Towards Generating Coherent and Diverse Text with Deep Generative Models

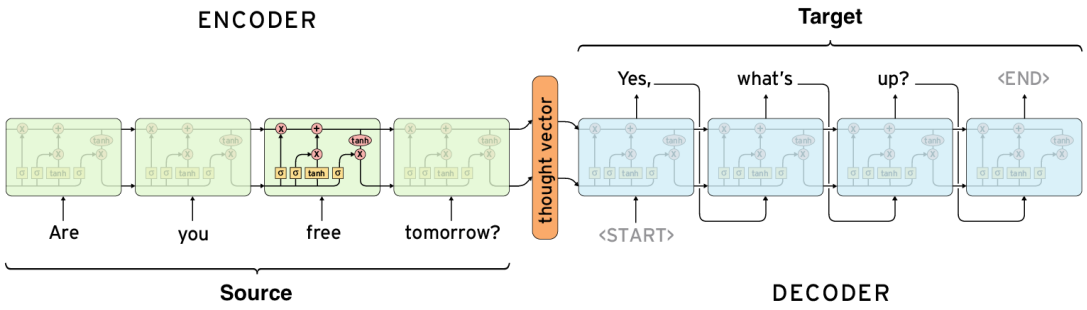
Presenter: Yizhe Zhang

Duke University

October 25, 2017

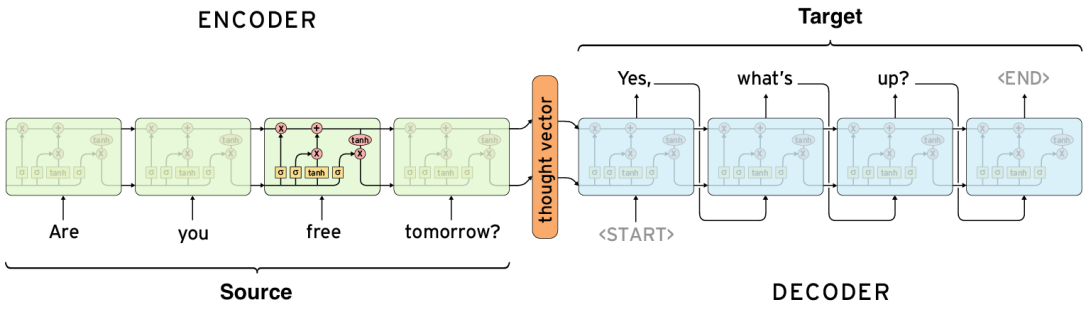
Text generation

- *Human:* ideas → utterance
- *Machine:* latent representation → understandable sentences.
- Language modeling, translation, summarization, conversation models, image/video captioning...
- MLE-based modeling with RNN (esp. LSTM) achieves success.
- Variants: Hierarchical, attentive, etc...



Challenges in text generation with standard model

- MLE-based(e.g. seq2seq) model with RNN has several issues:
  - *Teacher forcing:* the discrepancy between training and testing $\rightarrow$ inconsistency
  - *Exposure bias:* error aggregates with length. $\rightarrow$ generation quality decreases.
  - *MLE-based:* lose distributional information. $\rightarrow$ underestimate diversity and fail to capture multiple modes.



Related works

- Efforts had been made in alleviating above issues
- For *teacher forcing* and *exposure bias* issues
- Scheduled Sampling [Bengio, et al., NIPS 2015]
- Minimum Risk Training [Shen et al., ACL 2016]
- Professor Forcing [Alex et. al, NIPS 2016]
- RL methods:
  - MIXER [Ranzato et al., ICLR 2016]
  - Actor-Critic [Bahdanau et al., ICLR 2017]
  - SeqGAN [Yu et al., AAAI 2017]

Why Deep generative models for text ?

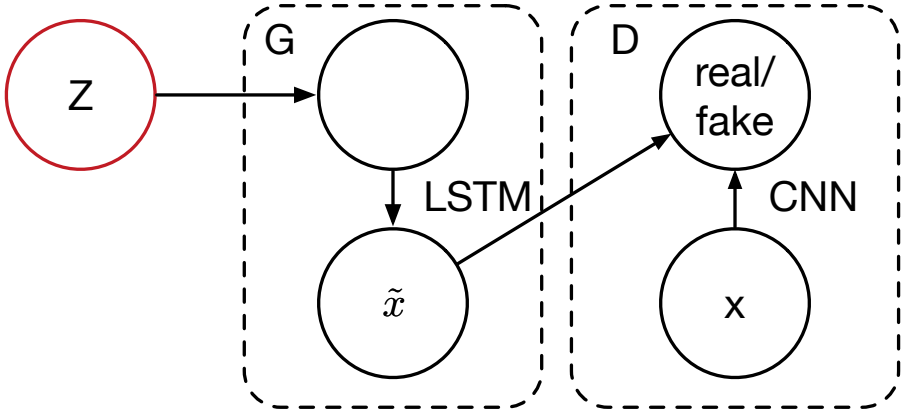
- Deep generative models (DGMs): generative adversarial network (GAN), variational autoencoder (VAE)
- Probabilistic framework account for uncertainty and diversity, natural recipe for generation.
- Ideally, can be leveraged to alleviate *teacher forcing* and *exposure bias* issues.
- Previous works
  - RNN-VAE[Bowman et al., Arxiv 2015]
  - SeqGAN [Yu et al., AAAI 2017]
  - Actor-Critic [Bahdanau et al., ICLR 2017]

Outline

Background

Generative Adversarial Networks

- A game between:
  - Discriminative model D
  - Generative model G
- *G:* trained to maximize the probability of D making a mistake
- *D:* trained to estimate the probability that a sample came from data distribution rather than G



Advantages and challenges

- **Advantages:**
  - *No teacher forcing:* the discriminator provides a consistent sequence-level loss for training and testing.
  - *Less exposure bias:* the discriminator judges the synthetic text evenly from beginning to the end.
  - *Fit density* rather than find MLE
- **Difficulties:** (due to discrete nature of text)
  - Synthetic data is not directly differentiable. $\rightarrow$ *optimizing difficulty*
  - Transitions in text are less smooth than in images. $\rightarrow$ *mode collapsing*.
- **Our approach:** [Zhang et al., ICML 2017]
  - Discretization approximations using *Gumbel-softmax*. (vs RL-based, eg. seqGAN)
  - Ameliorating mode-collapsing issue via *feature moment matching*.

Gumbel-softmax approximation

- We consider a *Gumbel-softmax* approach to approximate a draw from *categorical* distribution.

$$z \sim \text{Cat}(\alpha_1, \cdots, \alpha_k) \Leftrightarrow z = \text{argmax}_i(g_i + log\alpha_i) \quad (1)$$

where $g_i \sim \text{Gumbel}(0, 1)$

$$g_i = -\log(-\log(u_i)), u_i \sim \text{Unif}(0, 1) \quad (2)$$

- Argmax approximation: the $\hat{\boldsymbol{z}} = \text{one\_hot}(z)$ can be approximated by

$$\hat{\boldsymbol{z}}_i = \frac{\exp[(g_i + \log \alpha_i)/\tau]}{\sum_{j=1}^{k} \exp[(g_j + \log \alpha_j)/\tau]} \quad (3)$$

- The output for a softmax-categorical operation is differentiable

Feature moment matching

Overview

- The adversarial game is the following:
- $D(\cdot)$ attempts to select informative sentence features.
- $G(\cdot)$ aims to match these features.
- Features are selected according to **syn/real discrimination ability**, **latent code reconstruction** and **moment matching precision**.

