ICML | 2019

Thirty-sixth International Conference
on Machine Learning

CoT: Cooperative Training for Generative Modeling of Discrete Data

https://github.com/desire2020/CoT

Sidi Lu, Lantao Yu, Siyuan Feng, Yaoming Zhu, Weinan Zhang, and Yong Yu

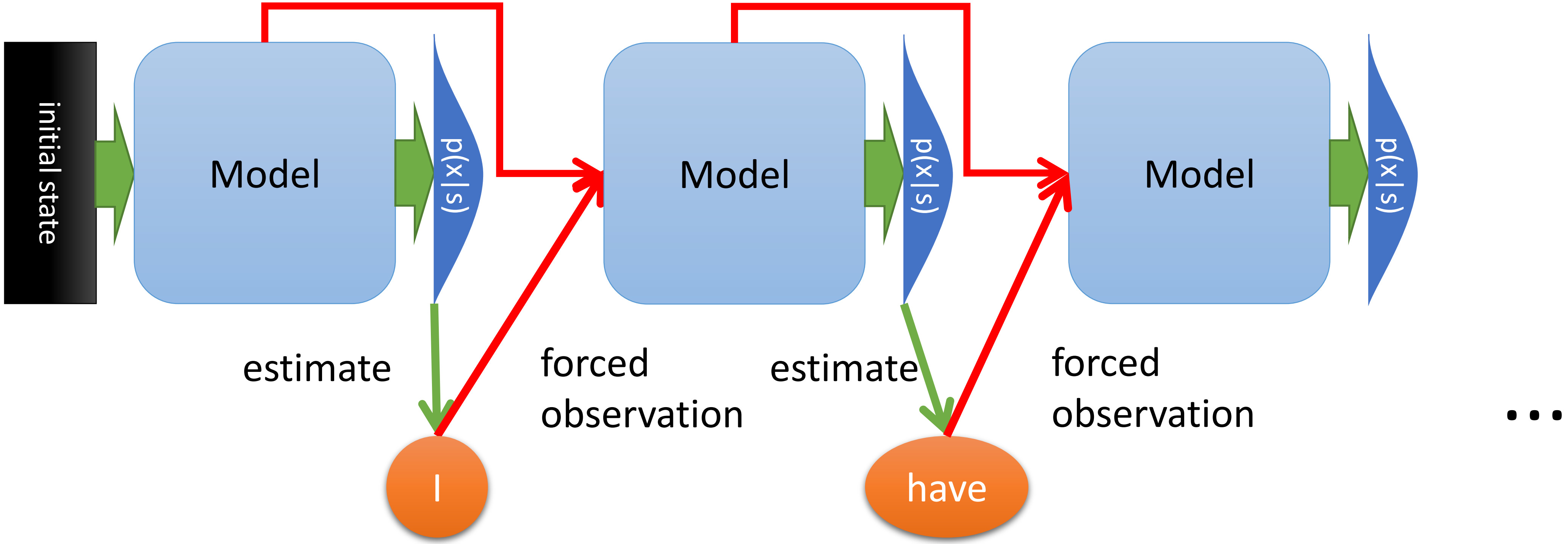
Shanghai Jiao Tong University

Autoregressive Models

- Autoregressive models factorize the distribution sequentially to build a fully tractable density function:
  - $p_\theta(x_0, x_1, \dots, x_{n-1}) =$ $p_\theta(x_0)\, p_\theta(x_1 \mid s_{[0:1)}) p_\theta(x_2 \mid s_{[0:2)}) p_\theta(x_3 \mid s_{[0:3)}) \dots p_\theta(x_{n-1} \mid s_{[0:n-1)})$

