GAN for Text Generation ?

Yongcheng Wang

Nov 27th, 2018

Contents

1、Preview of NLP and NLG

2、Classic Text Generation Models

3、GAN for Text Generation?

- Difficulties
- Possible solutions

4、Papers Review

- Adversarial Generation of Natural Language
- MASKGAN: Better Text Generation via Filling in the ______
- Generating Text via Adversarial Training
- Adversarial Feature Matching for Text Generation

Preview

Overview of Artificial Intelligence and Role of Natural Language Processing in Big Data

<原文链接>　<中文译文>

1、What is Natural Language Processing?

- 自然语言处理（NLP）是指机器理解并解释人类写作、说话方式的能力。
- NLP 的目标是让计算机 / 机器在理解语言上像人类一样智能。
- 最终目标是弥补人类交流（自然语言）和计算机理解（机器语言）之间的差距。

2、Three different levels of linguistic analysis done before performing NLP

- Syntax（句法） - What part of given text is grammatically true.
- Semantics （语义） - What is the meaning of given text?
- Pragmatics （语用） - What is the purpose of the text?

3、NLP deal with different aspects of language such as:

- Phonology（音韵学） - It is systematic organization of sounds in language.
- Morphology （词态学） - It is a study of words formation and their relationship with each other.

4、Two essential processes in NLP:

- Natural language comprehension & inference
- Natural language generation

5、Natural language generation divided into three proposed stages

NLG 是从结构化数据中以可读地方式自动生成文本的过程。自然语言生成的问题是难以处理。自然语言生成可被分为三个阶段:

1. 文本规划: 完成结构化数据中基础内容的规划。
2. 语句规划: 从结构化数据中组合语句，来表达信息流。
3. 实现: 产生语法通顺的语句来表达文本。

Classic Text Generation Models

——Problems of Auto-encoder based models

Cited from <<Generating Text via Adversarial Training>> NIPS 2016

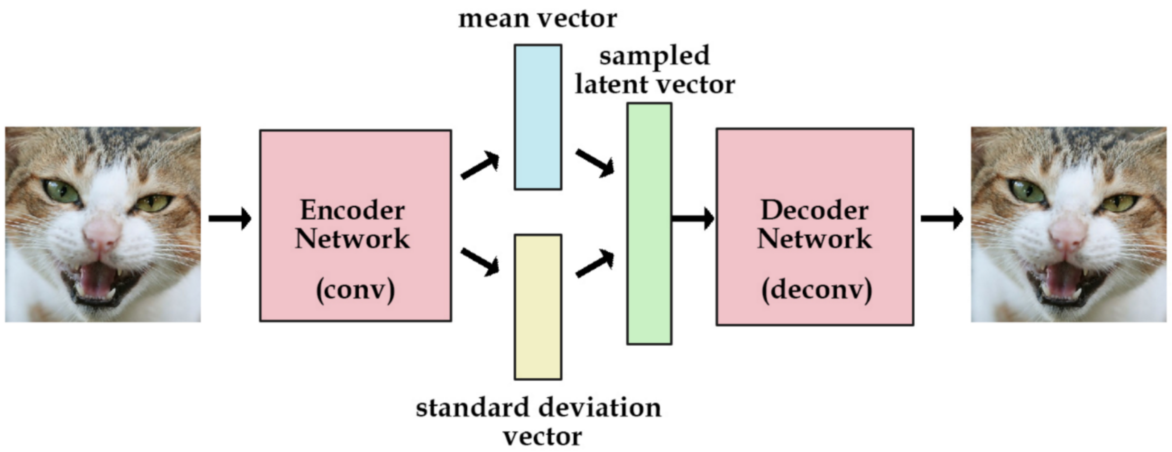
1 autoencoder-based methods may fail when generating realistic sentences from arbitrary latent representations. The reason behind this is that when mapping sentences to their hidden representations using an autoencoder, the representations of these sentences may often occupy a small region in the hidden space. Thereby, most of regions in the hidden space do not necessarily maps to a realistic sentence.

生成的hidden representation往往集中在整个隐层空间的一小部分，很大一部分隐层空间没有对应的词/句

2 Consequently, using a randomly generated hidden representation from a prior distribution would usually leads to implausible sentences.

当模型将那些没有对应词/句的hidden representation用于文本生成时，就会产生一些不靠谱的句子

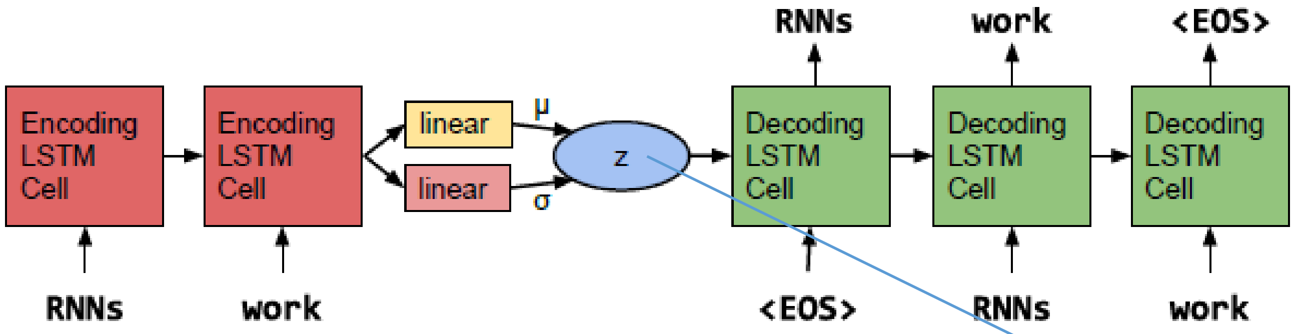
1、 Variational Autoencoder



left: 1st epoch, middle: 9th epoch, right: original

Auto-Encoding Variational Bayes 2014
{Diederik P. Kingma， Max Welling } @Universiteit van Amsterdam

VAE的一个劣势就是没有使用对抗网络，所以会更趋向于产生模糊的图片



Generating Sentences from a Continuous Space CoNLL 2016
Samuel R. Bowman(Stanford NLP Group), Luke Vilnis(University of Massachusetts Amherst),
{Oriol Vinyals, Andrew M. Dai(MaskGAN） , Rafal Jozefowicz & Samy Bengio} @ Google Brain

1、使用VAE方法训练到的隐层表示可以更好的在全局角度把握句子的生成，在句子的style, topic, high-level syntactic features等全局特性有优异的表现。

2、通过简单的decoding就可以得到更连贯并且更多样（diverse）的句子

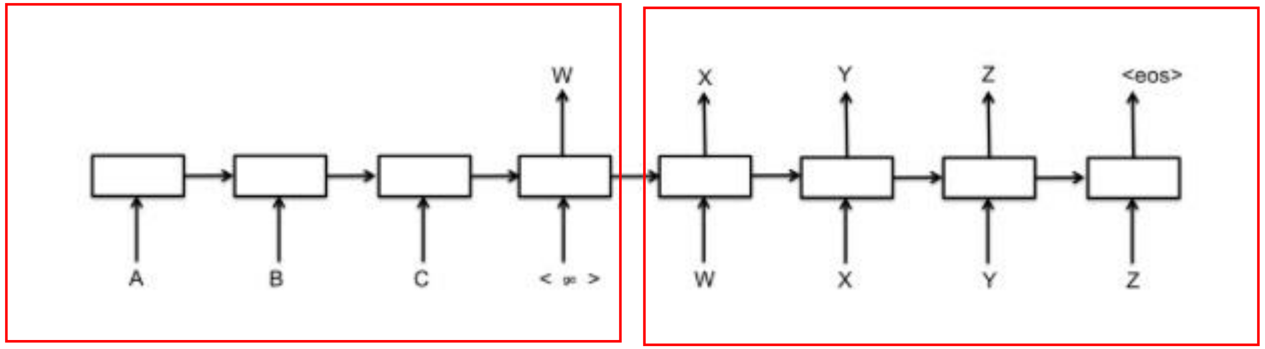
the hidden variables would not cover the hidden space

Classic Text Generation Models

2、Seq2Seq

Seq2Seq技术突破了传统的固定大小输入问题框架，并被证实在翻译人机短问快答的应用中有着不俗的表现。解决问题的主要思路是通过深度神经网络模型（常用的是LSTM，长短记忆网络，一种循环神经网络）将一个作为输入的序列映射为一个作为输出的序列，这一过程由编码输入与解码输出两个环节组成。如下图：

Evaluation model



encode

decode

$$\mathbf{h}_t = f(x_t, \mathbf{h}_{t-1}); \quad \mathbf{c} = \phi(\{\mathbf{h}_1, ..., \mathbf{h}_{T_S}\})$$

$$\mathbf{s}_t = f(y_{t-1}, \mathbf{s}_{t-1}, \mathbf{c})$$
$$p(y_t|y_{<t}, X) = g(y_{t-1}, \mathbf{s}_t, \mathbf{c})$$

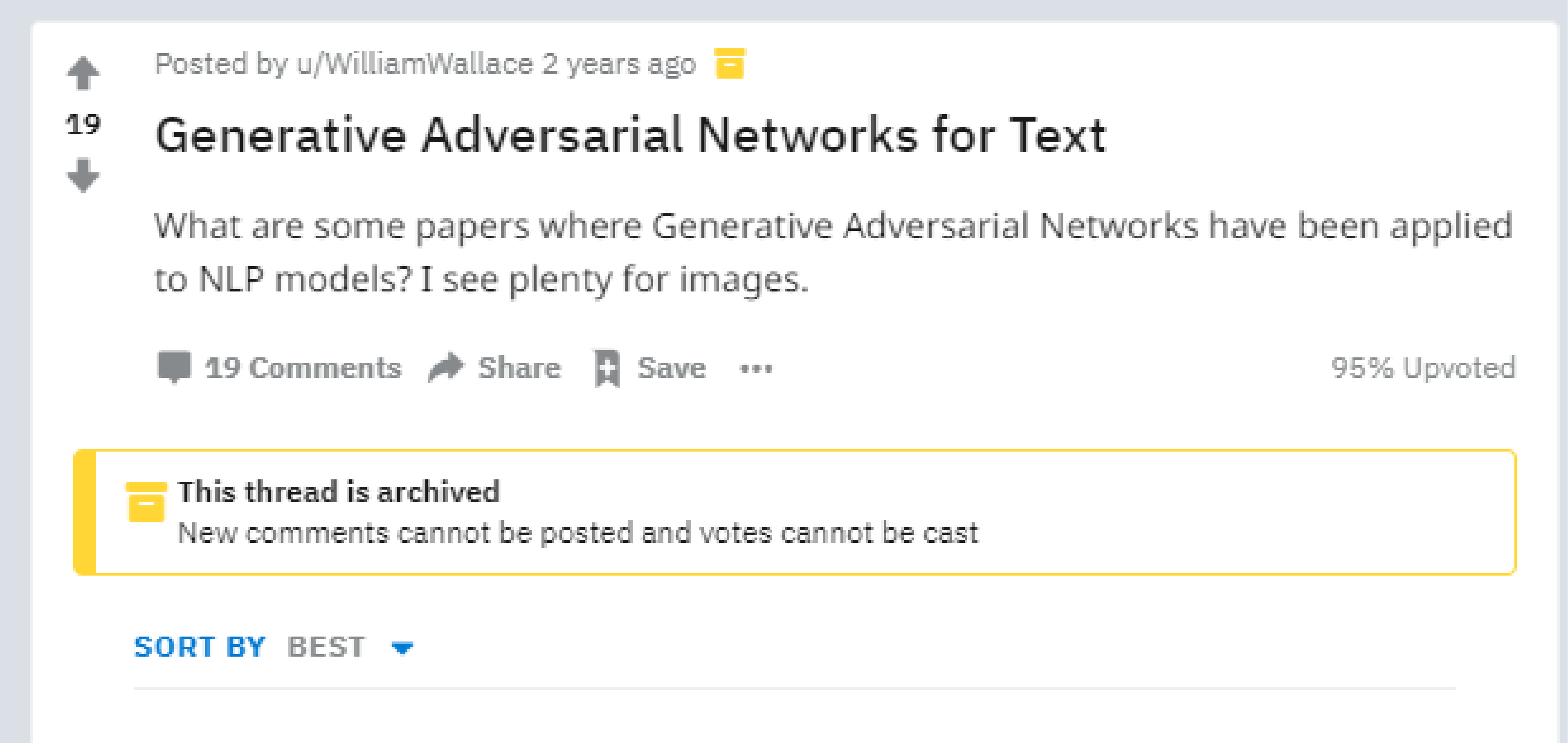
主要问题：1 随着生成的句子越来越长，RNN的误差会逐渐积累，后面的词会越来越离谱

2 从random latent representations生成的句子长度往往难以控制

Metrics

BLEU	一种基于精确度的相似性的度量方法，用于测量生成文本和参考译文中n元组共同出现的程度 常见的有BLEU-2, BLEU-3, BLEU-4
ROUGE	基于Recall的相似性度量方法，主要考察翻译的忠实性和充分性，但是难以考察流畅度 常见的有ROUGE-N，ROUGE-L，ROUGE-W，ROUGE-S四类
METEOR	基于recall和precision的调和平均
CIDEr	基于共识的评价标准
人工评价	除了客观的自动化度量标准以外，人工主观的打分虽然效率不高，但是也是评价很多算 法性能的重要指标， 微软在其VTTChallenge2016中提出了三点主观评价标准： 1） 流畅度：评价生成语句的逻辑和可读性。 2） 相关性：评价生成语句是否包含与原视频段相关和重要的物体/动作/事件等。 3） 助盲性：评价生成语句对一个实力有缺陷的人去理解其表示的视频片段到底有多大的帮助。

GANs for Text Generation?



