Text Generation Models with Auxiliary Objective

Bill Yang, Lucas Kabela

September 22, 2020

Overview

❖ Auxiliary objectives == supplemental, often helpful in nature

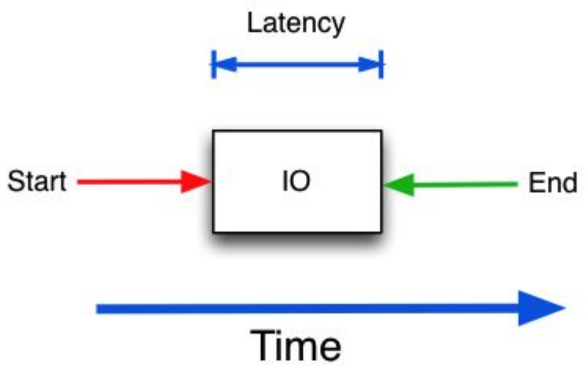
❖ Examples today:

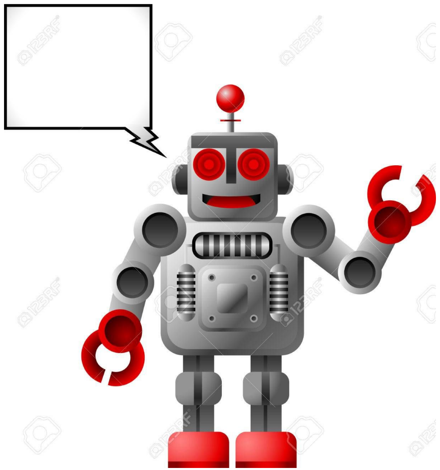
  ○ Latency

  ○ Controllability based on an attribute

❖ Other examples:

  ○ Predicting next word in NER





STACL: Simultaneous Translation with Implicit Anticipation and Controllable Latency using Prefix-to-Prefix Framework

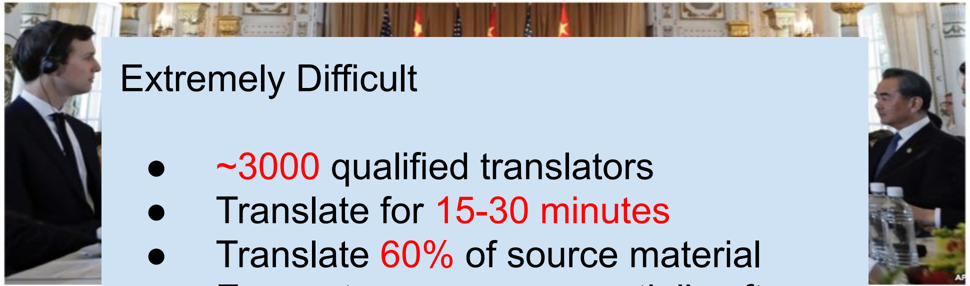
Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, Haifeng Wang

ACL 2019

Consecutive vs. Simultaneous Interpretation

consecutive interpretation
multiplicative latency (x2)

simultaneous interpretation
additive latency (+3 secs)



Extremely Difficult

- ~3000 qualified translators
- Translate for 15-30 minutes
- Translate 60% of source material
- Error rates grow exponentially after a few minutes