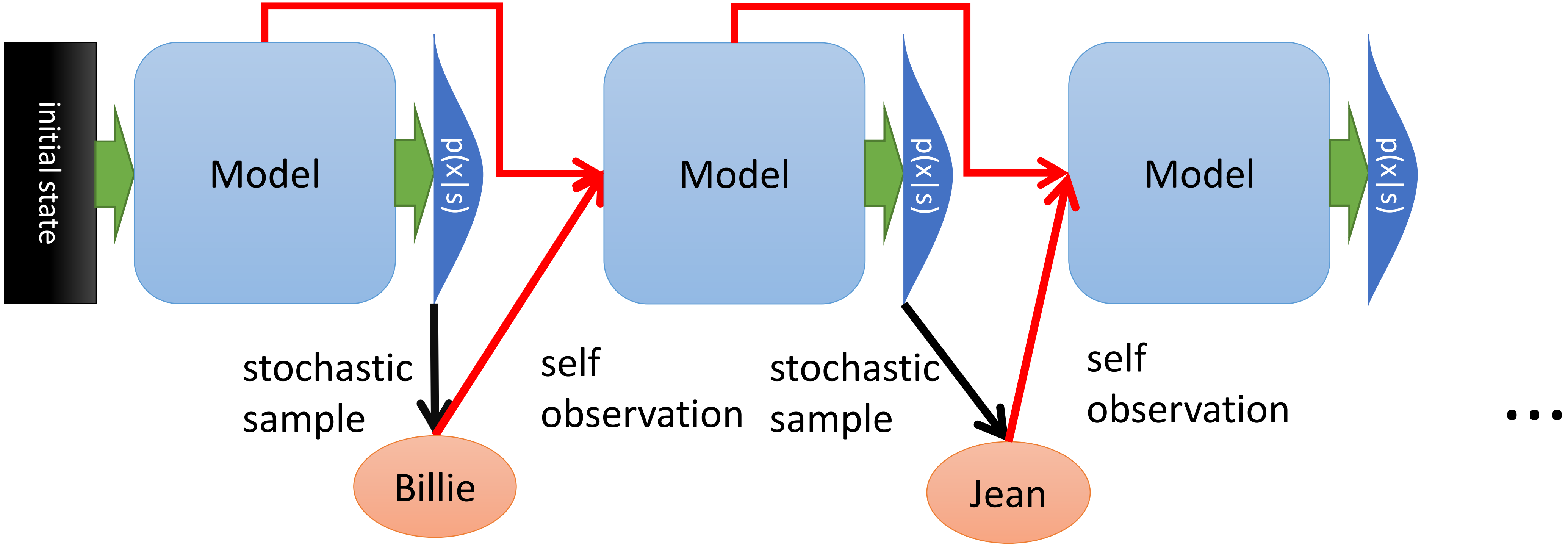
Teacher Forcing and Exposure Bias

- For each sequence in the training set, maximize the estimated likelihood in the log scale:



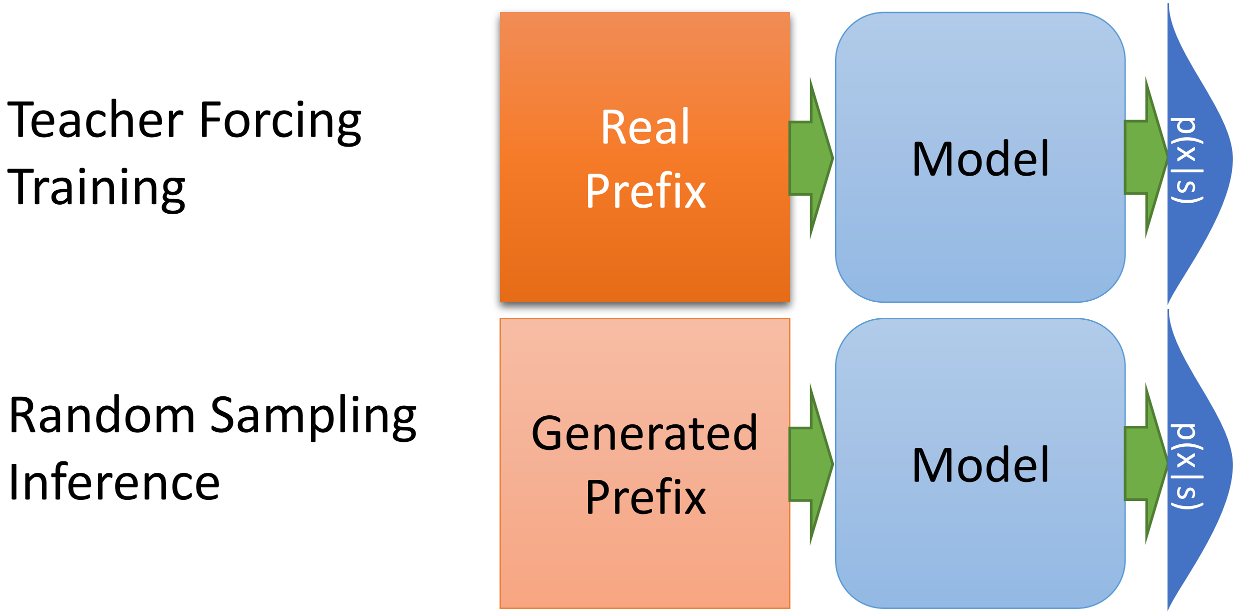
Teacher Forcing and Exposure Bias

- When used to generate random sample:



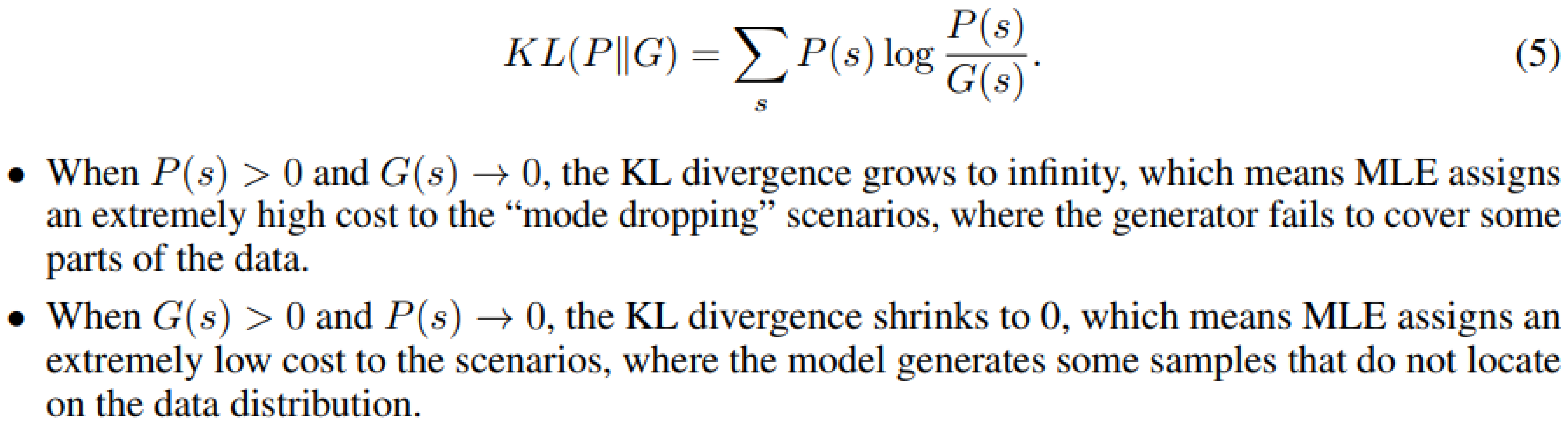
Teacher Forcing and Exposure Bias

- Exposure Bias [Ranzato et al., 2015]:
  - The intermediate process under training stage and inference stage is inconsistent.
  - The distribution shift would accumulate along the timeline.



Exposure Bias and Kullback-Leibler Divergence

- Exposure Bias could also be regarded as a result of optimization via minimizing Kullback-Leibler Divergence, denoted as KL(P||Q) for measured distributions P, Q.



$$KL(P\|G) = \sum_{s} P(s) \log \frac{P(s)}{G(s)}. \tag{5}$$

- When $P(s) > 0$ and $G(s) \to 0$, the KL divergence grows to infinity, which means MLE assigns an extremely high cost to the "mode dropping" scenarios, where the generator fails to cover some parts of the data.
- When $G(s) > 0$ and $P(s) \to 0$, the KL divergence shrinks to 0, which means MLE assigns an extremely low cost to the scenarios, where the model generates some samples that do not locate on the data distribution.