Posted by u/WilliamWallace 2 years ago

Generative Adversarial Networks for Text

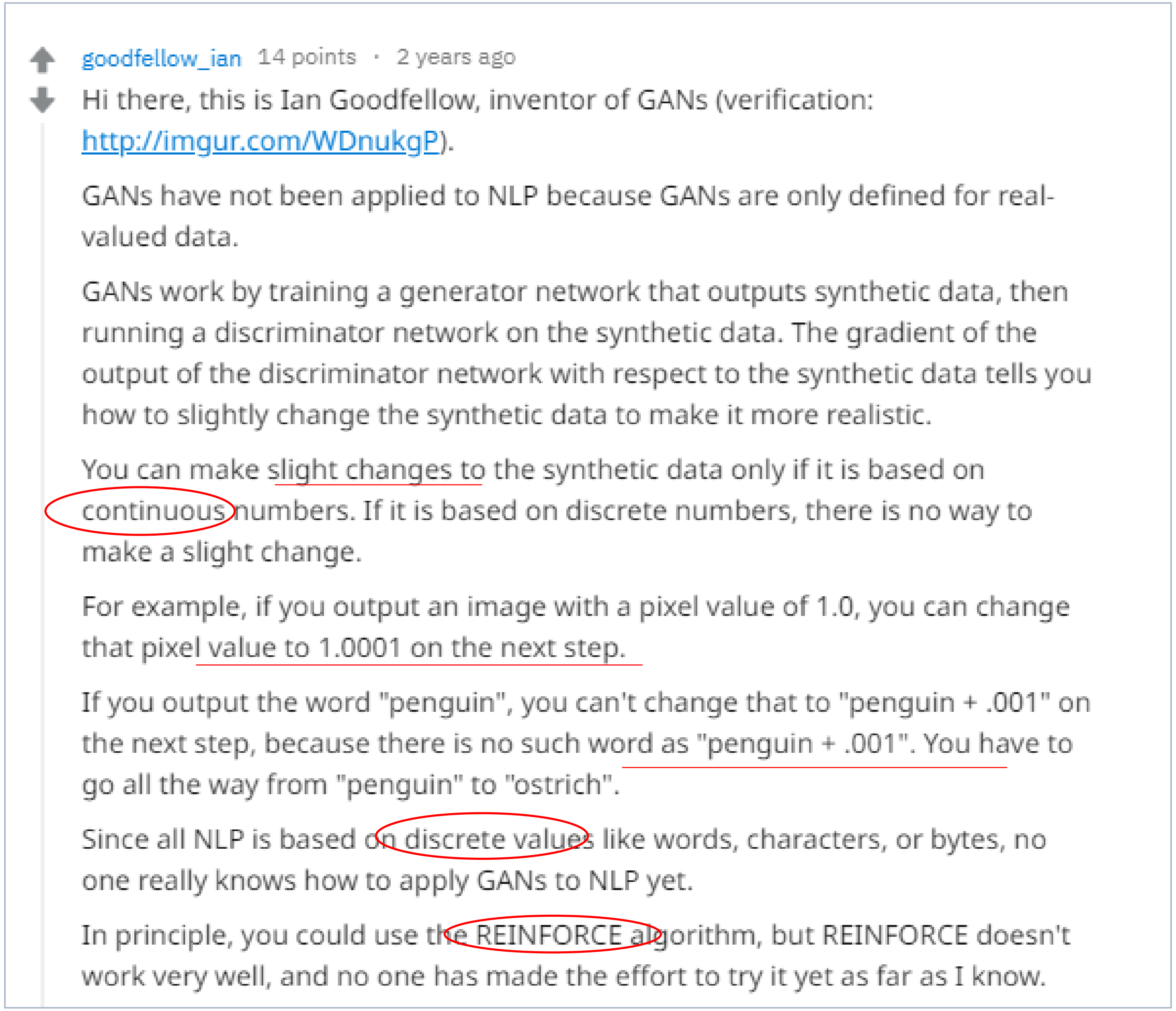
What are some papers where Generative Adversarial Networks have been applied to NLP models? I see plenty for images.

This thread is archived

New comments cannot be posted and votes cannot be cast

<LINK>

GANs for Text Generation?



Hi there, this is Ian Goodfellow, inventor of GANs (verification: http://imgur.com/WDnukgP).

GANs have not been applied to NLP because GANs are only defined for real-valued data.

GANs work by training a generator network that outputs synthetic data, then running a discriminator network on the synthetic data. The gradient of the output of the discriminator network with respect to the synthetic data tells you how to slightly change the synthetic data to make it more realistic.

You can make slight changes to the synthetic data only if it is based on continuous numbers. If it is based on discrete numbers, there is no way to make a slight change.

For example, if you output an image with a pixel value of 1.0, you can change that pixel value to 1.0001 on the next step.

If you output the word "penguin", you can't change that to "penguin + .001" on the next step, because there is no such word as "penguin + .001". You have to go all the way from "penguin" to "ostrich".

Since all NLP is based on discrete values like words, characters, or bytes, no one really knows how to apply GANs to NLP yet.

In principle, you could use the REINFORCE algorithm, but REINFORCE doesn't work very well, and no one has made the effort to try it yet as far as I know.

<LINK>

Papers Review

- Adversarial Generation of Natural Language (2016)

  {Sai Rajeswar, Sandeep Subramanian, Francis Dutil, Christopher Pal, Aaron Courville} @ Umontreal MILA

  注：蒙特利尔大学2017年暑期 深度学习&强化学习 课程视频/课件 在网上可以看

- MaskGAN: Better Text Generation via Filling in the ______

  {William Fedus, Ian Goodfellow and Andrew M. Dai}@Google Grain (ICLR 2018)

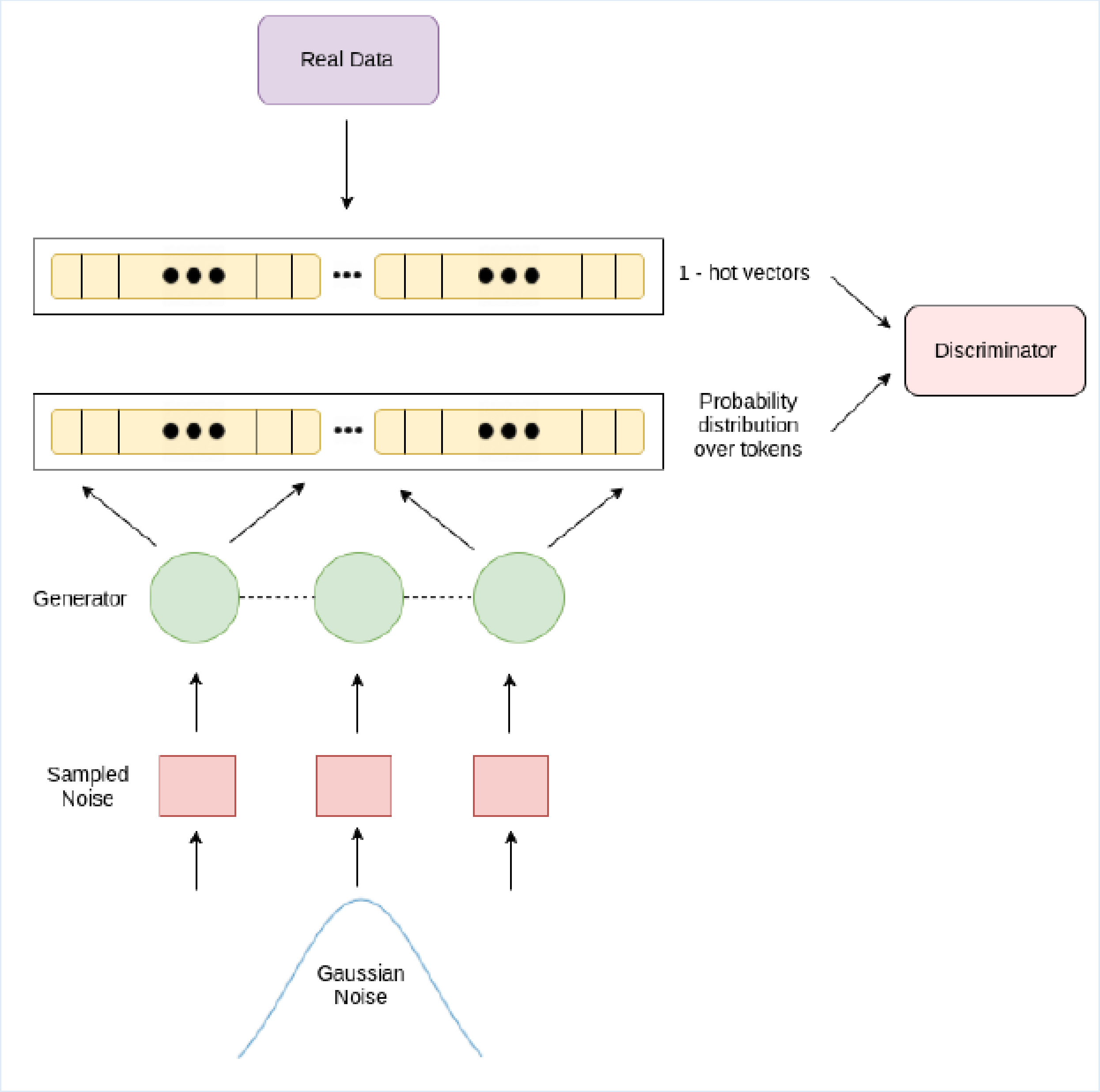
- Generating Text via Adversarial Training (NIPS workshop 2016)

  {Yizhe Zhang, Zhe Gan, Lawrence Carin} @ Duke University

- Adversarial Feature Matching for Text Generation (PMLR 2017)

  {Yizhe Zhang, Zhe Gan, Kai Fan, Zhi Chen, Ricardo Henao, Dinghan Shen 1 Lawrence Carin}@Duke University

Adversarial Generation of Natural Language (2016)



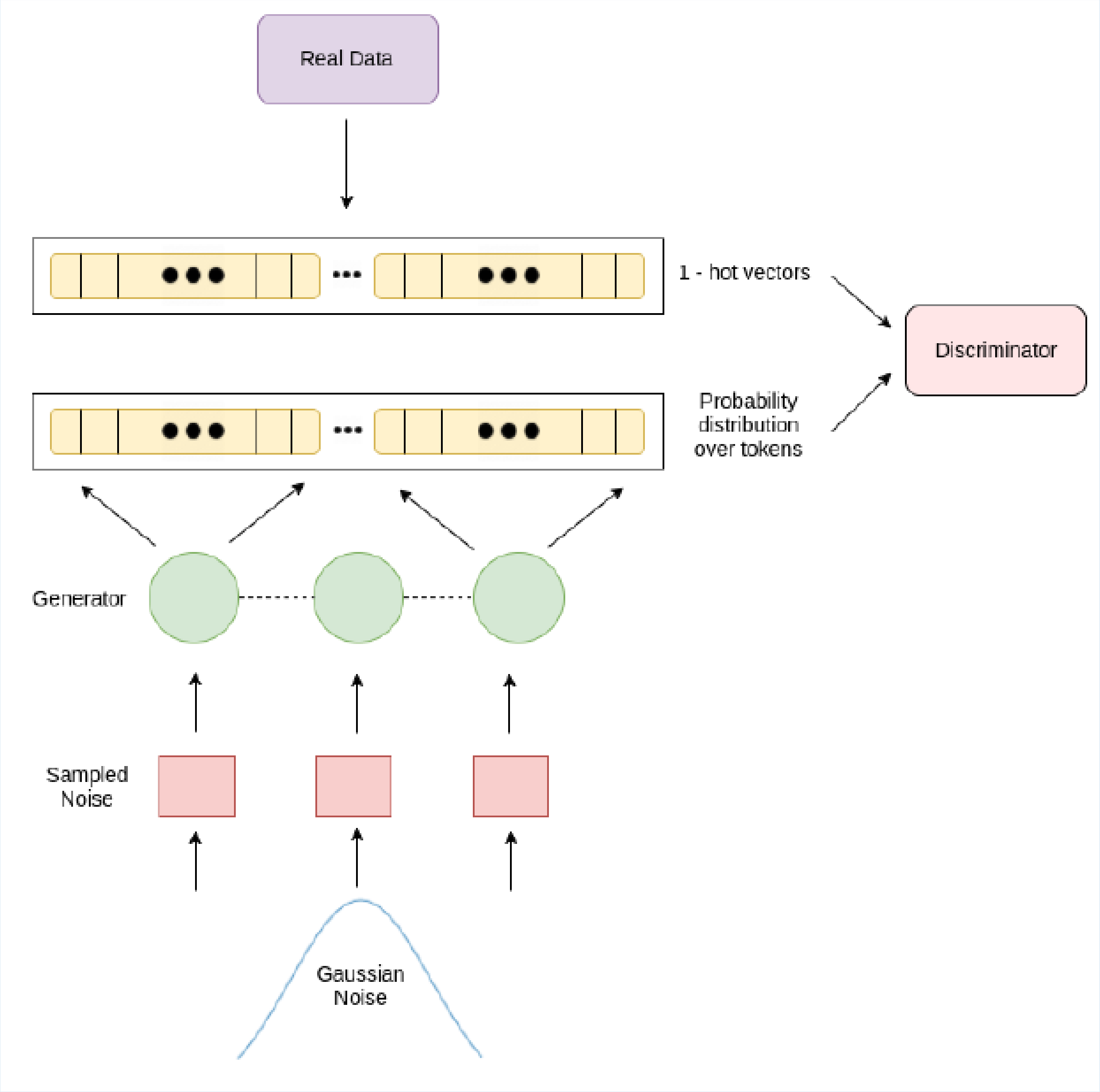
$Z \sim N(0, I)$

the sample z is of shape $n \times d$ where n is the length of sequence and d is a fixed length dimension of the noise vector at each time step.

The generator then transforms z into a sequence of probability distributions over the vocabulary $\mathrm{G}(z)$ of size $n \times k$ where k is the size of our true data distribution's vocabulary.

In this work, we address the discrete output space problem by simply forcing the discriminator to operate on continuous valued output distributions. The discriminator sees a sequence of probabilities over every token in the vocabulary from the generator and a sequence of 1-hot vectors from the true data distribution.