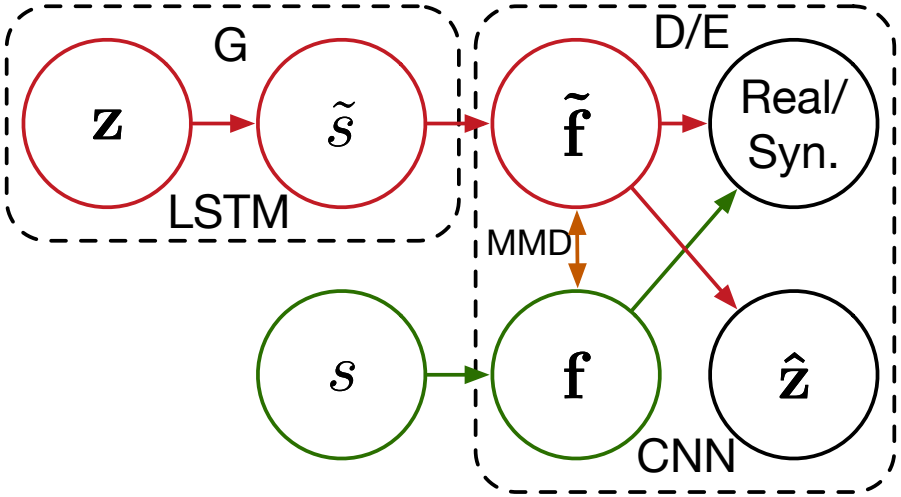


Figure: Model scheme of TextGAN.

Feature moment matching

Generator

- Optimization schemes:

$$\mathcal{L}_G = \mathcal{L}_{MMD^2}$$
$$\mathcal{L}_D = \mathcal{L}_{GAN} + \lambda_r \mathcal{L}_{recon} - \lambda_m \mathcal{L}_{MMD^2}$$

- For G, consider a moment matching loss over *feature vector* using *maximum mean discrepancy* (MMD).

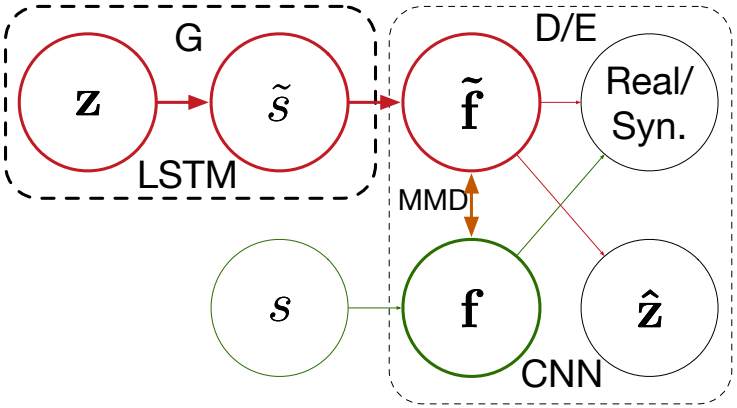
$$\begin{aligned}\mathcal{L}_{MMD^2} &= ||\mathbb{E}_{x\sim\mathcal{X}}\phi(x) - \mathbb{E}_{y\sim\mathcal{Y}}\phi(y)||^2_{\mathcal{H}} \\ &= \mathbb{E}_{x\sim\mathcal{X}}\mathbb{E}_{x'\sim\mathcal{X}}[k(x,x')] \\ &+ \mathbb{E}_{y\sim\mathcal{Y}}\mathbb{E}_{y'\sim\mathcal{Y}}[k(y,y')] - 2\mathbb{E}_{x\sim\mathcal{X}}\mathbb{E}_{y\sim\mathcal{Y}}[k(x,y)].\end{ali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

- With a Gaussian kernel , minimizing the MMD objective $\Leftrightarrow$ minimizing *all order of moments* of two empirical distributions.

Feature moment matching

Generator

- Vanilla GAN: D independently judge each syn/real data.
- The MMD loss for G: match distributions, enforce diversity.
- The gradient signal back-propagated from feature layer is more direct.



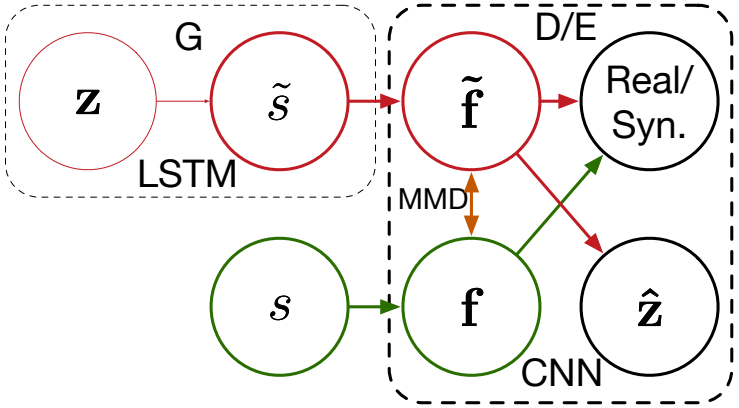
Feature moment matching

Discriminator

- Optimization schemes:

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

- The reconstruction loss in D : select the most *representative* (information-preserving) features.
- The MMD loss in D: select the most *challenging* features.



Framework components

LSTM generator

- We specify an LSTM generator to translate a *latent code vector*, z, into a *synthetic sentence* $\tilde{s}$.
- All other words in the sentence are sequentially generated using the RNN, *based on previously generated words*, until the end-sentence symbol is generated.

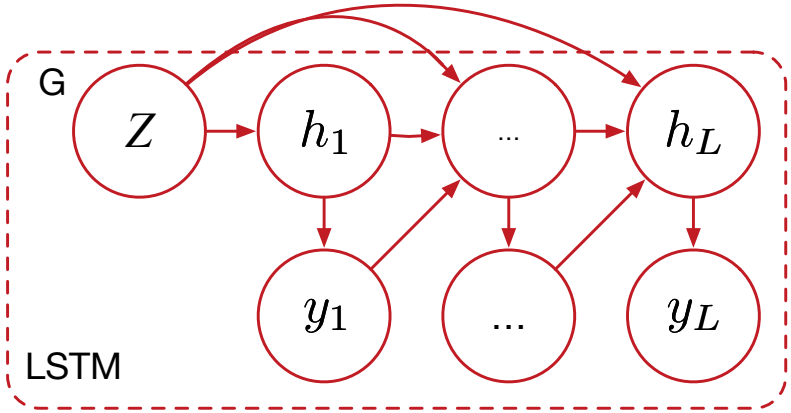


Figure: LSTM generator

Framework components

CNN discriminator

- CNNs weight each word equally and are empirically better at abstracting features particularly with long sentences.
- A sentence is represented as a matrix $\mathbf{X} \in \mathbb{R}^{\mathbf{k} \times \mathbf{T}}$, followed by a convolution operation.
- A max-over-time pooling operation is then applied.