Kullback-Leibler Divergence, Symmetry of Divergences

- For any P, Q, KL(P||Q) not necessarily equals to KL(Q||P)
- KL ---smoothed and symmetrized--> Jensen-Shannon Divergence

$$JSD(P\|G) = \frac{1}{2}\big(KL(P\|M) + KL(G\|M)\big)$$

  - where M = 0.5 * (P + G)

GAN, SeqGAN and Language GANs

- Ian Goodfellow proposed Generative Adversarial Network [2014]
  - Ideally, GAN minimizes the JSD
- Can’t be directly applied to discrete sequence generation
- SeqGAN uses the REINFORCE gradient estimator to resolve this.

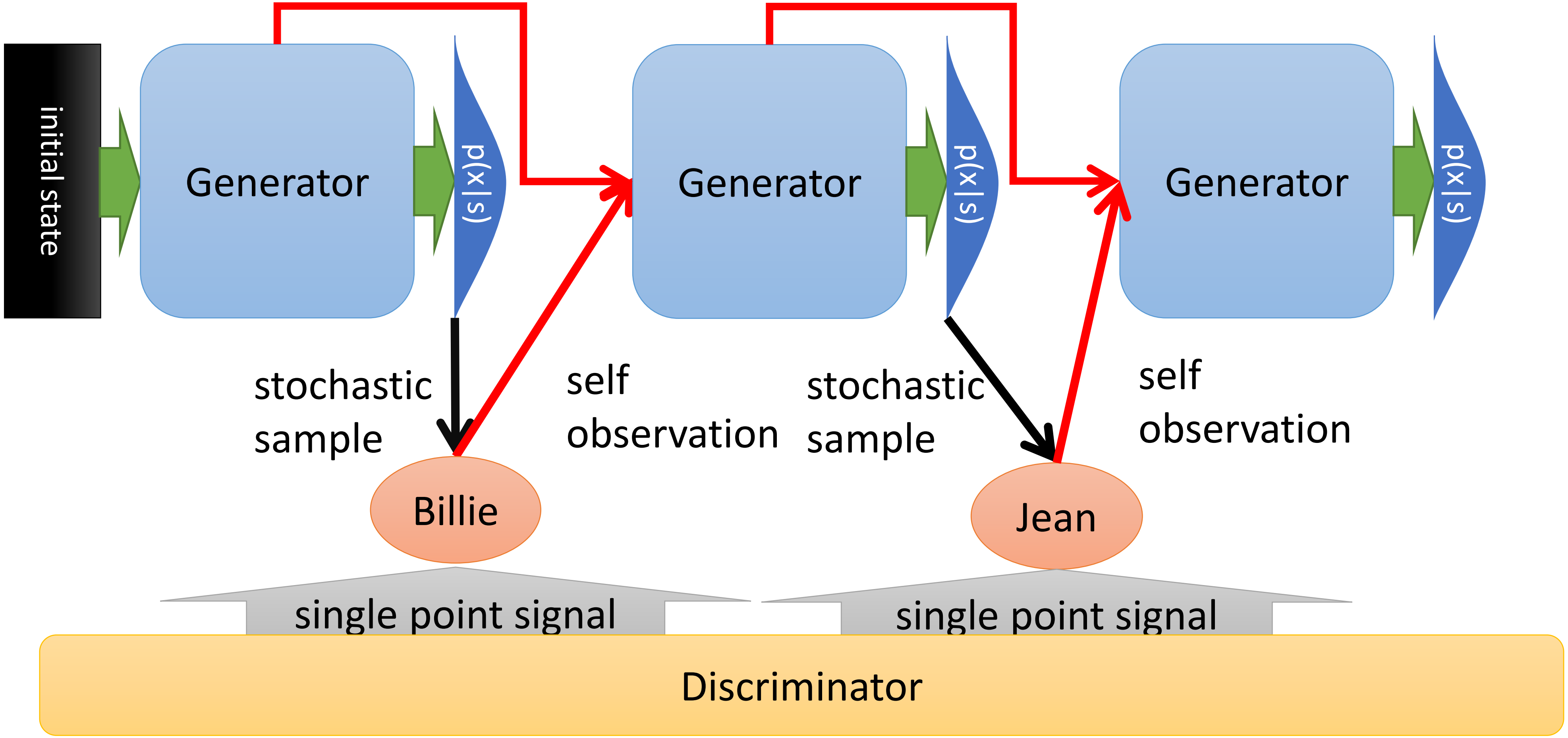
$$\min_{\theta} \max_{\phi} \mathbb{E}_{s \sim p_{\text{data}}} \left[\log(D_{\phi}(s))\right] + \mathbb{E}_{s \sim G_{\theta}} \left[\log(1 - D_{\phi}(s))\right]$$

Problems of SeqGAN

- Not trivially able to work from scratch.
  - SeqGAN’s work-around: Pre-training via teacher forcing.
- Trade diversity for quality (mode collapse)
  - According to previous reports([Lu et al. 2018; Caccia et al. 2018])

Problems of SeqGAN

- Training signal is too sparse.