Adversarial Generation of Natural Language (2016)

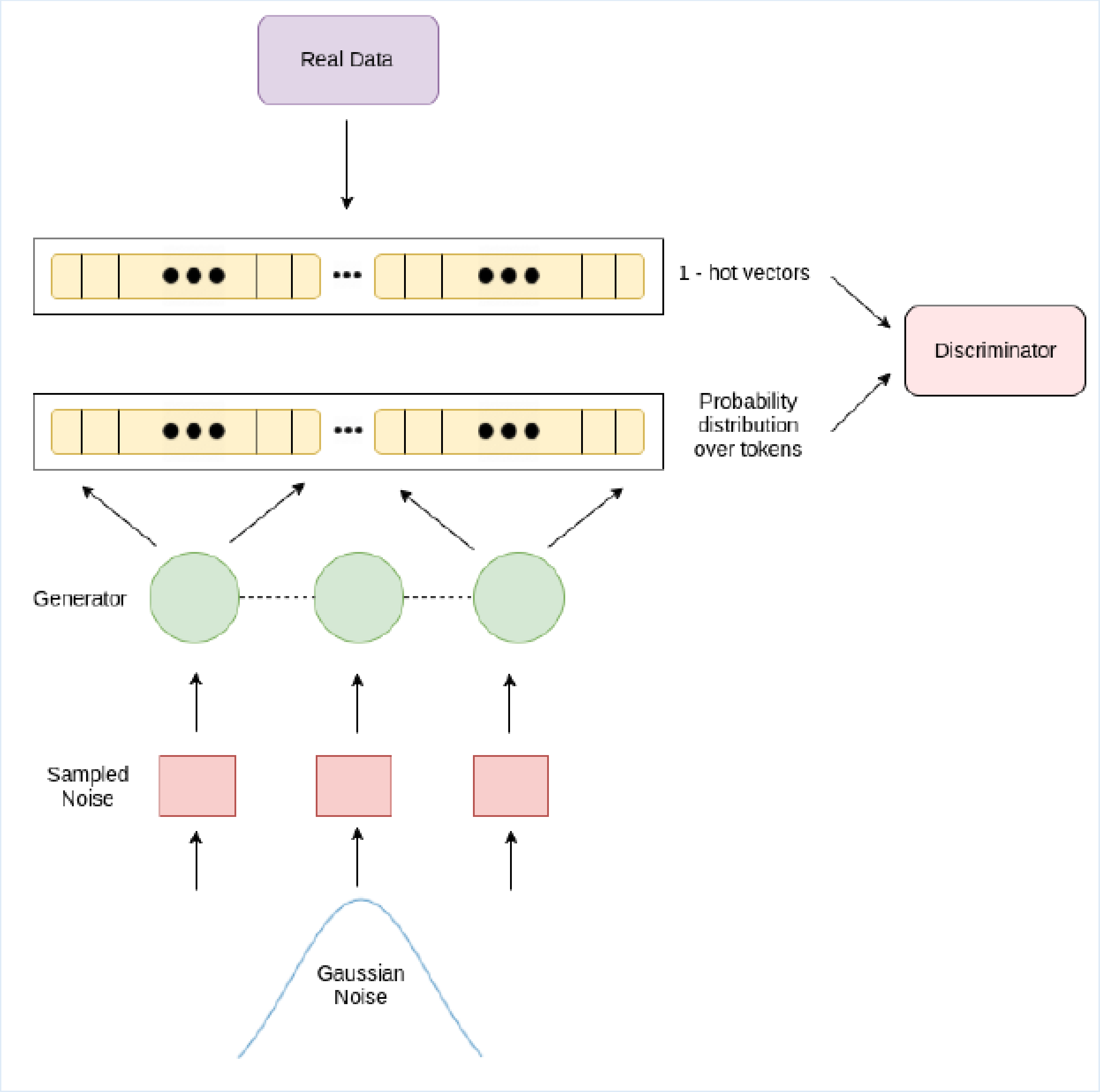


1 Use recurrent and convolutional architectures in both the generator as well as the discriminator.

2 When using the LSTM as a discriminator we use a simple binary logistic regression layer on the last hidden state h_n to determine the probability.

3 Curriculum learning: our generator is encouraged to generate entire sequences that match the true data distribution without explicit supervision at each step of the generation process. Gradually increasing lengths as training progresses.
(序列一开始就太长会导致训练难收敛，所以逐步增长。MaskGAN也用了类似的思想)

Adversarial Generation of Natural Language (2016)



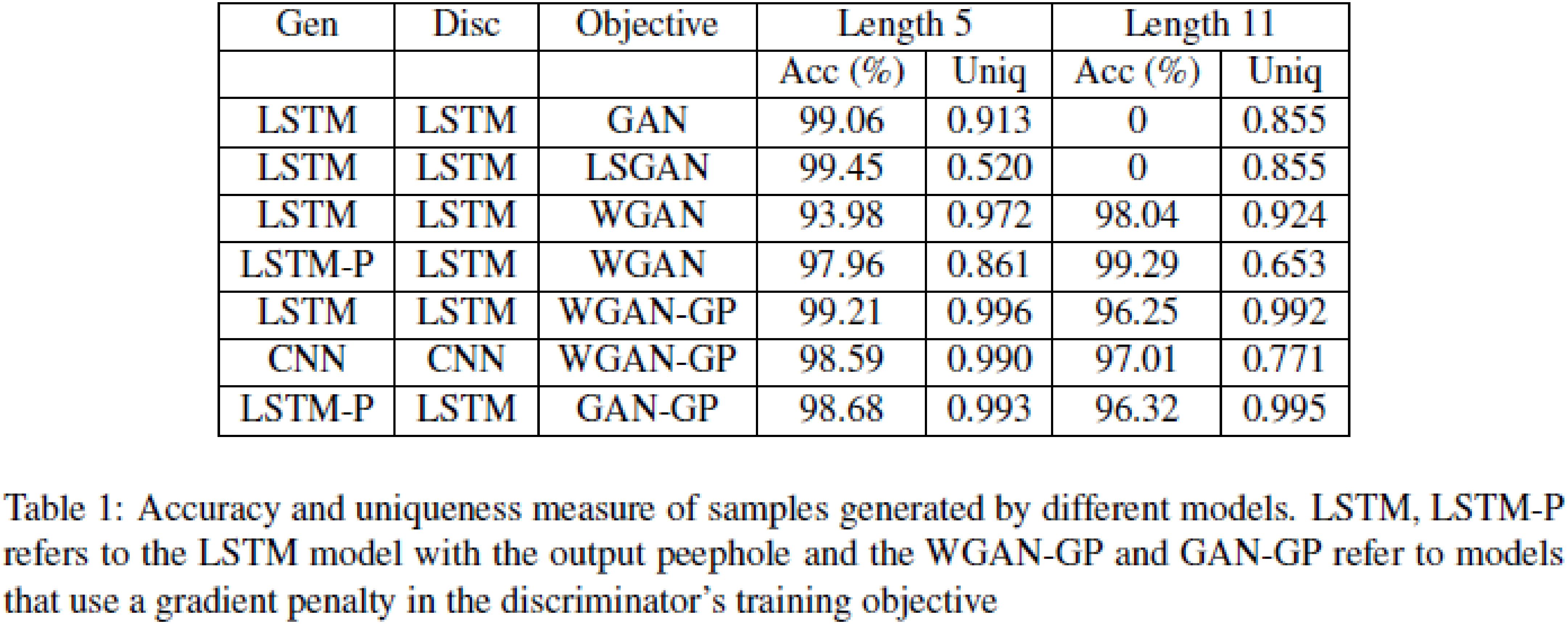
假设要把对抗式训练（adversarial training）推广到离散序列上，生成器使用RNN。一个主要的技术难点是：RNN每一步输出的是一个多项分布（取完softmax后得到的每个词的概率），但实际生成序列的时候，每一步只能取某一个词（one-hot）。这个离散输出不可导，所以不能像给图像用的GAN里的生成器G那样做反向传播。

这篇论文的主要思路就是直接把得到的softmax分布喂给判别器D，这个分布是可导的……

然而这个时候，判别器D最后做的事情其实是：区分one-hot表示（因为real data还是离散的）与G输出的概率分布（softmax输出，连续可导）。这其实跟判断是否是自然语言已经没有毛关系了。

最后的效果变成：让生成器G产生尽可能接近one-hot的输出，强行认为自然语言==尖峰分布。

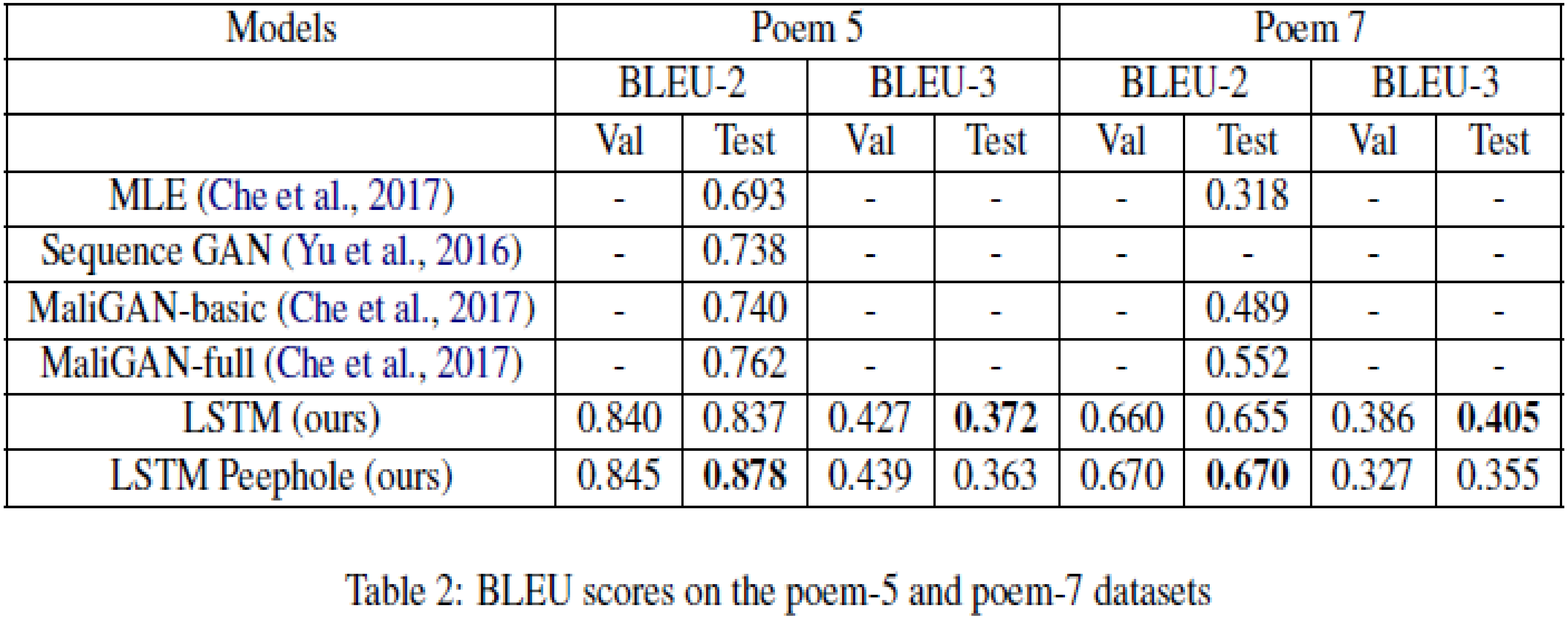
Adversarial Generation of Natural Language (2016)



Gen	Disc	Objective	Length 5		Length 11	
			Acc (%)	Uniq	Acc (%)	Uniq
LSTM	LSTM	GAN	99.06	0.913	0	0.855
LSTM	LSTM	LSGAN	99.45	0.520	0	0.855
LSTM	LSTM	WGAN	93.98	0.972	98.04	0.924
LSTM-P	LSTM	WGAN	97.96	0.861	99.29	0.653
LSTM	LSTM	WGAN-GP	99.21	0.996	96.25	0.992
CNN	CNN	WGAN-GP	98.59	0.990	97.01	0.771
LSTM-P	LSTM	GAN-GP	98.68	0.993	96.32	0.995

Table 1: Accuracy and uniqueness measure of samples generated by different models. LSTM, LSTM-P refers to the LSTM model with the output peephole and the WGAN-GP and GAN-GP refer to models that use a gradient penalty in the discriminator's training objective

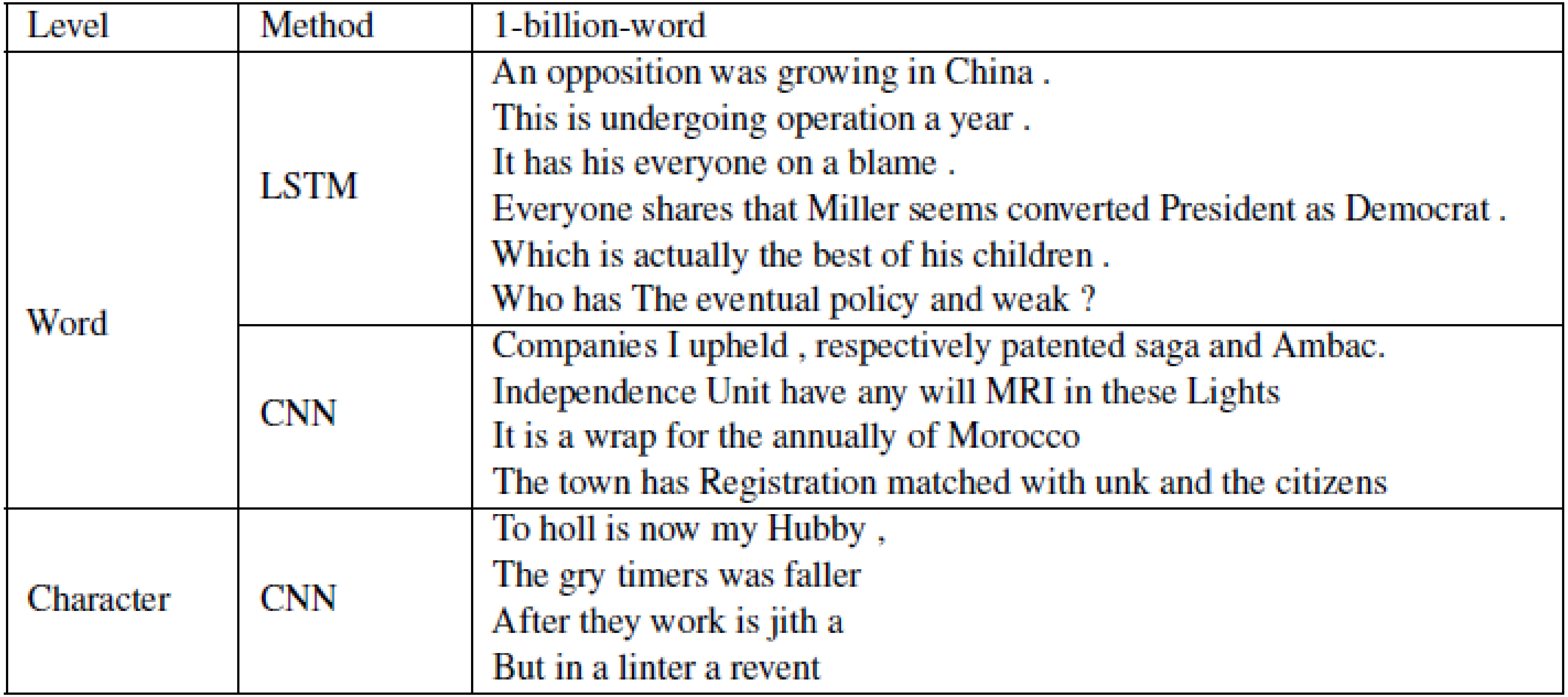
Adversarial Generation of Natural Language (2016)