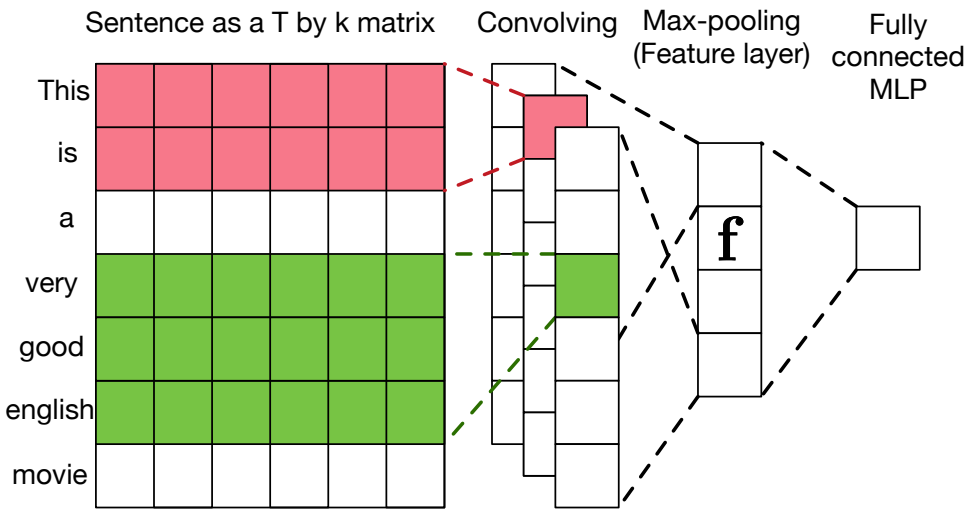


Figure: CNN discriminator

Empirical evaluation

- **Dataset:** 0.5M Arxiv sentences + 0.5M BookCorpus sentences .
- **Evaluation:** Kernel density estimation (KDE) .
- **Evaluation:** Corpus-level BLEU score .
- **Comparison:** baseline auto-encoder, variational auto-encoder and seqGAN [Yu et al. 2016]

Experimental Result

Generated text

- Produce novel phrases. (b)
- The synthetic sentences seem syntactically reasonable.
- The semantic meaning is less preserved. (e)

Table: 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 .

Experimental Result

Moment Matching

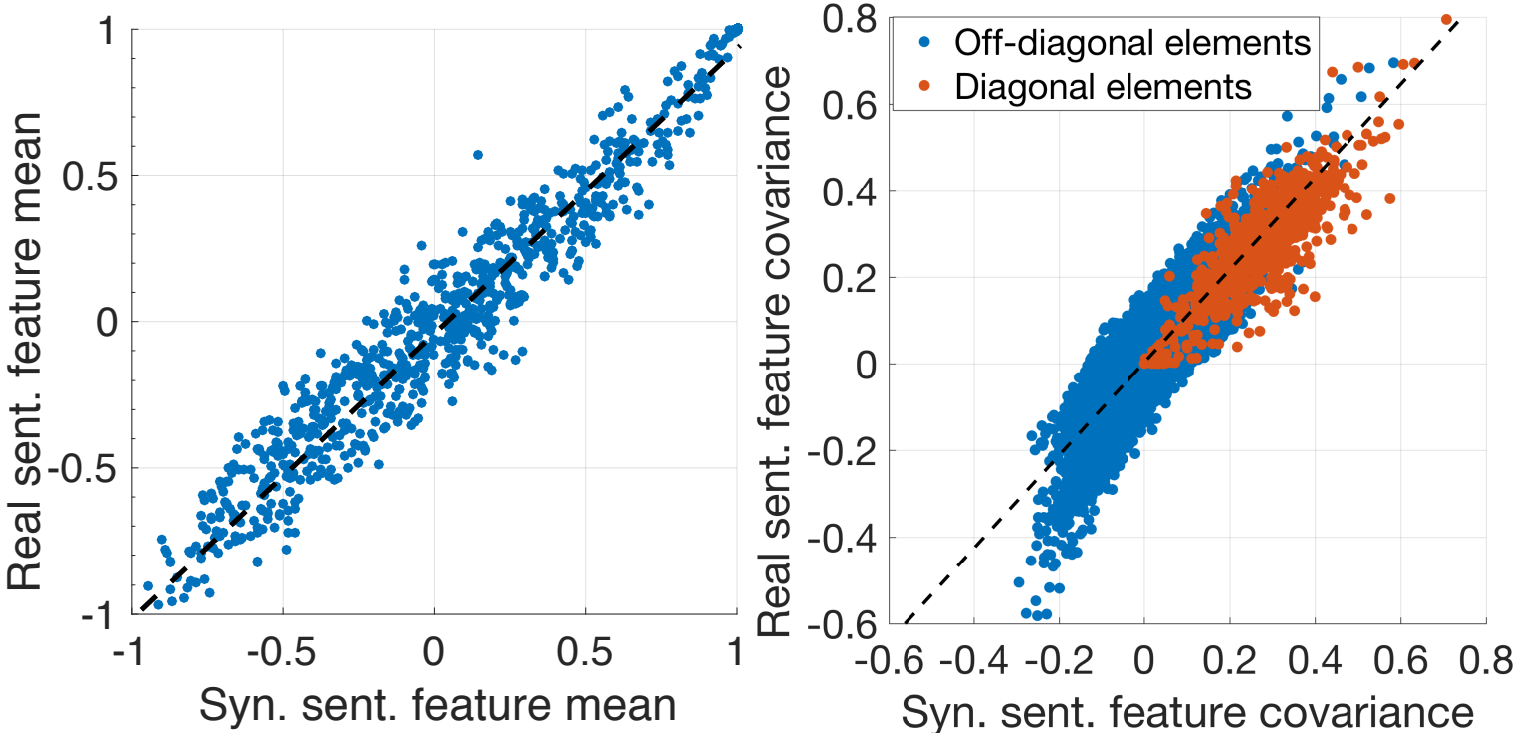


Figure: Moment matching comparison. Left: expectations of latent features from real *vs.* synthetic data. Right: elements of covariance matrix for real and synthetic data, respectively.

Experimental Result

Sentence transition

Table: Intermediate sentences by linear transition.

	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 .	

Experimental Result

Quantitative evaluation

- Higher BLEU, lower KDE is better.