Cooperative Training: Back to Formula!

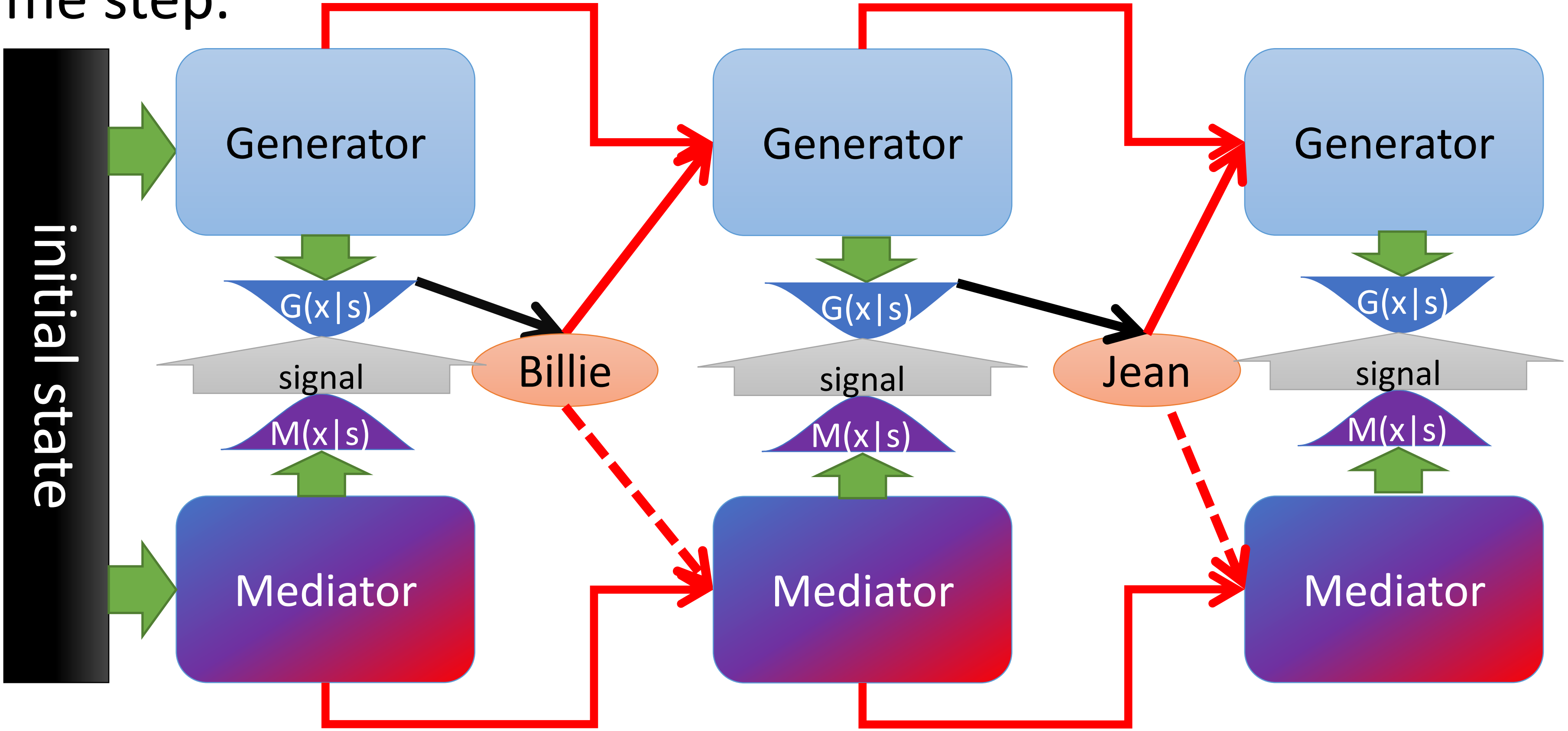
- Reconsider the algorithm from estimating & minimizing JSD:

$$JSD(P\|G) = \frac{1}{2}\big(KL(P\|M) + KL(G\|M)\big)$$

  - where M = 0.5 * (P + G)
- Instead of using a discriminator to achieve this, use another sequence model called “Mediator” to approximate the mixture density M.