Models	Poem 5				Poem 7			
	BLEU-2		BLEU-3		BLEU-2		BLEU-3	
	Val	Test	Val	Test	Val	Test	Val	Test
MLE (Che et al., 2017)	-	0.693	-	-	-	0.318	-	-
Sequence GAN (Yu et al., 2016)	-	0.738	-	-	-	-	-	-
MaliGAN-basic (Che et al., 2017)	-	0.740	-	-	-	0.489	-	-
MaliGAN-full (Che et al., 2017)	-	0.762	-	-	-	0.552	-	-
LSTM (ours)	0.840	0.837	0.427	**0.372**	0.660	0.655	0.386	**0.405**
LSTM Peephole (ours)	0.845	**0.878**	0.439	0.363	0.670	**0.670**	0.327	0.355

Table 2: BLEU scores on the poem-5 and poem-7 datasets

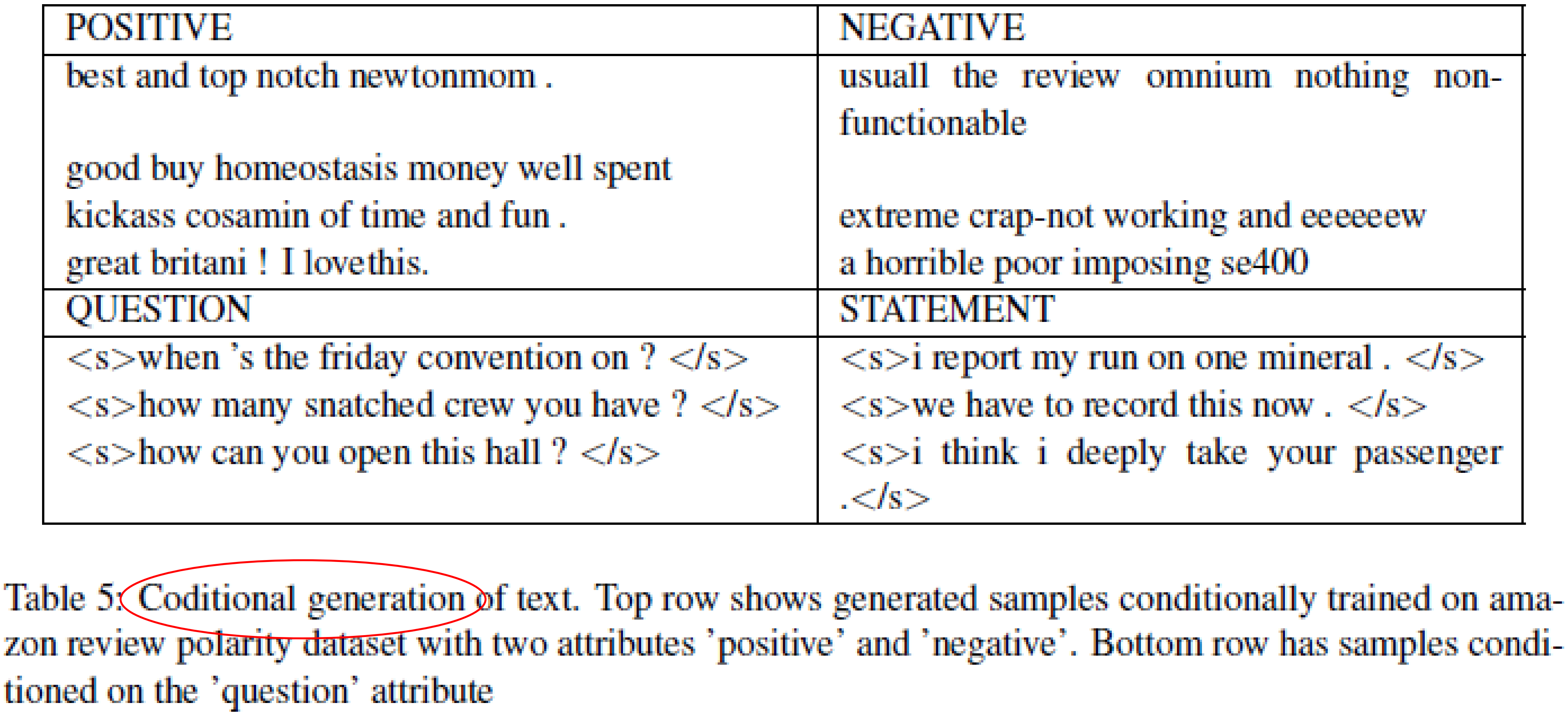
Adversarial Generation of Natural Language (2016)



Level	Method	1-billion-word
Word	LSTM	An opposition was growing in China . This is undergoing operation a year . It has his everyone on a blame . Everyone shares that Miller seems converted President as Democrat . Which is actually the best of his children . Who has The eventual policy and weak ?
	CNN	Companies I upheld , respectively patented saga and Ambac. Independence Unit have any will MRI in these Lights It is a wrap for the annually of Morocco The town has Registration matched with unk and the citizens
Character	CNN	To holl is now my Hubby , The gry timers was faller After they work is jith a But in a linter a revent

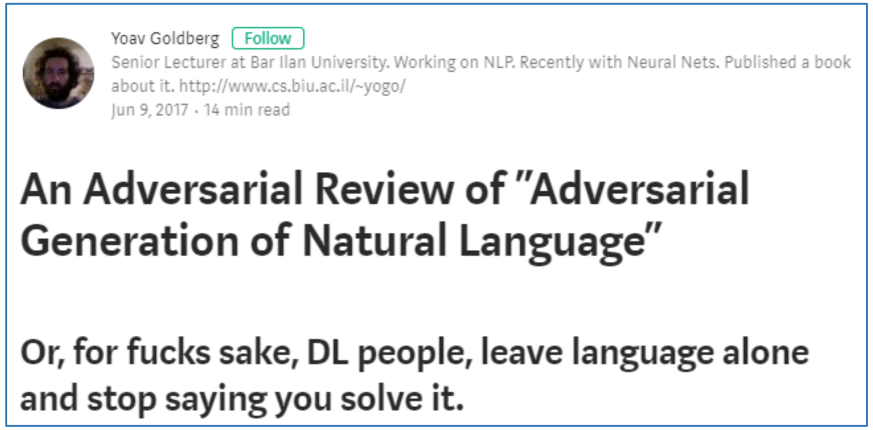
Table 3: Word and character-level generations on the 1-billion word dataset

Adversarial Generation of Natural Language (2016)



POSITIVE	NEGATIVE
best and top notch newtonmom . good buy homeostasis money well spent kickass cosamin of time and fun . great britani ! I lovethis.	usuall the review omnium nothing non-functionable extreme crap-not working and eeeeeew a horrible poor imposing se400
QUESTION	STATEMENT
<s>when ’s the friday convention on ? </s> <s>how many snatched crew you have ? </s> <s>how can you open this hall ? </s>	<s>i report my run on one mineral . </s> <s>we have to record this now . </s> <s>i think i deeply take your passenger .</s>

Table 5: Coditional generation of text. Top row shows generated samples conditionally trained on amazon review polarity dataset with two attributes ’positive’ and ’negative’. Bottom row has samples conditioned on the ’question’ attribute



Yoav Goldberg Follow

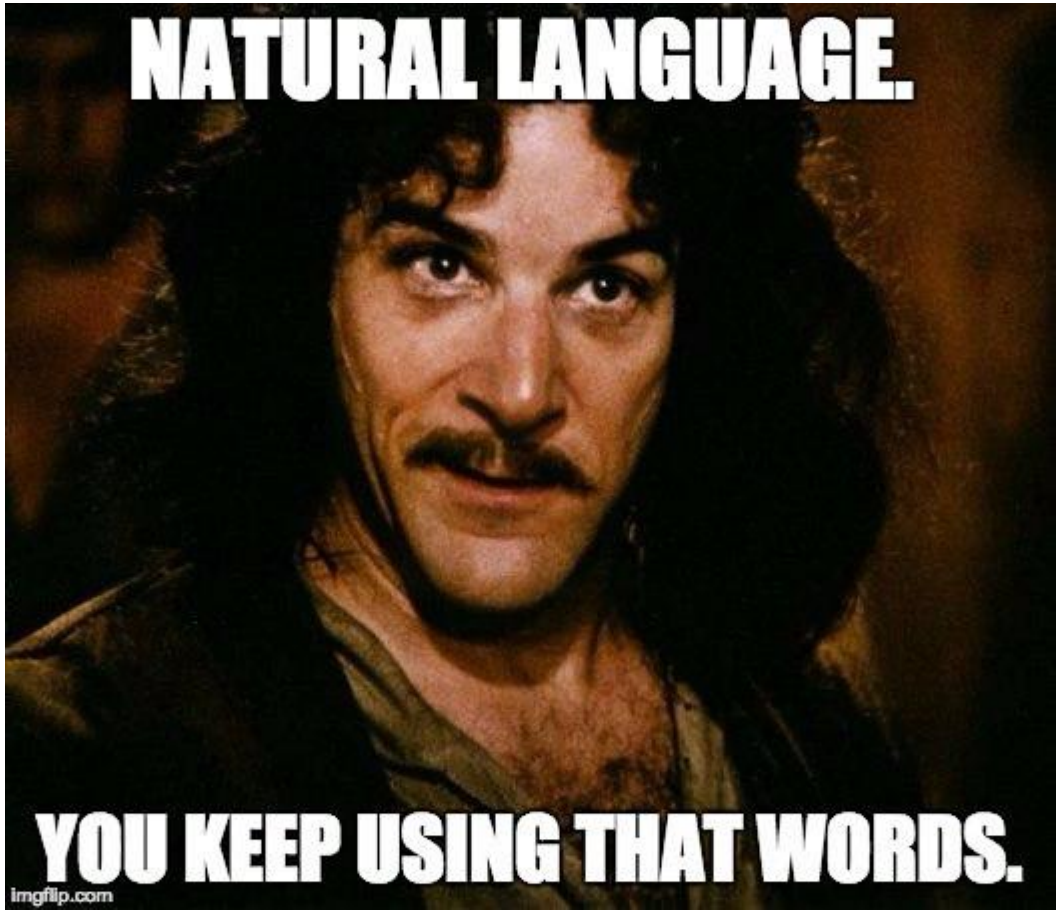
Senior Lecturer at Bar Ilan University. Working on NLP. Recently with Neural Nets. Published a book about it. http://www.cs.biu.ac.il/~yogo/

Jun 9, 2017 · 14 min read

An Adversarial Review of "Adversarial Generation of Natural Language"

Or, for fucks sake, DL people, leave language alone and stop saying you solve it.

Yoav Goldberg



<LINK>

What did Yoav Goldberg say ?

1 MILA那篇论文用了两套作者们自己都没仔细研究过的简单PCFG来产生语言，然后用这个语言语句的似然函数来评价生成效果。但大家都知道自然语言显然不是PCFG能建模的。有限语料库上导出的PCFG生成概率也并不能代表语法流畅度。同时，他们效仿先前工作，也在中文古诗数据上做了点实验。且不论实验用的诗句按长度看只有五言七言这么短的长度，所有这些工作最后评价的时候都只是孤立地去评判每一行。更甚者，评价方式不是去让人判断生成质量，而仅仅是算个BLEU完事。

2 如果你是审稿人：审稿的时候请一定要尊重自然语言，不要被做法花哨、实际上只能处理极简化情形的overclaims蒙蔽双眼。一定要看他们如何进行了什么样的实验评估、实验结果能证明什么结论，而不是他们在论文里宣称提出了什么方法达到什么效果。更不要强求处理真实数据的NLP研究人员去引用、比较那些质量底下或者缺陷明显的“开创性论文”。

3 如果你是论文作者：尊重并试图更多了解自然语言，真正明白自己实验用的数据集、评价汇报的那些数值指标是否就是真正能验证自己的研究发现的东西。搞清楚自己在做什么，不要忘了和最明显的baseline进行对照。同时在论文中尽可能点明自己研究内容的局限性。

<LINK>

MASKGAN: Better Text Generation via Filling in the ______

$$G(x_t) \equiv P(\hat{x}_t | \hat{x}_1, \cdots, \hat{x}_{t-1}, m(x))$$

仍然用的argmax，所以D->G无法梯度下降,需借助强化学习policy gradient

如果没有上下文的限制就退化为普通的文本生成模型

The Generator

Generator infills missing text.