Table: 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	0.12±0.06	0.40±0.06	0.61±0.07	2025±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*

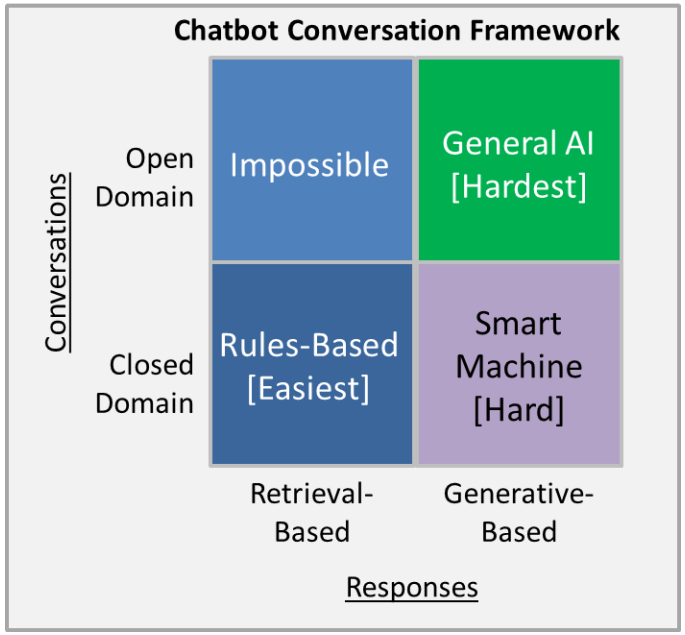
Summary

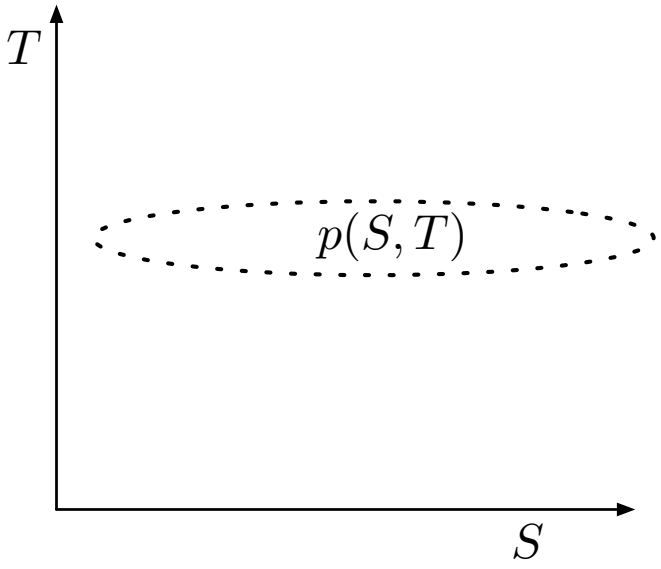
- A framework for text generation using adversarial training
- Techniques to alleviate practical issues when training GAN on text domain.
- Future works:
  - Variance reduction for gradient estimation.
  - Better approach for reducing exposure bias
- code available at: https://github.com/dreasysnail/textGAN_public

Outline

Open-domain generative neural converation model

- Single turn conversation. S: source prompt, T: target response
- Seq2seq MLE: $\hat{T} = \text{argmax}_T \log p(T|S)$
- **Issue:** bland and not diverse responses: `I do n't know`





Previous methods

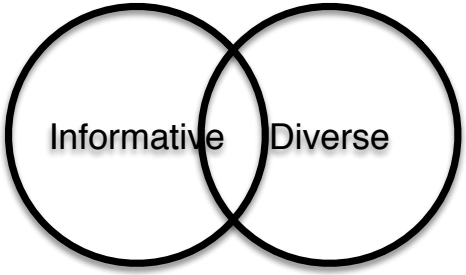
- Maximum mutual information (MMI) [Li, et al., NAACL 2016]:

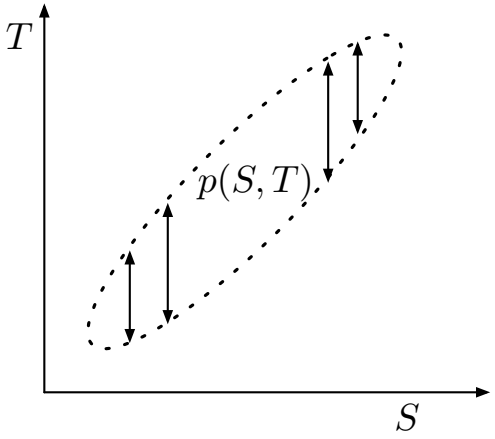
$$\hat{T} = \text{argmax}_T \log p(T|S) - \lambda \log p(T)$$

- **Issue:** 1) hyperparameter makes it less principled. 2) training/testing discrepancy

Diversity and blandness

- *Diversity*: variance of $p(T|S)$ with fixed S
- *Blandness*: variation of $p(T|S)$ w.r.t. S
- Diverse but bland:
  `I do n't know; Not sure; We'll see; I agree`
- Informative but monotonous: deterministic response





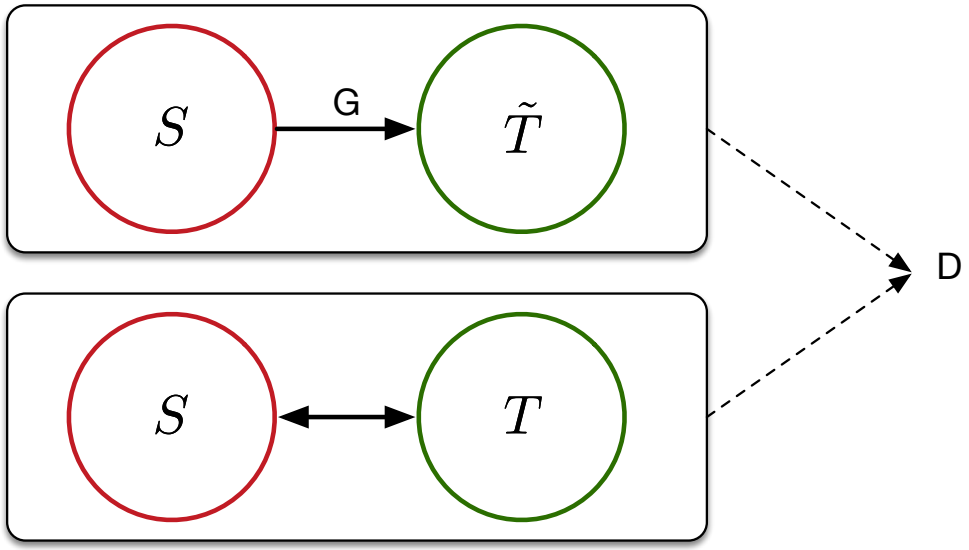
Our aim

- Improve *diversity* and reduce *blandness*
- *diversity*: use adversarial training to fit conditional distribution $P(T|S)$.
- *blandness*: use variational information maximization (VIM) to generate informative response.

Conditional GAN

Objective

- Using conditional GAN:
  1. Rather than model $\text{argmax}_T p(T|S)$, consider $p(T|S)$.
  2. Sequence-level loss, alleviating exposure bias.



Conditional GAN

Generator

- $(S, z) \mapsto \hat{T}$, CNN-LSTM seq2seq model.
- *Gumbel-softmax* instead of policy gradient.
- Truncated back-propagation.