Cooperative Training: More Information from Mediator

- Key Idea: The mediator provides DISTRIBUTION level signal in each time step.



Cooperative Training: Factorizing the Cumulative Gradient Through Time, Final Objectives

- Generator Gradient:

$$\nabla_\theta J_g^{0.0}(\theta) = \sum_{t=0}^{n-1} \mathop{\mathbb{E}}_{s_t \sim G_\theta} \left[\nabla_\theta \pi_g(s_t)^\top \left(\log \frac{\pi_m(s_t)}{\pi_g(s_t)} \right) \right]. \tag{14}$$

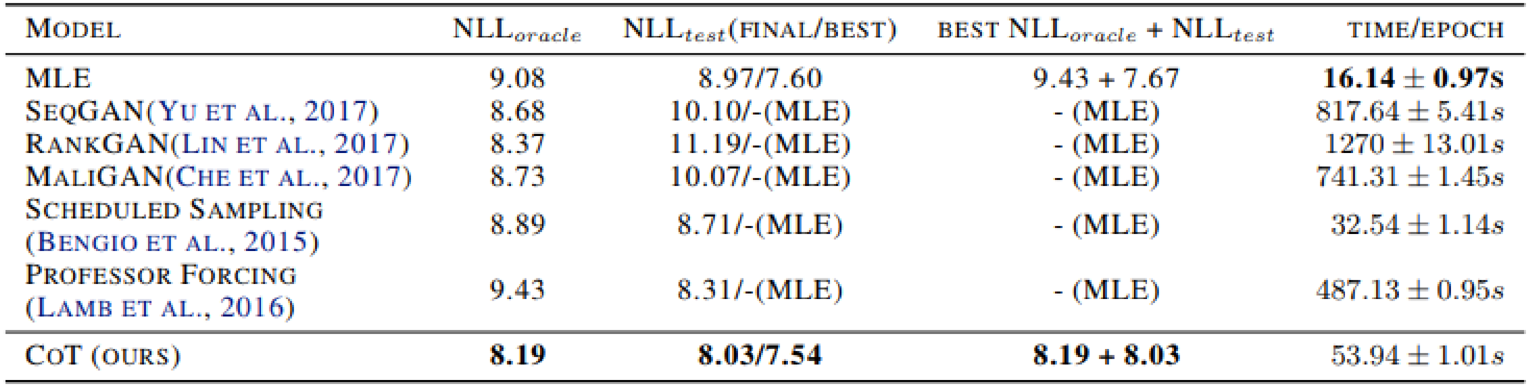
  - where $\pi_g(s_t) = G_\theta(x|s_t), \pi_m(s_t) = M_\phi(x|s_t),$

- Mediator Objective:

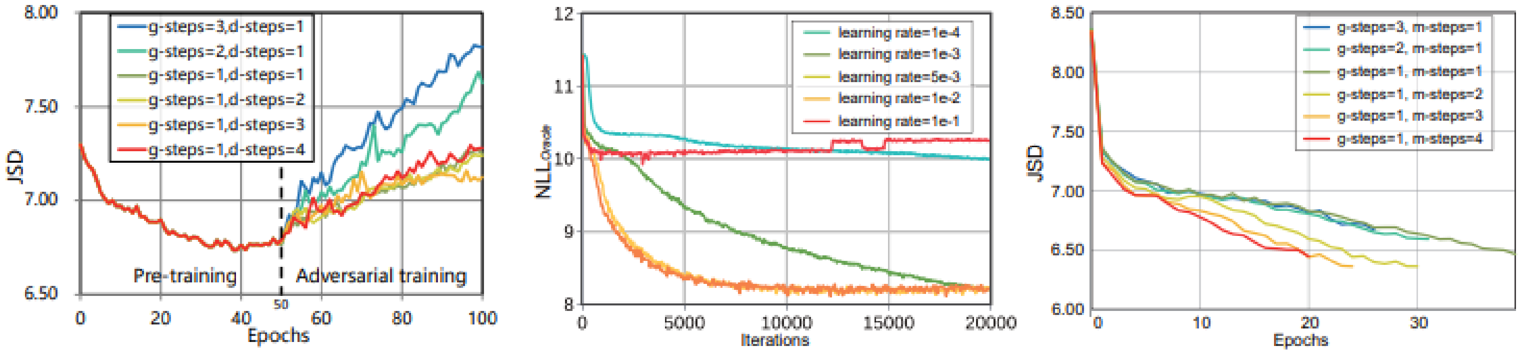
$$J_m(\phi) = \frac{1}{2} \left(\mathop{\mathbb{E}}_{s \sim G_\theta} [-\log(M_\phi(s))] + \mathop{\mathbb{E}}_{s \sim p_{\text{data}}} [-\log(M_\phi(s))] \right). \tag{9}$$

Experiment: Synthetic Turing Test

Table 1. Likelihood-based benchmark and time statistics for synthetic Turing test. '-(MLE)' means the best performance is acquired during MLE pre-training.



MODEL	NLL_{oracle}	NLL_{test}(FINAL/BEST)	BEST NLL_{oracle} + NLL_{test}	TIME/EPOCH
MLE	9.08	8.97/7.60	9.43 + 7.67	**16.14 ± 0.97s**
SEQGAN(YU ET AL., 2017)	8.68	10.10/-(MLE)	- (MLE)	817.64 ± 5.41s
RANKGAN(LIN ET AL., 2017)	8.37	11.19/-(MLE)	- (MLE)	1270 ± 13.01s
MALIGAN(CHE ET AL., 2017)	8.73	10.07/-(MLE)	- (MLE)	741.31 ± 1.45s
SCHEDULED SAMPLING (BENGIO ET AL., 2015)	8.89	8.71/-(MLE)	- (MLE)	32.54 ± 1.14s
PROFESSOR FORCING (LAMB ET AL., 2016)	9.43	8.31/-(MLE)	- (MLE)	487.13 ± 0.95s
COT (OURS)	**8.19**	**8.03/7.54**	**8.19 + 8.03**	53.94 ± 1.01s



(a) JSD of SeqGAN (b) NLL_{oracle} of CoT (c) JSD of CoT

Figure 2. Curves of evaluation on JSD, NLL_{oracle} during iterations of CoT under different training settings. To show the hyperparameter robustness of CoT, we compared it with a typical language GAN *i.e.* SeqGAN (Yu et al., 2017).