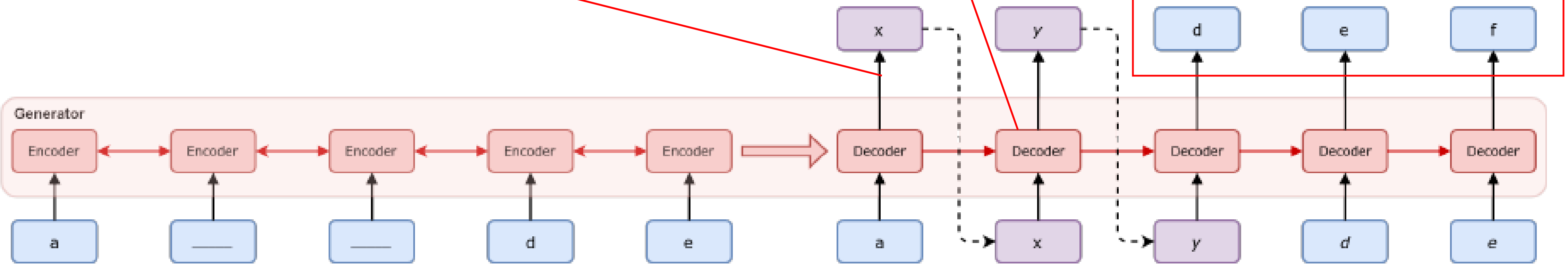
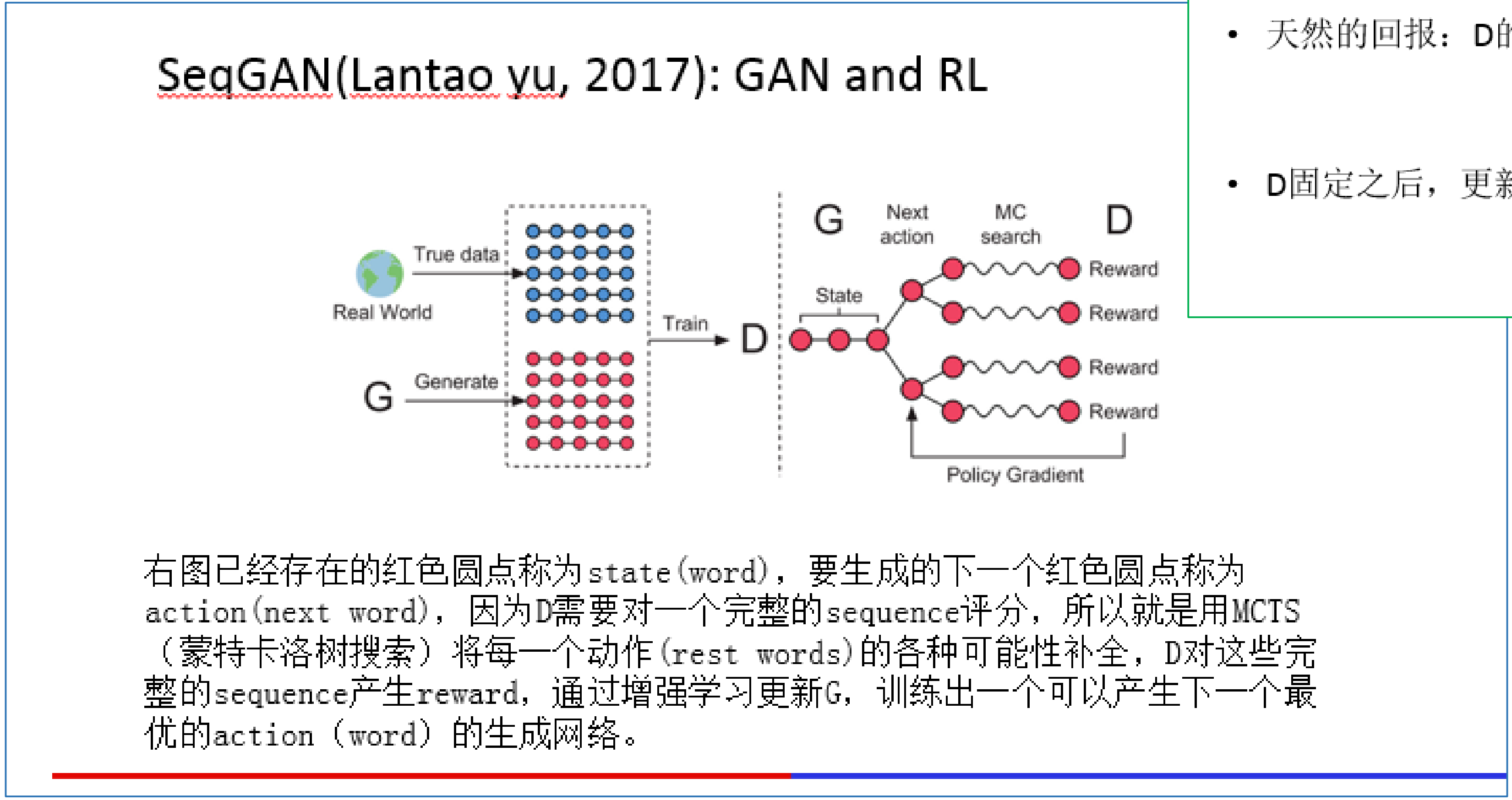


Figure 1: seq2seq generator architecture. Blue boxes represent known tokens and purple boxes are imputed tokens. We demonstrate a sampling operation via the dotted line. The encoder reads in a masked sequence, where masked tokens are denoted by an underscore, and then the decoder imputes the missing tokens by using the encoder hidden states. In this example, the generator should fill in the alphabetical ordering, (a,b,c,d,e).

MaskGAN的 generator 和 discriminator 都是采用一样的seq2seq 结构（如上图）。其中，generator 是采用强化学习的方法训练的，discriminator 是采用以往maximum likelihood + SGD 方式训练。

MASKGAN: Better Text Generation via Filling in the ______



SeqGAN(Lantao yu, 2017): GAN and RL

右图已经存在的红色圆点称为state(word)，要生成的下一个红色圆点称为action(next word)，因为D需要对一个完整的sequence评分，所以就是用MCTS（蒙特卡洛树搜索）将每一个动作(rest words)的各种可能性补全，D对这些完整的sequence产生reward，通过增强学习更新G，训练出一个可以产生下一个最优的action（word）的生成网络。

- 生成器G的目标是生成sequence来最大化reward的期望J(θ)

$$J(\theta) = \mathbb{E}[R_T|s_0, \theta] = \sum_{y_1 \in \mathcal{Y}} G_\theta(y_1|s_0) \cdot Q_{D_\phi}^{G_\theta}(s_0, y_1)$$

s_0和θ的条件下，产生某个完全的sequence的reward的期望

$G_\theta(y_1|s_0)$：生成y_1 的概率　　$Q_{D_\phi}^{G_\theta}(s_0, y_1)$ 生成y_1 的回报

- 天然的回报：D的返回值

$$Q_{D_\phi}^{G_\theta}(a = y_T, s = Y_{1:T-1}) = D_\phi(Y_{1:T}).$$

- D固定之后，更新G的参数

$$\theta \leftarrow \theta + \alpha_h \nabla_\theta J(\theta)$$

MASKGAN: Better Text Generation via Filling in the ______

1、Discriminator是和Generator一样的架构，用的LSTM。作者也尝试了CNN，但是发现效果不好。Discriminator 是采用传统的maximum likelihood + SGD 方式训练。

2、还有很重要的一点，传给D的不只是生成的填空词，而且还有填空词的上下文。

因为假设我们要填空完成这么一句话：the ______ ______ guided the series，现在我们的模型生成了the director director guided the series，然后真实数据中同时存在the *associate* director guided the series 和 the director *expertly* guided the series 这两个答案。

假设我们没有给D上下文，那么D就无法正确惩罚这两个director了，因为这两个director都可能是正确的。现在有了上下文，D可以先计算两个director各自出现的概率，再根据概率进行惩罚。

3、本文用强化学习的policy gradient进行梯度下降，避免了文本的discrete问题。用RL还可以让D的reward指定为你需要的指标，例如比如BLEU，而不仅仅是做一个真假判别

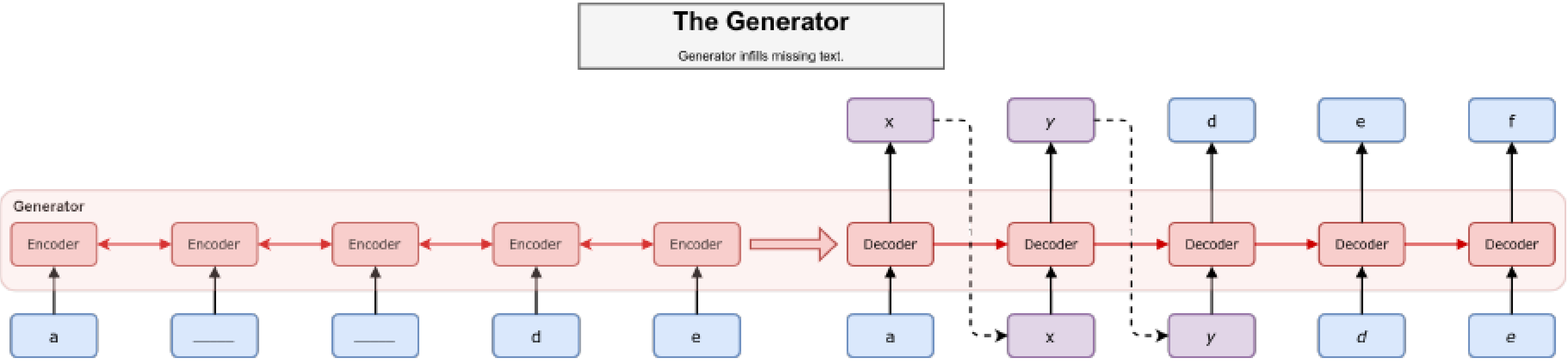
Generator

$$\nabla_\theta \mathbb{E}[R] = \mathbb{E}_{\hat{x}_t \sim G}\left[\sum_{t=1}^{T}(R_t - b_t)\nabla_\theta \log(G_\theta(\hat{x}_t))\right]$$
$$= \mathbb{E}_{\hat{x}_t \sim G}\left[\sum_{t=1}^{T}\left(\sum_{s=t}^{T}\gamma^s r_s - b_t\right)\nabla_\theta \log(G_\theta(\hat{x}_t))\right]$$

Discriminator

$$\nabla_\phi \frac{1}{m}\sum_{i=1}^{m}\left[\log D(x^{(i)})] + \log(1 - D(G(z^{(i)})\right]$$

MASKGAN: Better Text Generation via Filling in the ______



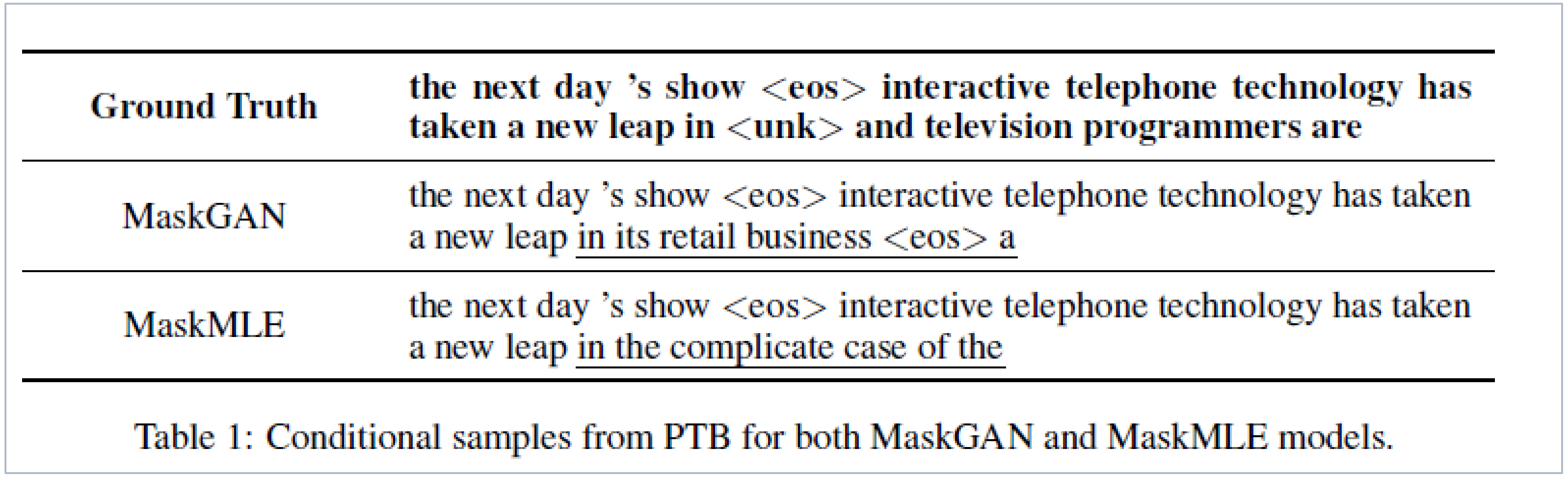
主要理解两点：

1、普通seq2seq训练时，不管你预测出什么，下一个词都会填入ground truth 的词，相当于普通gan训练时，都用标准答案妨碍了generator的探索过程。这也叫做teacher forcing.然后训练用的是maximum likelihood来调参。而MaskGAN中，训练时预测出什么，下一个词就传入什么，训练时也要用到上下文。

2、传统的seq2seq没有离散这个问题，因为用的是概率分布。现在因为要生成一句话给Discriminator，所以要用argmax得到一句完整的话。这才导致了离散的问题。所以引进了RL来解决这个问题。

注：作者在3.4节提到自己训练时为了训练稳定，sequence的长度是逐渐增加的。同时RL的reward利用了全部的generator distribution，而不仅仅是生成的那个token。

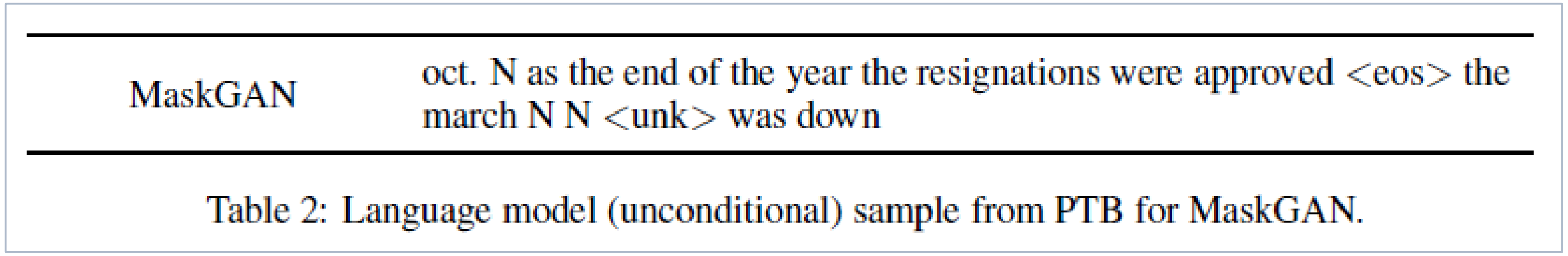
MASKGAN: Better Text Generation via Filling in the ______



Ground Truth	**the next day 's show <eos> interactive telephone technology has taken a new leap in <unk> and television programmers are**
MaskGAN	the next day 's show <eos> interactive telephone technology has taken a new leap in its retail business <eos> a
MaskMLE	the next day 's show <eos> interactive telephone technology has taken a new leap in the complicate case of the

Table 1: Conditional samples from PTB for both MaskGAN and MaskMLE models.