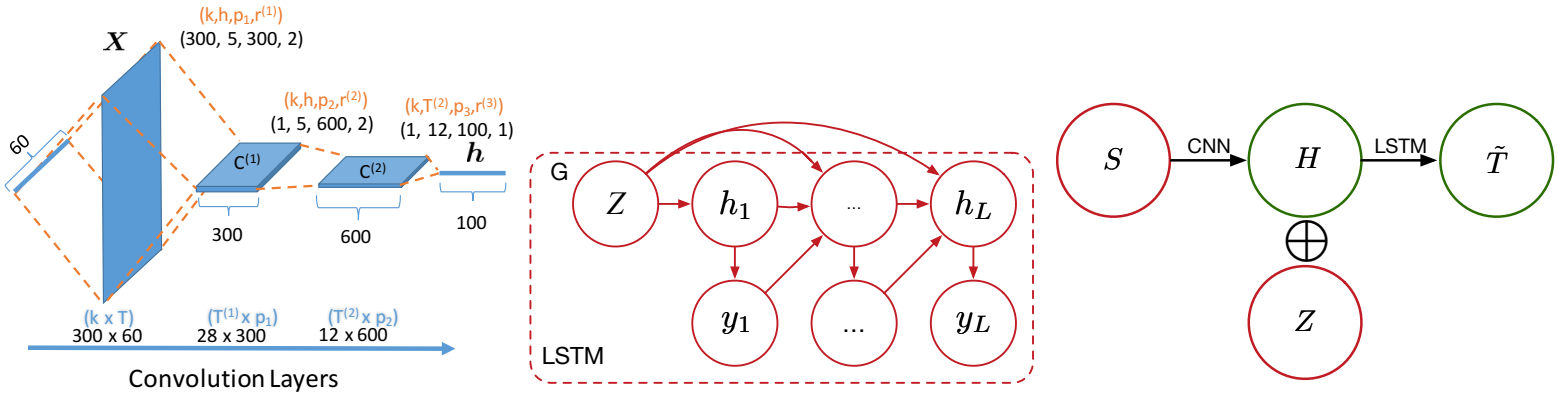


Figure: CNN-LSTM generator

Conditional GAN

Discriminator

$$\textit{Sent.feature}: E_s \triangleq E_{w_s}(S), E_t \triangleq E_{w_t}(T)$$

$$\textit{Discr.output}: D((T,S)) = D_{cos}(E_{w_s}(S), E_{w_t}(T))$$

- Inspired by DSSM [Huang et al., CIKM 2013]

$$\mathcal{L}_G = \mathbb{E}_{(T,\tilde{T},S)\sim p,p^e} f'(D((T,S)) - D((\tilde{T},S))) \ , \ \mathcal{L}_D = -\mathcal{L}_G$$

Where $f' : [-2,2] \mapsto \mathbb{R}$

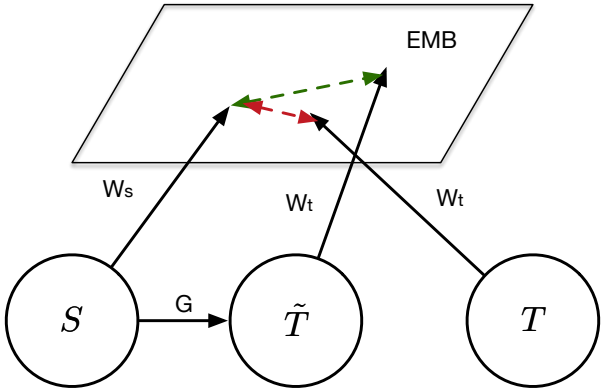
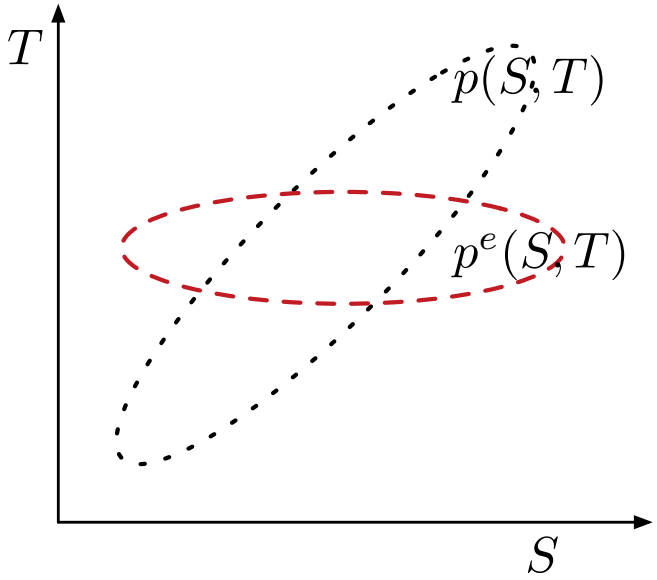


Figure: Discriminator

Blandness problem

- *Oracle joint distr.*: $p(T, S)$
- *Encoder joint distr.*: $p_\theta^e(T, S) = p_\theta^e(T|S)p(S)$
- Our aim: learn θ, such that $p_\theta^e(T, S)$ approximate $p(T, S)$
- **Problem:** learned $p_\theta^e(T|S)$ tends to be similar with different S.



Maximize mutual information

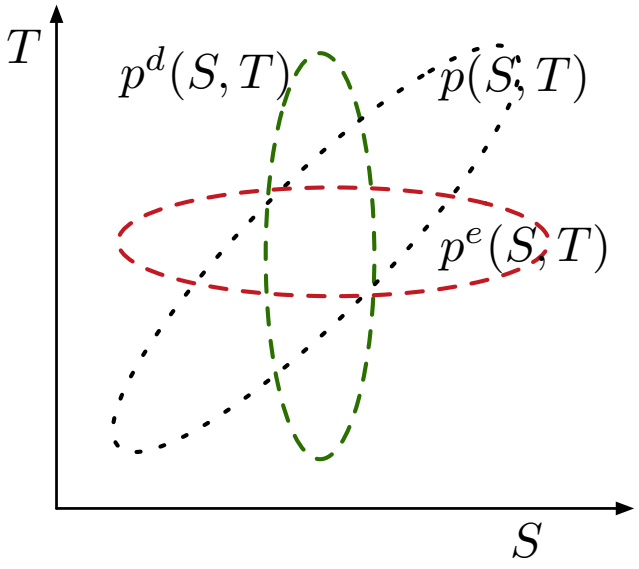
- **Solution:** explicitly maximize mutual information $I_{p^e}(S,T)$ over p^e.
- $I_{p^e}(S,T)$ is hard to directly optimize
- Use *variational information maximization* (VIM):
- A lower bound of $I_{p^e}(S,T)$

$$\begin{aligned} I_{p^e}(S,T) &= \mathbb{E}_{p^e(S,T)} \log \frac{p^e(S,T)}{p(S)p(T)} \\ &= -\mathbb{E}_{p(S)} \log p(S) + \mathbb{E}_{p(T)} \mathbb{E}_{p^e(S|T)} \log p^e(S|T) \\ &\geq \mathbb{E}_{p(S)} \mathbb{E}_{p_\theta(T|S)} \log q_\phi(S|T) + \text{Constant} \end{aligned} \tag{5}$$

- Different from Li et al. 2015, where forward and backward models are separable.

