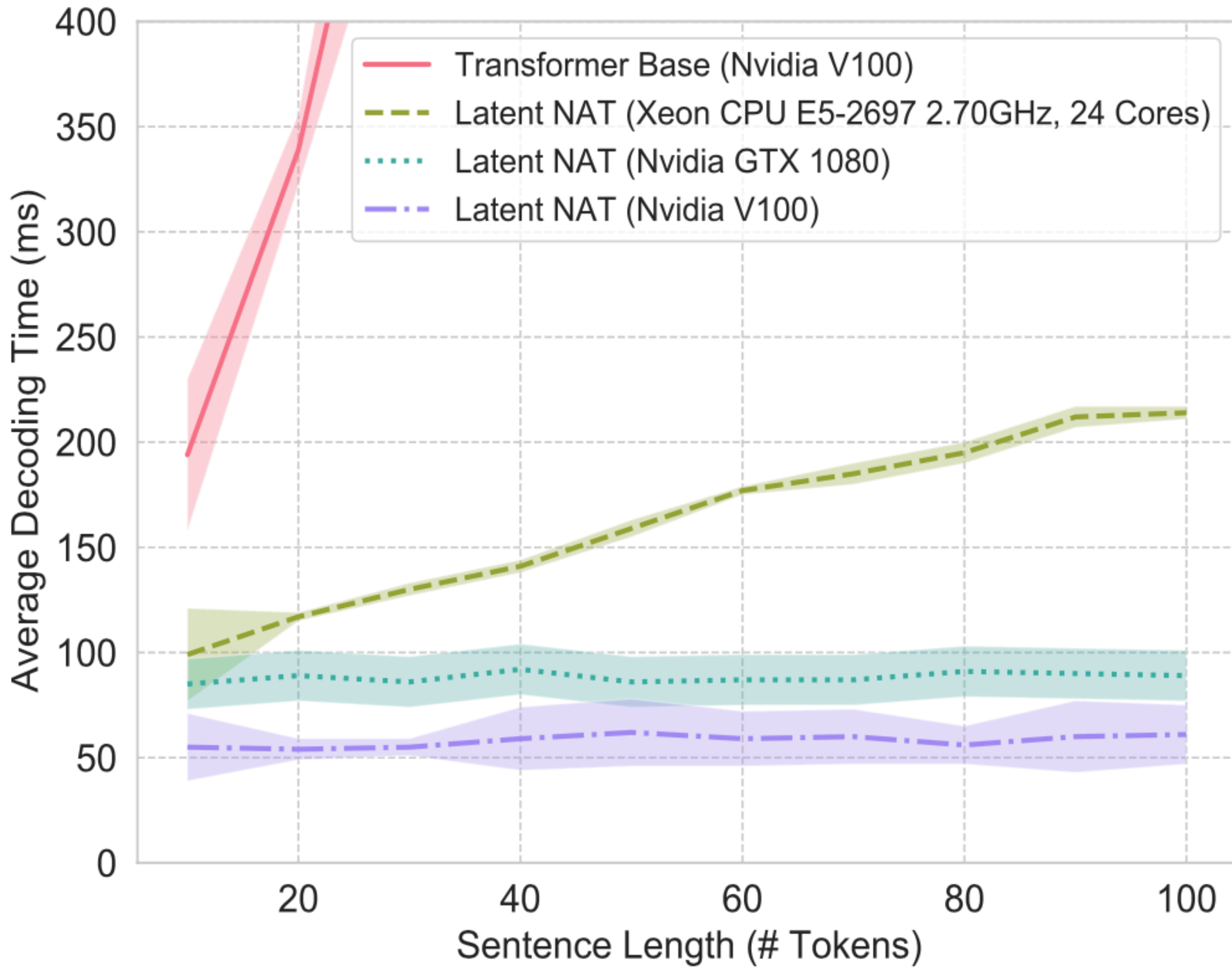
- Dataset:
  - WMT'14 English -> German
  - IWSLT'16 Romanian -> English
  - IWSLT'16 German -> English
- Evaluation
  - Translation quality: BLEU
  - Translation speed: averaged decoding time for one sentence

Experiment results on machine translation

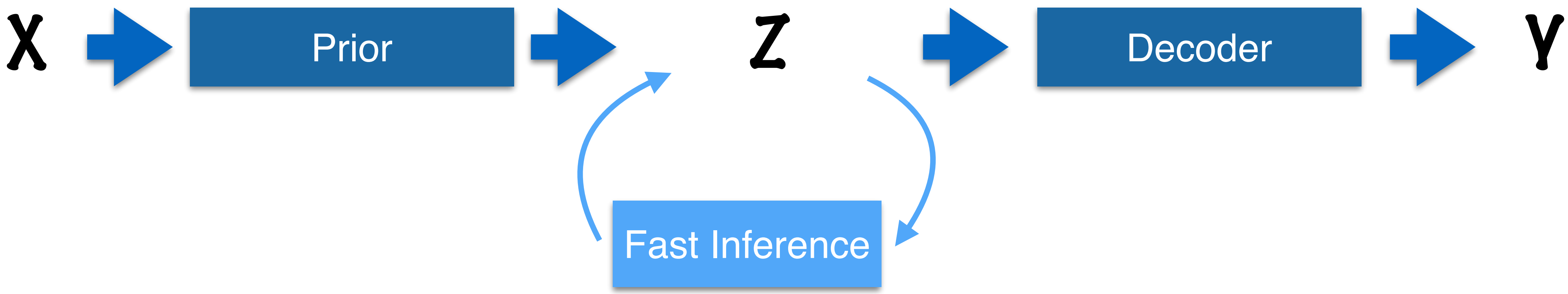
	WMT'14 En-De		WMT'16 Ro -> En		IWSLT'16 De -> En	
	BLEU	Speed	BLEU	Speed	BLEU	Speed
Transformer baseline, beam = 3	28.3	1x	31.5	1x	31.5	1x
Transformer baseline, beam = 1	27.5	1.1x	30.9	1.1x	31.1	1.1x
Latent NAT (Naive Inference)	25.7	15x	28.4	34x	27.0	19x
+ Delta Inference	26.1	6.3x	29.0	19x	28.3	11x
+ Energy Inference (w/ approximation)	26.3	10x	29.1	24x	28.8	13x
+ Score Inference + Latent Search	27.4	6.2x	30.4	15x	30.2	6.3x

- Latent Search: parallel decoding by sampling multiple latent variable

Translation speed related to computational capacity



Conclusion



- We show a novel sequence generation framework
  - sequence prediction problem is solved by latent-variable inference in the continuous space
- Continuous setting and low dimensionality enable us to updating efficiently
- Fit for on-device computing

Thanks