填空任务

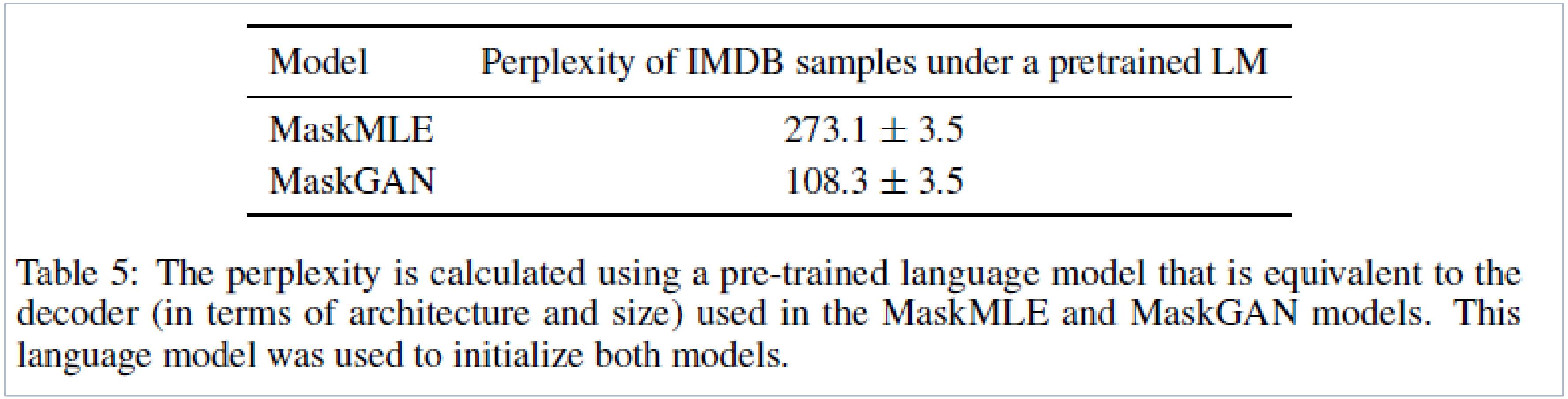


MaskGAN	oct. N as the end of the year the resignations were approved <eos> the march N N <unk> was down

Table 2: Language model (unconditional) sample from PTB for MaskGAN.

文本生成任务

MASKGAN: Better Text Generation via Filling in the ______



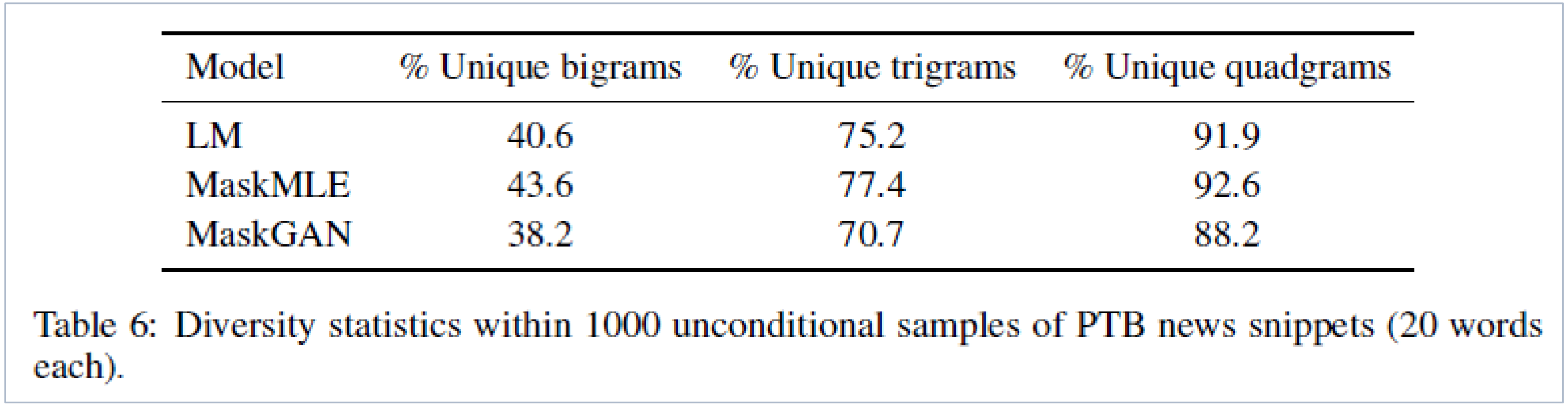
Model	Perplexity of IMDB samples under a pretrained LM
MaskMLE	273.1 ± 3.5
MaskGAN	108.3 ± 3.5

Table 5: The perplexity is calculated using a pre-trained language model that is equivalent to the decoder (in terms of architecture and size) used in the MaskMLE and MaskGAN models. This language model was used to initialize both models.

Evaluation on Perplexity

详见5.4节：

1、作者指出perplexity与sample quality没有必然的联系



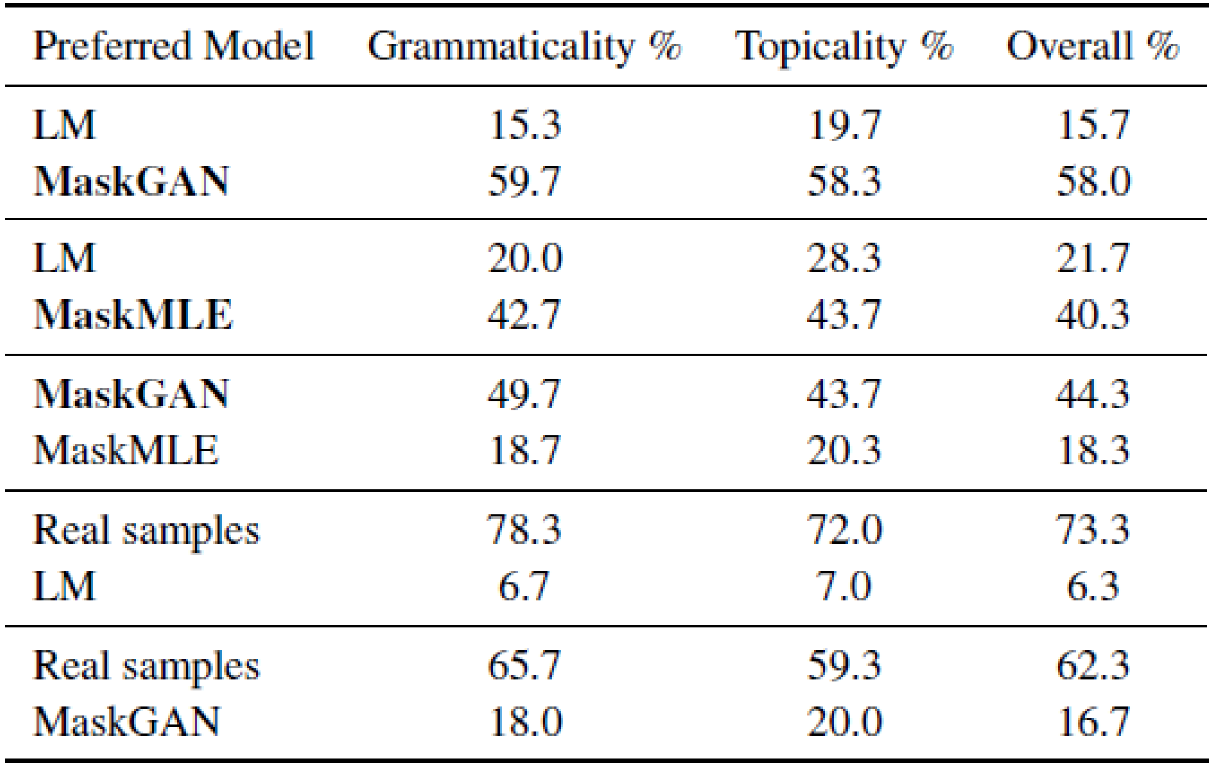
Model	% Unique bigrams	% Unique trigrams	% Unique quadgrams
LM	40.6	75.2	91.9
MaskMLE	43.6	77.4	92.6
MaskGAN	38.2	70.7	88.2

Table 6: Diversity statistics within 1000 unconditional samples of PTB news snippets (20 words each).

Evaluation on Mode Collapse

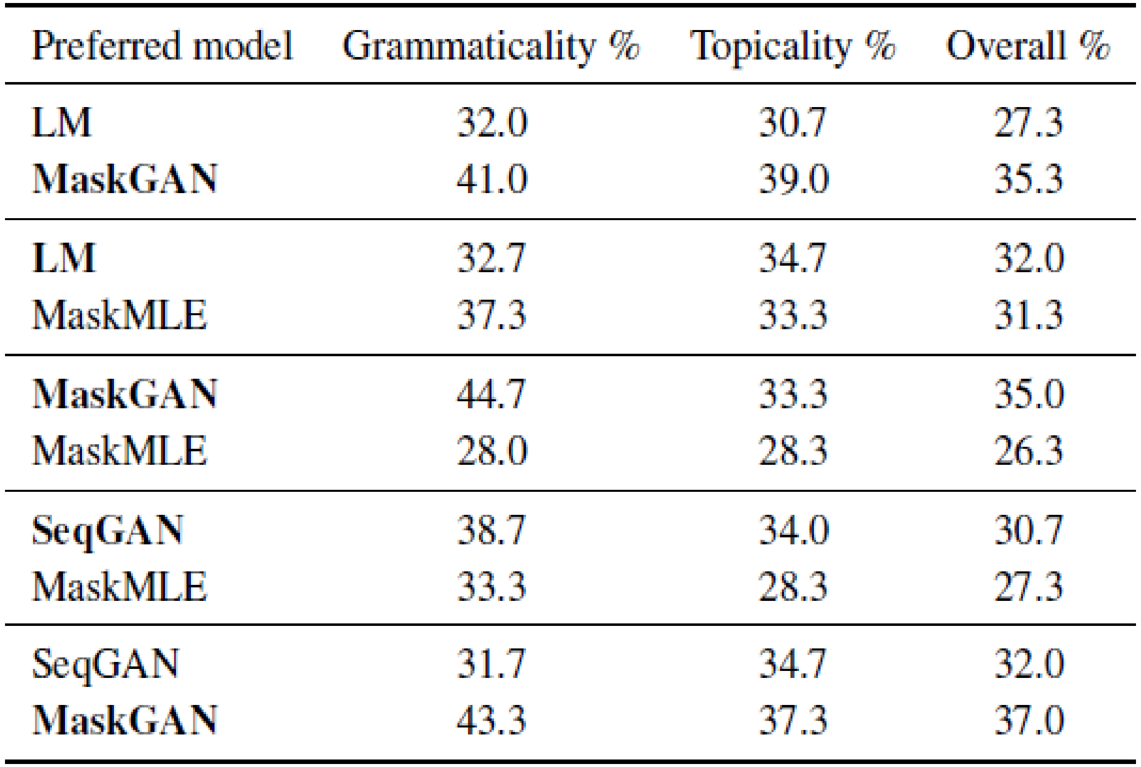
2、作者提到了自己这个模型会导致一定程度上的mode dropping。并且指出这个问题往往出现在长文本的尾部。

MASKGAN: Better Text Generation via Filling in the ______



Preferred Model	Grammaticality %	Topicality %	Overall %
LM	15.3	19.7	15.7
MaskGAN	59.7	58.3	58.0
LM	20.0	28.3	21.7
MaskMLE	42.7	43.7	40.3
MaskGAN	49.7	43.7	44.3
MaskMLE	18.7	20.3	18.3
Real samples	78.3	72.0	73.3
LM	6.7	7.0	6.3
Real samples	65.7	59.3	62.3
MaskGAN	18.0	20.0	16.7

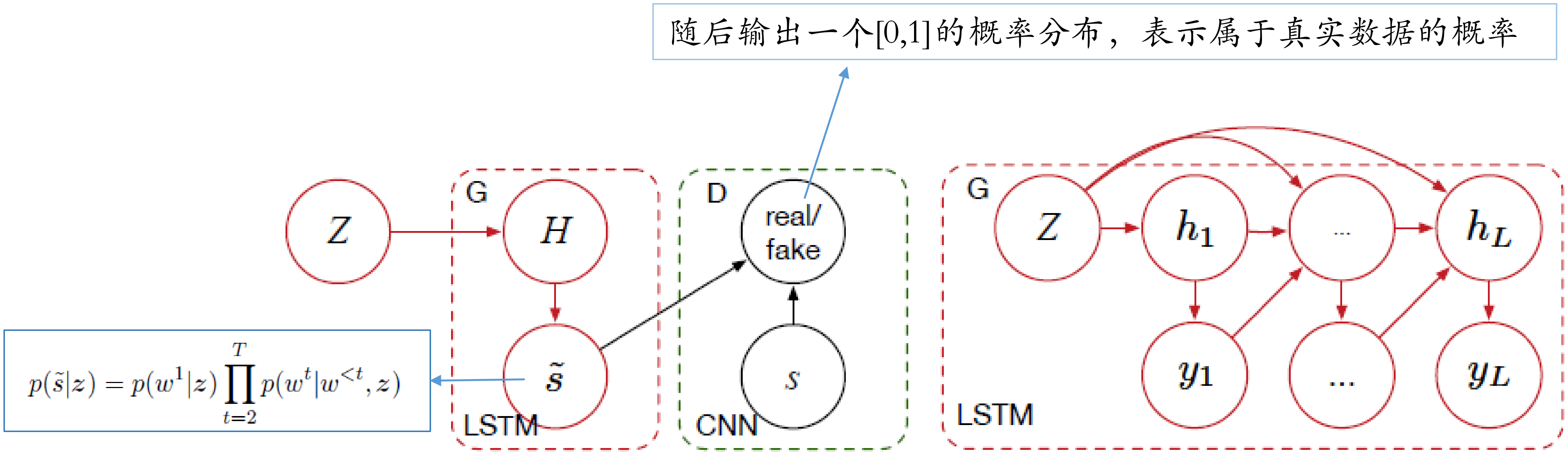
Human Evaluation on IMDB dataset



Preferred model	Grammaticality %	Topicality %	Overall %
LM	32.0	30.7	27.3
MaskGAN	41.0	39.0	35.3
LM	32.7	34.7	32.0
MaskMLE	37.3	33.3	31.3
MaskGAN	44.7	33.3	35.0
MaskMLE	28.0	28.3	26.3
SeqGAN	38.7	34.0	30.7
MaskMLE	33.3	28.3	27.3
SeqGAN	31.7	34.7	32.0
MaskGAN	43.3	37.3	37.0