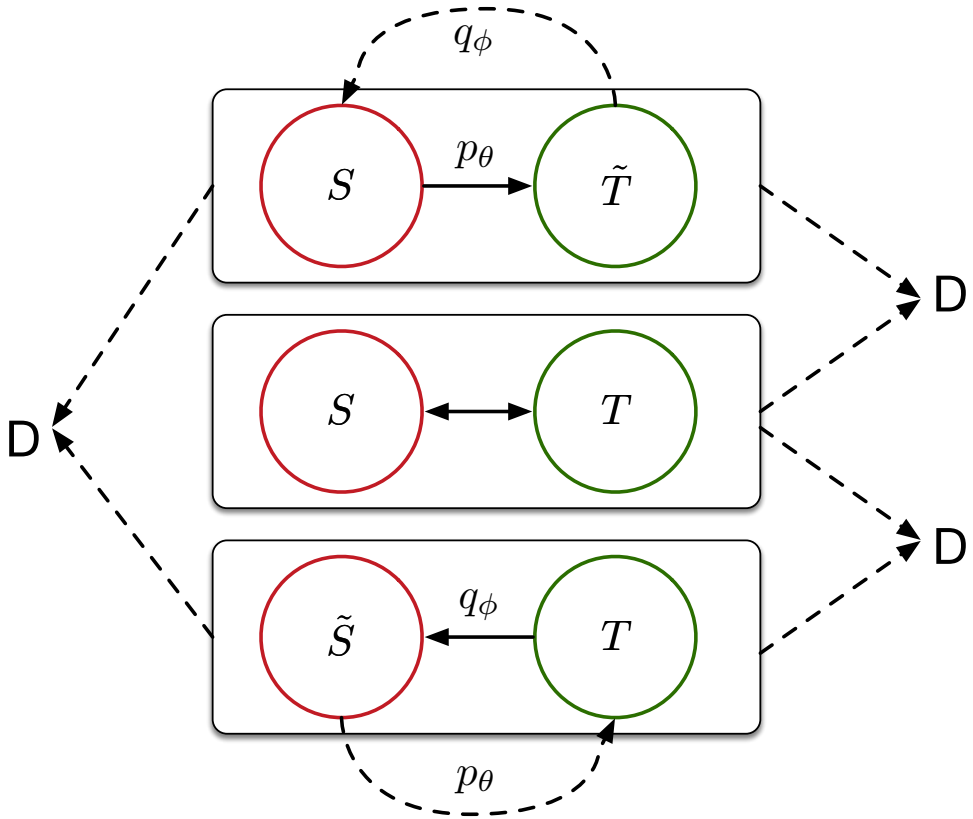
Learn a better joint distribution

- $q_\phi(S|T)$ may also tend to be bland in generating source.



- Forward and backward model are *symmetric* and *collaborative*.
- What about make the objective symmetric?
- *Decoder joint distr.*: $p^d_\theta(T,S) = q^d_\phi(S|T)p(T)$

Learn a better joint distribution (Cont'd)



- Augmented paired data for discriminator training
- Forward and backward model works in a synergistic manner

Final Objective

- Joint train a supervised loss, similar to Xia et. al, ICML 2017
- Final objective:

$$
\begin{aligned}
&\text{argmin}_{\psi}\text{argmax}_{\theta,\phi}\mathcal{L} \\
&= -\mathbb{E}_{(T,\tilde{T},S)\sim p,p_{\theta}^{e}} f'(D_{\psi}((T,S)) - D_{\psi}((\tilde{T},S))) \\
&- \mathbb{E}_{(T,\tilde{S},S)\sim p,p_{\phi}^{d}} f'(D_{\psi}((T,S)) - D_{\psi}((T,\tilde{S}))) \\
&+\lambda\mathbb{E}_{p(S)}\mathbb{E}_{p_{\theta}(T|S)} \log q_{\phi}(S|T) \\
&+\lambda\mathbb{E}_{p(T)}\mathbb{E}_{q_{\phi}(S|T)} \log p_{\theta}(T|S)
\end{aligned}
\tag{6}
$$

where λ is a balance parameter.

Experiments

- Reddit 2M dataset, src-tgt pairs.
- Maximum length 53.
- Use BLEU-4, entropy-4, distinct-1,2 for relevancy and diversity evaluation.
- Distinct: (# of unique token)/(# of total token)
- Entropy:

$$E = -\frac{1}{T}\sum_{w} F(w)[\log F(w) - \log T] \tag{7}$$

$$T = \sum_{w} F(w) \tag{8}$$

where $F(w)$ denote the occurence of token w.

Experimental Result

Quantitative evaluation

- Higher BLEU, entropy and distinct value is better.

Table: Quantitative results

	BLEU-4	Dist-1	Dist-2	Ent-4
Seq2seq	1.77	0.005	0.041	7.421
CondGAN	1.85	0.008	0.072	9.679
Dual-GAN+VIM	**2.32**	**0.009**	**0.093**	**10.213**

Experimental Result

Generated responses

- Random questions

Source:	How do you think of seattle ?
Seq2seq:	i was n't thinking about it .
CondGAN:	i think it 's not that i want to know .
Bidir-GAN	it 's a great place, i 'd say to visit !

- CondGAN

Source:	which country do you want to travel ?
Seq2seq:	american , but i 'm not sure what the country is .
CondGAN:	i 'm not sure i know of any country , but i 'm just saying what the other is for .
Bidir-GAN	i think it 's called " south africa ".

Summary

- Summary
  - Adversarial seq2seq training scheme with specially designed sequence-level discrimination loss.
  - Simultaneously learning forward and backward models, which enhance each other.
- Future work:
  - Theoretical perspective for the adversarial loss.
  - Stabilize GAN training.

Outline

Fully convolutional text autoencoder

- *RNN as generator*: sequential generation process. → exposure bias, even with GAN...
- Different decoder/generator?
- Recent work: DCNN+RNN [Semeniuta, et al., 2017, arxiv], PixelCNN [Yang, et al., 2017, arxiv] → *still sequential model*
- *Purely convolutional/transpose-convolutional framework* for decoder/generator. [Zhang et al., NIPS 2017]

Fully convolutional text autoencoder

- Consider a text-autoencoder with CNN as encoder and deconvolutional network (DCNN) as decoder