Experiment: Real World Data

Quality Test on EMNLP2017 WMT News Section

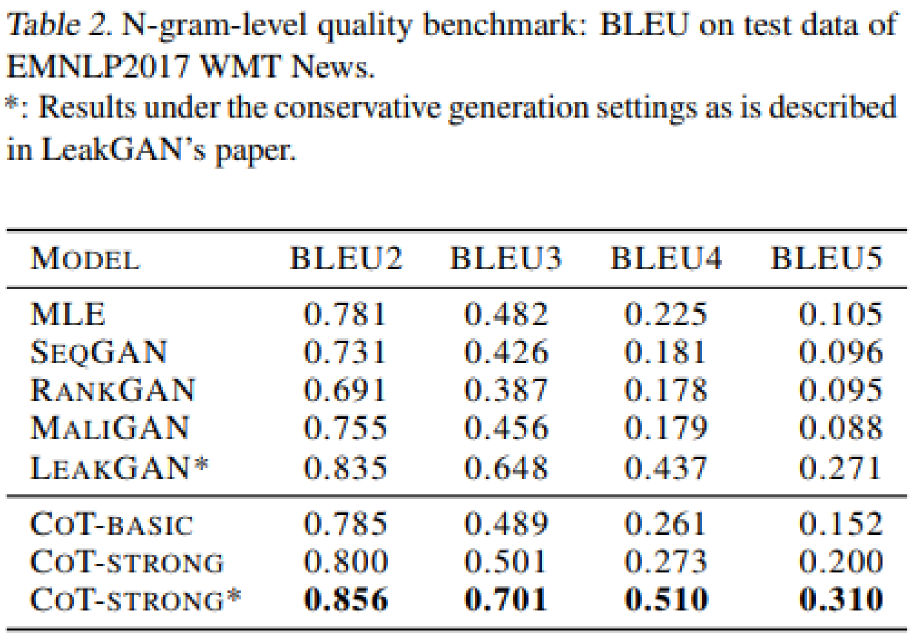


Table 2. N-gram-level quality benchmark: BLEU on test data of EMNLP2017 WMT News.
*: Results under the conservative generation settings as is described in LeakGAN's paper.

MODEL	BLEU2	BLEU3	BLEU4	BLEU5
MLE	0.781	0.482	0.225	0.105
SEQGAN	0.731	0.426	0.181	0.096
RANKGAN	0.691	0.387	0.178	0.095
MALIGAN	0.755	0.456	0.179	0.088
LEAKGAN*	0.835	0.648	0.437	0.271
COT-BASIC	0.785	0.489	0.261	0.152
COT-STRONG	0.800	0.501	0.273	0.200
COT-STRONG*	**0.856**	**0.701**	**0.510**	**0.310**

Reasonable Diversity Test on EMNLP2017 WMT News Section

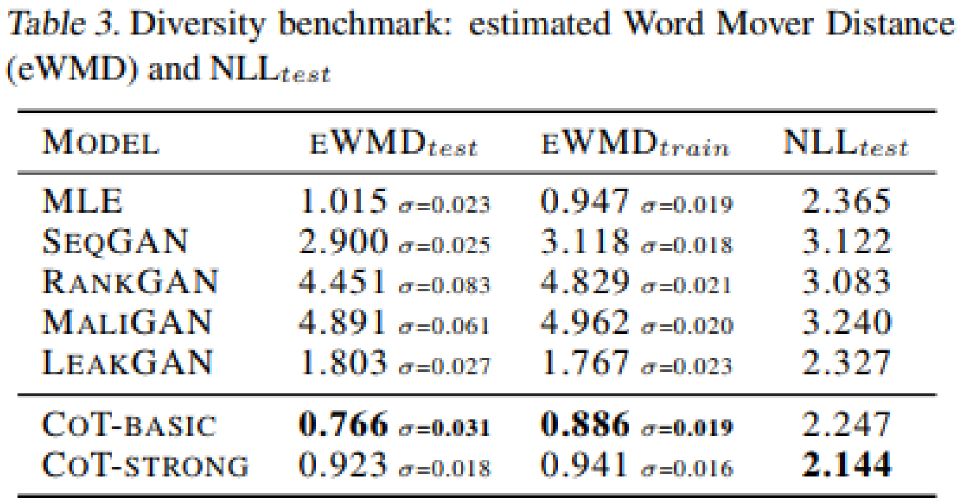


Table 3. Diversity benchmark: estimated Word Mover Distance (eWMD) and NLL$_{test}$

MODEL	EWMD$_{test}$	EWMD$_{train}$	NLL$_{test}$
MLE	1.015 $_{\sigma=0.023}$	0.947 $_{\sigma=0.019}$	2.365
SEQGAN	2.900 $_{\sigma=0.025}$	3.118 $_{\sigma=0.018}$	3.122
RANKGAN	4.451 $_{\sigma=0.083}$	4.829 $_{\sigma=0.021}$	3.083
MALIGAN	4.891 $_{\sigma=0.061}$	4.962 $_{\sigma=0.020}$	3.240
LEAKGAN	1.803 $_{\sigma=0.027}$	1.767 $_{\sigma=0.023}$	2.327
COT-BASIC	**0.766** $_{\sigma=0.031}$	**0.886** $_{\sigma=0.019}$	2.247
COT-STRONG	0.923 $_{\sigma=0.018}$	0.941 $_{\sigma=0.016}$	**2.144**

Conclusion

Poster #44

- Key Ideas:
  - Use a max-max game to replace min-max game of GANs, while still focusing on minimization of JSD.
  - Use distribution-level signal from the introduced mediator in each step.
- Advantage:
  - Works from scratch.
  - Trade-off invariant performance gain while still being computationally cheap enough.