Human Evaluation on PTB. 100 dataset

Generating Text via Adversarial Training (NIPS workshop 2016)



随后输出一个[0,1]的概率分布，表示属于真实数据的概率

$$p(\tilde{s}|z) = p(w^1|z)\prod_{t=2}^{T} p(w^t|w^{<t}, z)$$

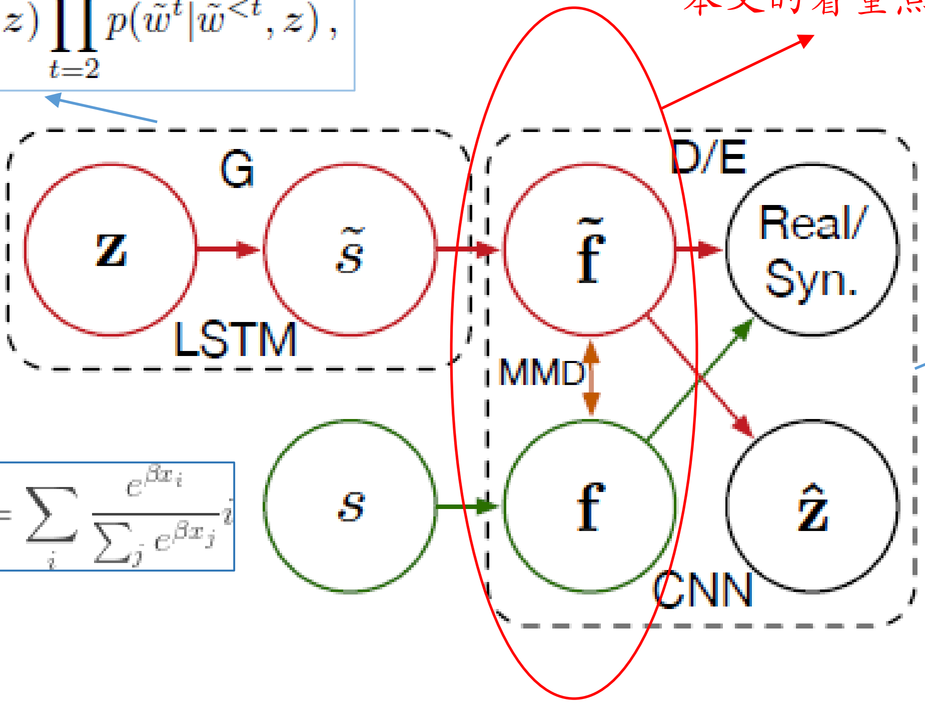
Figure 1: Left: Illustration of the textGAN model. The discriminator is a CNN, the sentence decoder is an LSTM. Right: the structure of LSTM model

隐向量 Z 经过一个LSTM生成一句话，然后交给CNN discriminator来判断真假

Adversarial Feature Matching for Text Generation (PMLR 2017)

$$p(\tilde{s}|z) = p(\tilde{w}^1|z) \prod_{t=2}^{T} p(\tilde{w}^t|\tilde{w}^{<t}, z) \,,$$

本文的着重点在于迫使latent-feature的分布趋同



$$\mathrm{softargmax}(x) = \sum_i \frac{e^{\beta x_i}}{\sum_j e^{\beta x_j}} i$$

Figure 1. Model scheme of TextGAN. Latent codes z are fed through a generator $G(\cdot)$, to produce synthetic sentence $\tilde{s}$. Synthetic and real sentences ($\tilde{s}$ and s) are fed into a binary discriminator $D(\cdot)$, for real *vs.* fake (synthetic) prediction, and also for latent code reconstruction $\hat{z}$. $\tilde{f}$ and f represent features of $\tilde{s}$ and s, respectively.

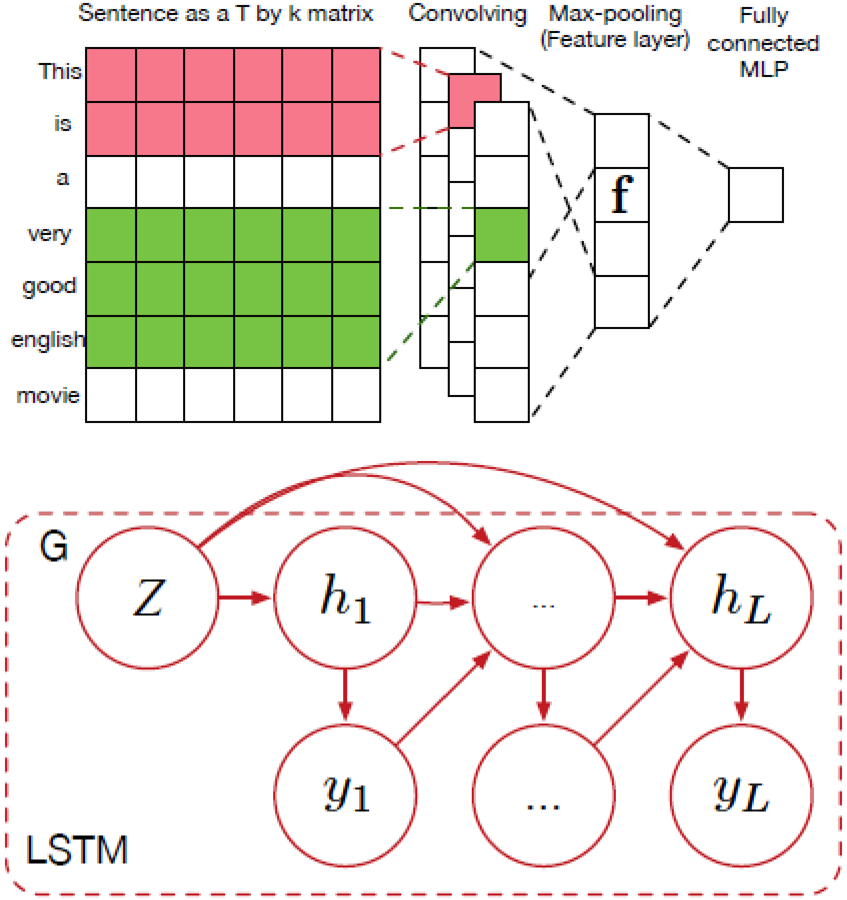


Figure 2. Top: CNN-based sentence discriminator/encoder. Bottom: LSTM sentence generator.

Adversarial Feature Matching for Text Generation (PMLR 2017)

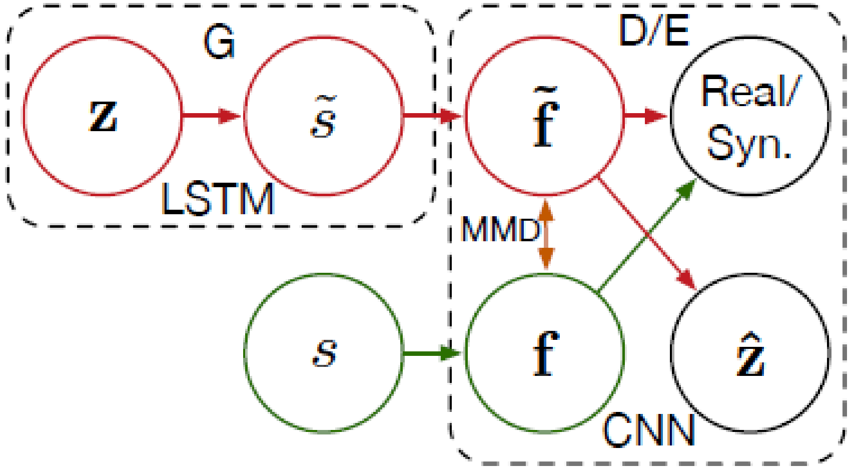


Figure 1. Model scheme of TextGAN. Latent codes z are fed through a generator $G(\cdot)$, to produce synthetic sentence $\tilde{s}$. Synthetic and real sentences ($\tilde{s}$ and s) are fed into a binary discriminator $D(\cdot)$, for real *vs.* fake (synthetic) prediction, and also for latent code reconstruction $\hat{z}$. $\tilde{f}$ and f represent features of $\tilde{s}$ and s, respectively.

$$
\begin{aligned}
\mathcal{L}_D &= \mathcal{L}_{GAN} - \lambda_r \mathcal{L}_{recon} + \lambda_m \mathcal{L}_{MMD^2} \\
\mathcal{L}_G &= \mathcal{L}_{MMD^2} \\
\mathcal{L}_{GAN} &= \mathbb{E}_{s \sim \mathcal{S}} \log D(s) + \mathbb{E}_{z \sim p_z} \log[1 - D(G(z))] \\
\mathcal{L}_{recon} &= ||\hat{z} - z||^2 \,,
\end{aligned}
$$

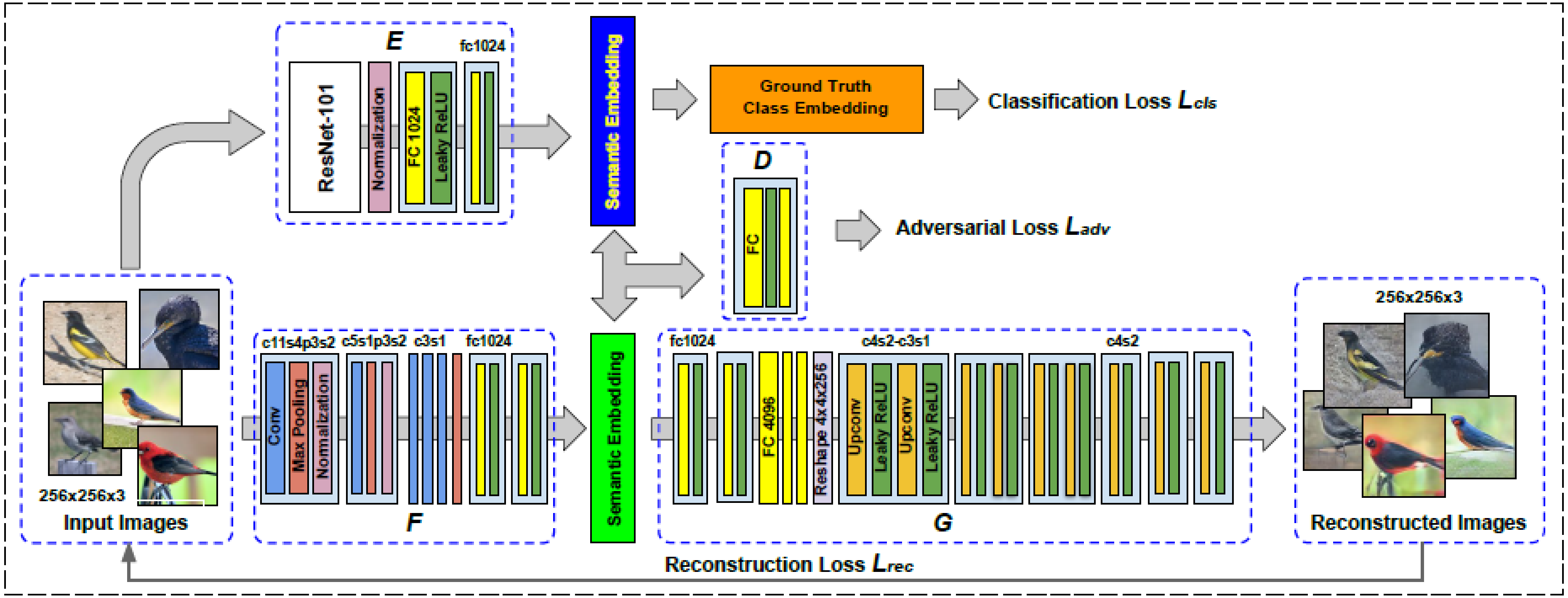
L_{GAN}想让和$\tilde{f}$和f尽量可区分，L_{recon}想让$\tilde{f}$尽量保留重构信息，L_{MMD2}想让两个尽量分布吻合

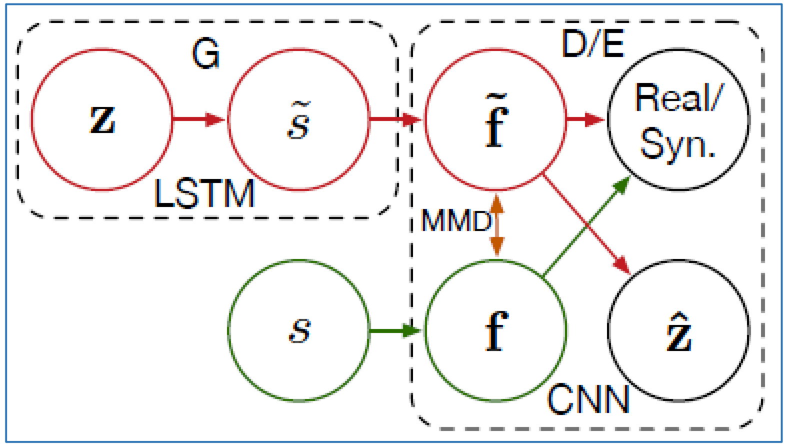
$$
\begin{aligned}
\mathcal{L}_{MMD^2} &= ||\mathbb{E}_{x \sim \mathcal{X}} \phi(x) - \mathbb{E}_{y \sim \mathcal{Y}} \phi(y)||^2_{\mathcal{H}} \qquad (4) \\
&= \mathbb{E}_{x \sim \mathcal{X}} \mathbb{E}_{x' \sim \mathcal{X}} [k(x, x')] \\
&\quad + \mathbb{E}_{y \sim \mathcal{Y}} \mathbb{E}_{y' \sim \mathcal{Y}} [k(y, y')] - 2\mathbb{E}_{x \sim \mathcal{X}} \mathbb{E}_{y \sim \mathcal{Y}} [k(x, y)].
\end{aligned}
$$