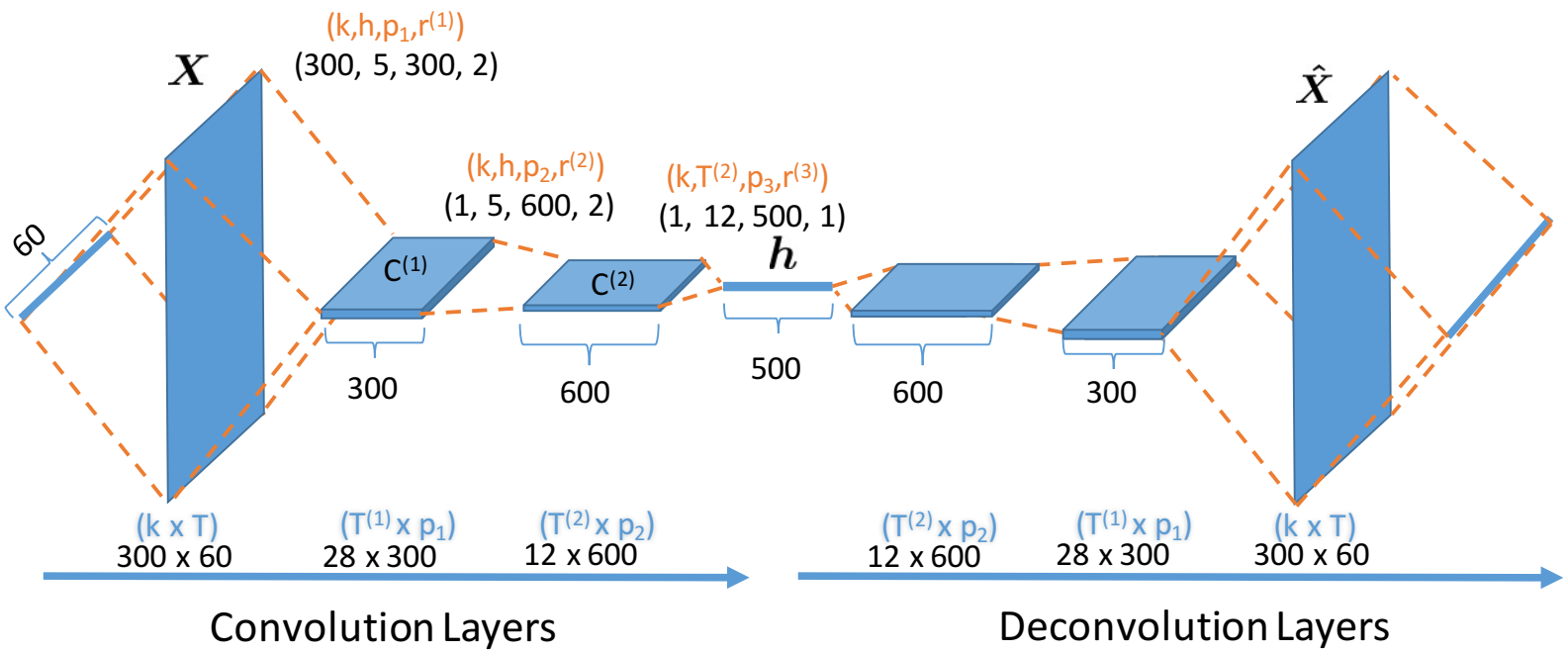


Figure: Framework

Model details

- *Input* sentence of length T: $\mathbf{X} \in \mathbb{R}^{\mathbf{k} \times \mathbf{T}}$
- *Embedding matrix*: $\mathbf{W_e} \in \mathbb{R}^{\mathbf{k} \times \mathbf{V}}$.
- The *CNN* encodes sentence s into a vector $\boldsymbol{h}$

$$\boldsymbol{h} = \text{multi-layer-conv}(\mathbf{X}) \tag{9}$$

- The *DCNN* decodes $\boldsymbol{h}$ into $\hat{\mathbf{Y}}$

$$\hat{\mathbf{X}} = \text{multi-layer-deconv}(\boldsymbol{h}) \tag{10}$$

- Let $\hat{w}^t$ denote the t-th word in generated sentence $\hat{s}$, the probability of $\hat{w}^t$ to be a specific word v is

$$p(\hat{w}^t = v) = \frac{\exp[D_{cos}(\hat{\boldsymbol{x}}^t, \mathbf{W_e^v})/\tau]}{\sum_{v' \in V} \exp[D_{cos}(\hat{\boldsymbol{x}}^t, \mathbf{W_e^{v'}})/\tau]} \tag{11}$$

$D_{cos}(\boldsymbol{x}, \boldsymbol{y})$ denotes cosine similarity, τ is a temperature parameter ($\tau = 0.01$).

DCNN as a decoder/generator

- Consistent training and testing → *free of teacher forcing*
- Without a pre-specified ordering structure → *free of exposure bias*
- *Fast* and easy to be *parallelized*.
- *Weaker* dependency, less order-preserving, better for higher mutual information source-target pairs.
- scenarios for applying DCNN decoder rather than LSTM decoder:
  Reconstruction > *summarization* > *translation* > *conversation*

Reconstruction

Ground-truth:	on every visit to nyc , the hotel beacon is the place we love to stay . so conveniently located to central park , lincoln center and great local restaurants . the rooms are lovely . beds so comfortable , a great little kitchen and new wizz bang coffee maker . the staff are so accommodating and just love walking across the street to the fairway supermarket with every imaginable goodies to eat .
Hier. LSTM [Li et al, 2015]	every time in new york , lighthouse hotel is our favorite place to stay . very convenient , central park , lincoln center , and great restaurants . the room is wonderful , very comfortable bed , a kitchenette and a large explosion of coffee maker . the staff is so inclusive , just across the street to walk to the supermarket channel love with all kinds of what to eat .
Our CNN-DCNN	on every visit to nyc , the hotel beacon is the place we love to stay . so closely located to central park , lincoln center and great local restaurants . biggest rooms are lovely . beds so comfortable , a great little kitchen and new UNK suggestion coffee maker . the staff turned so accommodating and just love walking across the street to former fairway supermarket with every food taxes to eat .

Table: Reconstructed paragraph of the Hotel Reviews example

DCNN as a decoder for reconstruction task

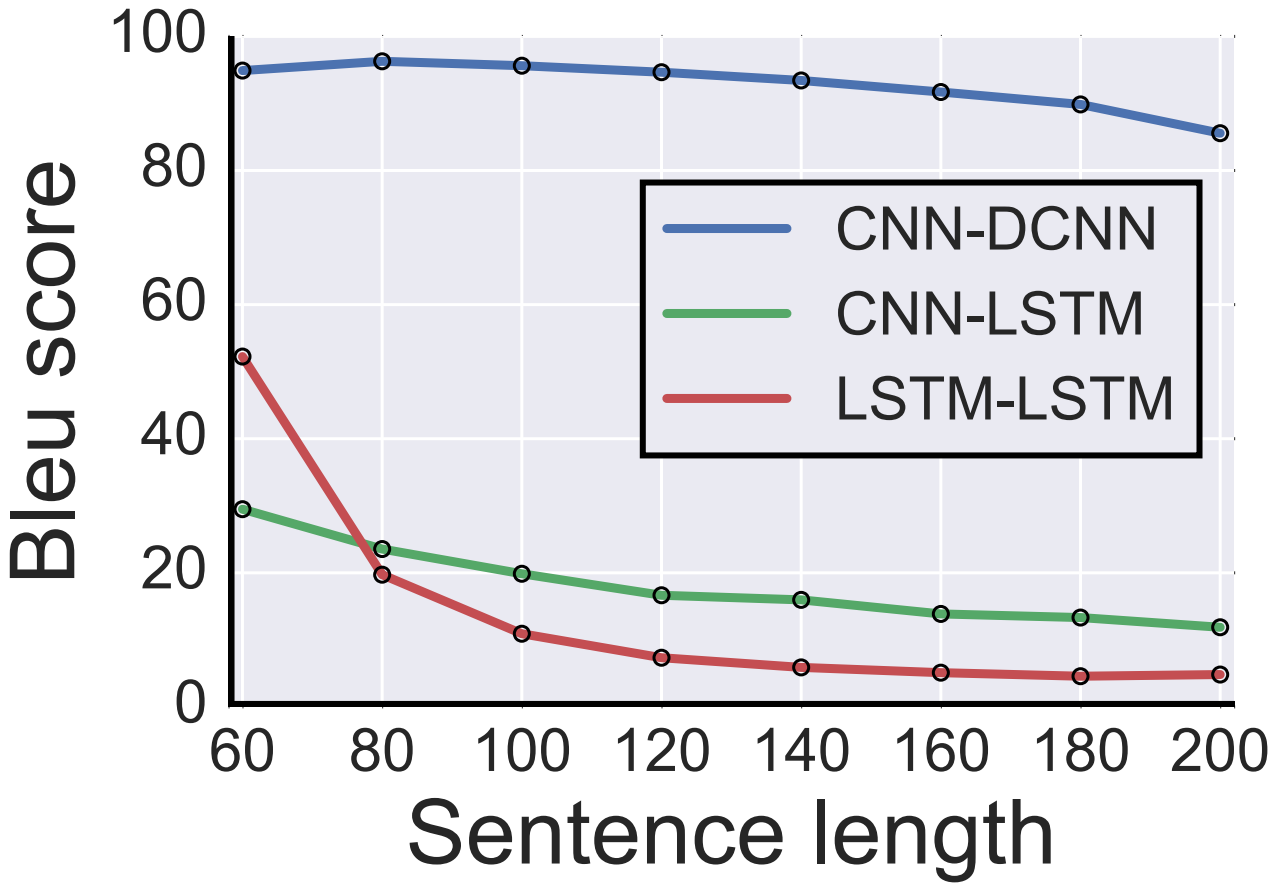
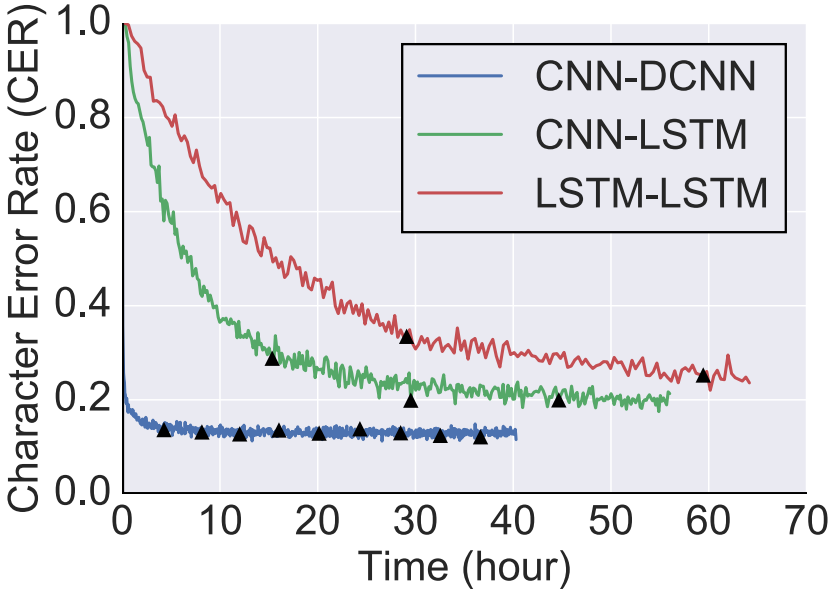


Figure: BLEU score *vs.* sentence length for Hotel Review data.

DCNN as a decoder for spelling correction task



Original	what s your idea of a stepping stone to better things to come ?
Modified	wuat s yogr idem of t stepukng jtzne ti better thingz tt coee ?
ActorCritic	what s your idem of t stepuang jokne ti better thing itt come ?
LSTM-LSTM	what s your idea of a speaking stand to better things to come ?
CNN-LSTM	what s your idem of a stepping start to better thing to come ?
CNN-DCNN	what s your idea of a stepping stone to better things to come ?

Figure: Spelling error denoising comparison.

DCNN as a decoder for title prediction task

Model	CNN-LSTM	CNN-LSTM w/ rec	CNN-DCNN	CNN-DCNN w/ rec
Rouge-L	16.37	**18.14**	14.75	16.83

Table: *arXiv* data. w/ rec: with DCNN reconstruction

Abstract:	this paper presents a new state of the art for document image classification and retrieval , using features learned by deep convolutional neural networks (cnns) . in object and scene analysis , deep neural nets are capable of learning a hierarchical chain of abstraction. experiments show that features extracted from cnns are robust to compression , cnns trained on non document images transfer well to document analysis tasks , and enforcing region specific feature learning is unnecessary given sufficient training data . this work also makes available a new labelled subset of the collection , useful for training new cnns for document analysis .
Ground-truth	evaluation of deep convolutional nets for document image classification and retrieval
CNN-LSTM	deep learning for image recognition
CNN-LSTM w/ r	a classification algorithm for deep neural networks
CNN-DCNN	deep neural convolutional for for recognition
CNN-DCNN w/ r	residual and based neural based classification for recognition net-

Outline

Deconvolutional text VAE (on-going work)

- Leveraging variational auto-encoder to generate text continuously.
- For an input x, the distribution of latent code z is modelled as $q_\phi(z|x)$:

$$\mu = g_1(f^{cnn}(x;\phi_1);\phi_1), \log\sigma = \mu = g_2(f^{cnn}(x;\phi_2);\phi_2)$$
$$z \sim \mathcal{N}(\mu,\sigma)$$

- The generative model $p_\theta(x|z)$ is modelled as deconvolutional network.
- Variational lower bound

$$\mathcal{L} = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p(z)) \tag{12}$$

Discussion

- Improves representation learning and downstream tasks

Figure: t-SNE embeddings (left: DCNN-VAE, right: LSTM-VAE) for BookCorpus and arXiv sentences, colored as orange and blue.

- Generation is less coherent comparing with LSTM.
- Have the best of both worlds to generate long sentences?
- *Globally* DCNN + *locally* RNN?

Summary

- *A deconvolutional decoder*: performance is irrespective of length
- Significant computational savings.
- Better at representation learning.
- Still experimental for generation task.
- Future directions:
  - Could it be improved for language modeling/generation task by incorporating with RNN?
  - Following up, could it be used in GAN setup to avoid exposure bias issue?
- Code availability: https://github.com/dreasysnail/textCNN_public

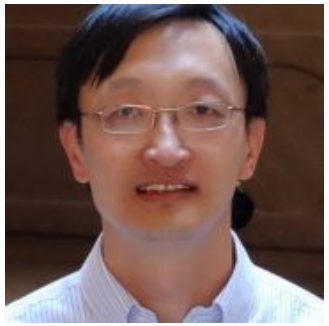
Conclusion

- Towards generating *consistent* and *diverse* text and alleviating issues of traditional RNN-based model, by leveraging DGMs.
- Text and conversion generation via *adversarial training*
- *Deconvolutional decoder* and its potential applications.

Acknowledgment

Lawrence Carin
(Duke)

Michel Galley
(MSR)



Jianfeng Gao
(MSR)

Bill Dolan
(MSR)

My collaborators: Zhe Gan, Kai Fan, Dinghan Shen, Ricardo Henao, ...

Thank You!