当两个分布完全重合的时候，MMD^2取到最小值。文中作者用了高斯分布的协方差性质，把MMD^2用下式替代。

$$
\begin{aligned}
\mathcal{L}_G^{(c)} &= \mathrm{tr}(\tilde{\Sigma}^{-1}\Sigma + \Sigma^{-1}\tilde{\Sigma}) \\
&\quad + (\tilde{\mu} - \mu)^T (\tilde{\Sigma}^{-1} + \Sigma^{-1})(\tilde{\mu} - \mu) \,,
\end{aligned}
$$

本问与之前讲过的 CVPR2018 Semantics-Preserving Adversarial Embedding Networks十分相似





Zero-Shot Visual Recognition using Semantics-Preserving Adversarial Embedding Networks CVPR2018

主要思想：

1、f与f^接近，说明生成的句子在latent space与真实的语句比较接近

2、z重构以后与原来的差距较小，说明这个feature space保留了较多的高层语义信息

Adversarial Feature Matching for Text Generation (PMLR 2017)

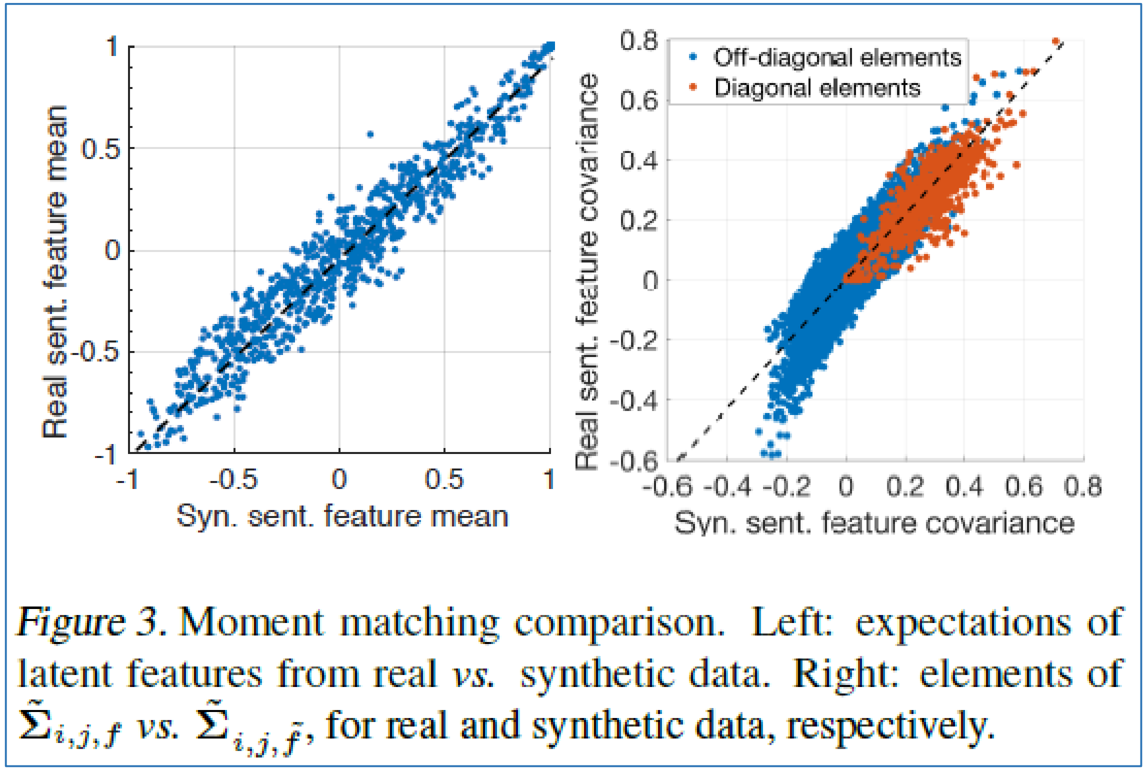
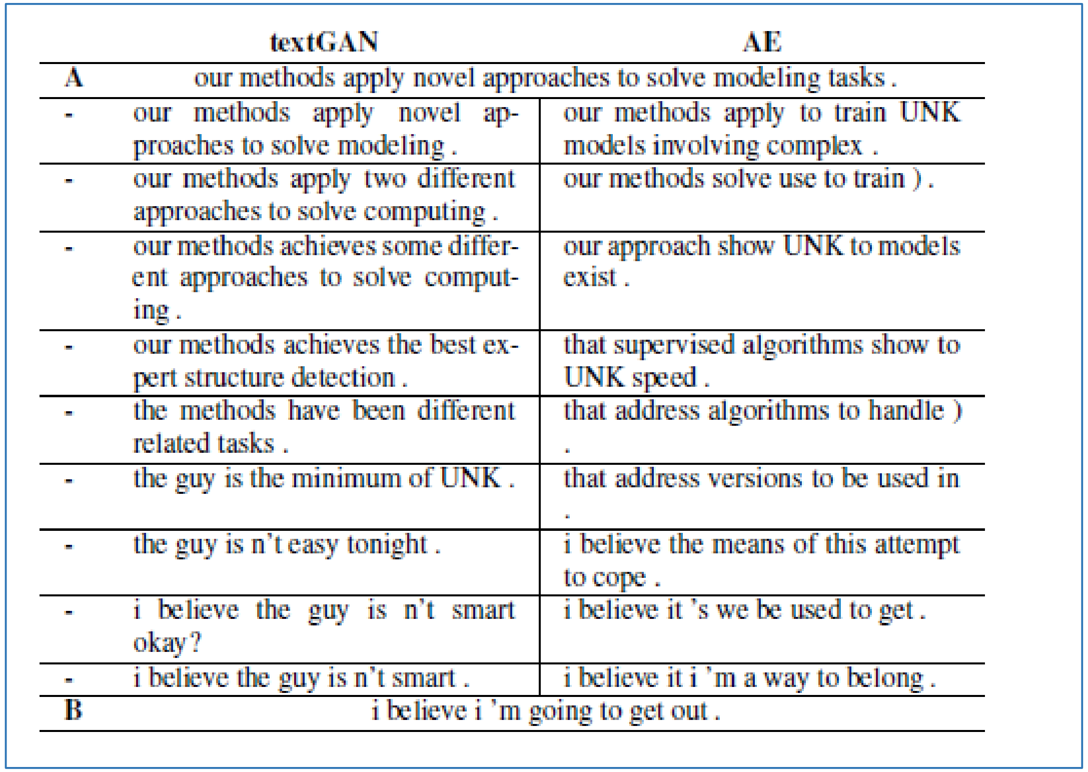


Figure 3. Moment matching comparison. Left: expectations of latent features from real *vs.* synthetic data. Right: elements of $\tilde{\Sigma}_{i,j,f}$ *vs.* $\tilde{\Sigma}_{i,j,\tilde{f}}$, for real and synthetic data, respectively.



	textGAN	AE
A	our methods apply novel approaches to solve modeling tasks .	
-	our methods apply novel approaches to solve modeling .	our methods apply to train UNK models involving complex .
-	our methods apply two different approaches to solve computing .	our methods solve use to train) .
-	our methods achieves some different approaches to solve computing .	our approach show UNK to models exist .
-	our methods achieves the best expert structure detection .	that supervised algorithms show to UNK speed .
-	the methods have been different related tasks .	that address algorithms to handle) .
-	the guy is the minimum of UNK .	that address versions to be used in .
-	the guy is n't easy tonight .	i believe the means of this attempt to cope .
-	i believe the guy is n't smart okay?	i believe it 's we be used to get .
-	i believe the guy is n't smart .	i believe it i 'm a way to belong .
B	i believe i 'm going to get out .	

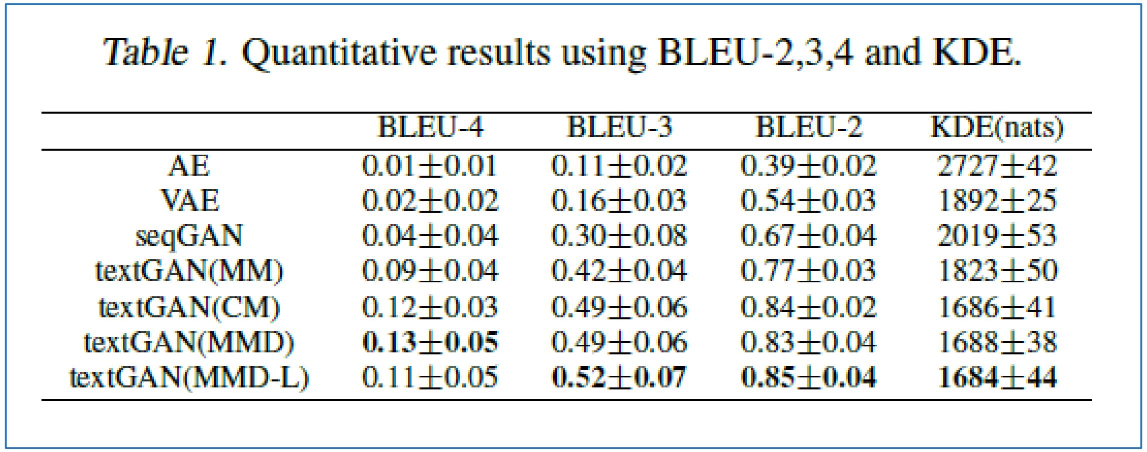


Table 1. Quantitative results using BLEU-2,3,4 and KDE.

	BLEU-4	BLEU-3	BLEU-2	KDE(nats)
AE	0.01±0.01	0.11±0.02	0.39±0.02	2727±42
VAE	0.02±0.02	0.16±0.03	0.54±0.03	1892±25
seqGAN	0.04±0.04	0.30±0.08	0.67±0.04	2019±53
textGAN(MM)	0.09±0.04	0.42±0.04	0.77±0.03	1823±50
textGAN(CM)	0.12±0.03	0.49±0.06	0.84±0.02	1686±41
textGAN(MMD)	**0.13±0.05**	0.49±0.06	0.83±0.04	1688±38
textGAN(MMD-L)	0.11±0.05	**0.52±0.07**	**0.85±0.04**	**1684±44**

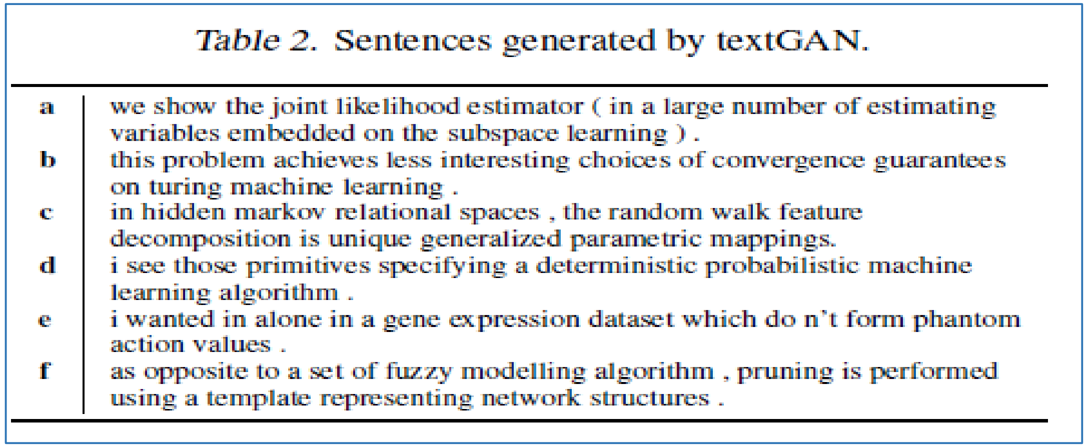


Table 2. Sentences generated by textGAN.

a	we show the joint likelihood estimator (in a large number of estimating variables embedded on the subspace learning) .
b	this problem achieves less interesting choices of convergence guarantees on turing machine learning .
c	in hidden markov relational spaces , the random walk feature decomposition is unique generalized parametric mappings.
d	i see those primitives specifying a deterministic probabilistic machine learning algorithm .
e	i wanted in alone in a gene expression dataset which do n't form phantom action values .
f	as opposite to a set of fuzzy modelling algorithm , pruning is performed using a template representing network structures .

我的感受

1、评测指标

- 指标五花八门，人工评测还是比较重要的一部分。有时候应该想想这些追求指标上的提升会不会让我们陷入文本生成的“局部最优”，而错过了一些全局的东西。就像Yoav Goldberg 前面提到的一样，不能在某些任务的某些指标有些许提升就想着先举旗占坑。

2、语言的复杂性

- 语言的多样性、语法句法的多边性、高层语义空间的特殊性质都是相当复杂的研究领域。
- 每篇论文都提到了句子生成在“全局”范围内的优化，因为我们遣词造句都是整体性的，不是一个字一个字拼起来的，这是我们以后设计模型要考虑的一个方面。

3、GAN for Text generation ?

- 离散型问题在GAN for text generation上是不可避免的。有没有更好的方法?
- 现在有很多研究在改进softmax用于解决argmax导致的不可导问题。比如Gumbel-Softmax。这些trick基于很多概率和统计的知识，而且很有意思，但是推导也比较复杂。大家有兴趣可以看一下Gumbel-Softmax 。
- 训练时的instability和mode collapse问题。还有长文本生成的困难，句子越到后面错误越多。Goldberg批判时也说道，蒙特利尔大学做的都是短文本，平均只有15词，但是wikipedia上句子平均都有17词。

4、RUC智能情报站

- 网上查资料时看到了知乎上的专栏“RUC智能情报站”，是我们学校赵鑫老师实验室做的，也是介绍一些NLP相关的论文，大家有兴趣可以去看一下。

LET’S DO IT!

THANKS !