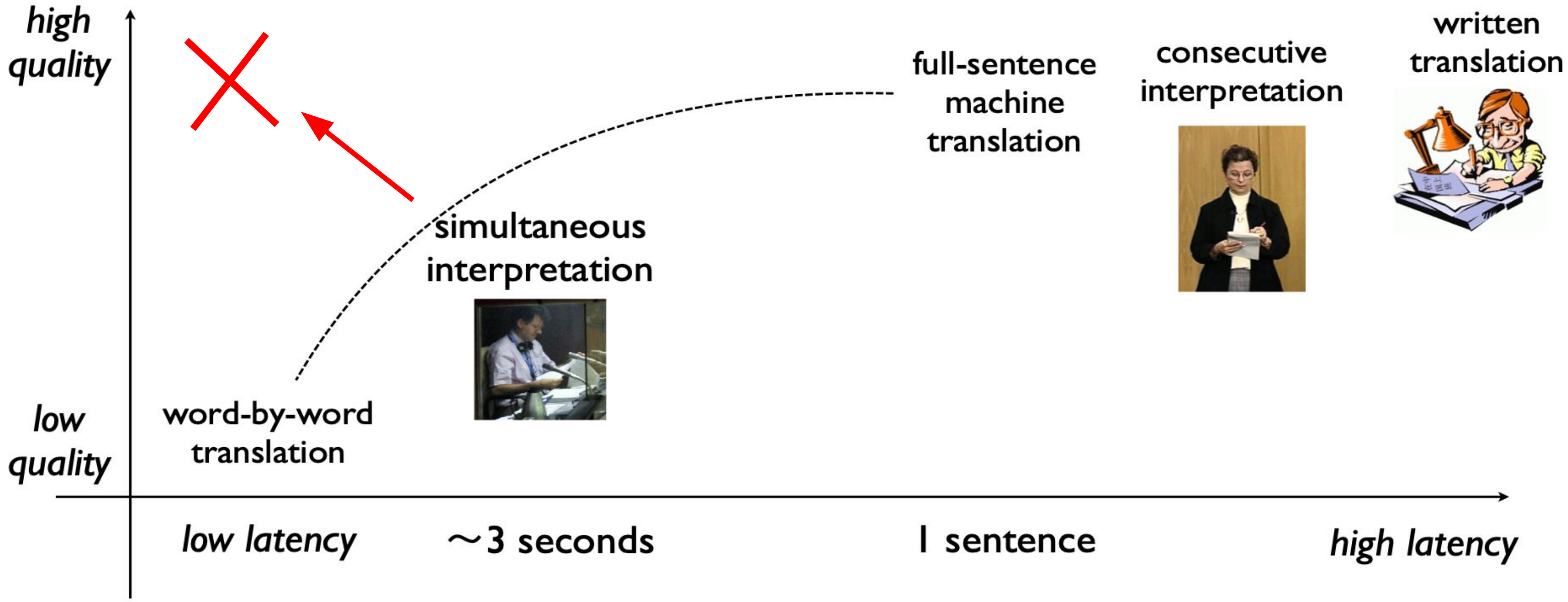
From Huang Liang's [presentation]

Difficulties

- Anticipation (Word Order), Omission, Paraphrasing, Summarization, etc.

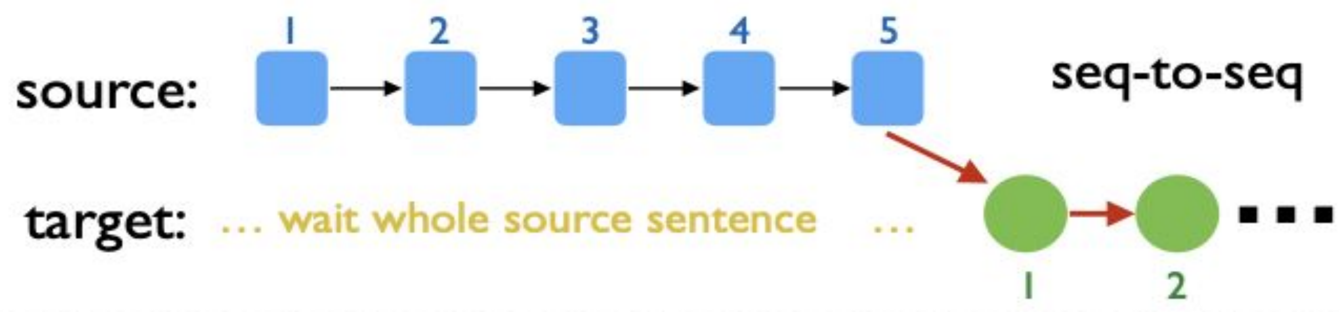
From Huang Liang's [presentation]

Tradeoff between Latency and Quality



From Huang Liang's [presentation]

Full-Sentence Machine Translation



Segment Translation

- Decide when/how to segment a sentence
- Translate sentences segments

Segment Translation

- Simple, Lexicalized Choice of Translation Timing for Simultaneous Speech Translation (Fujita et al. 2013)
  - Decides segments based on phrase table
  - Uses RP for phrase reordering

Table 1: *Phrase table and right probability (RP)*

Source	Target	RP
watashi	I	0.8
watashi ha	I	0.9
otoko	man	0.2
otoko desu	am a man	0.6

Table 2: *Segmentation result*

Unit	Result
watashi ha	I
otoko desu	am a man

- Optimizing Segmentation Strategies for Simultaneous Speech Translation (Oda et al. 2014)
  - Segmentation Model + Greedy DP Search

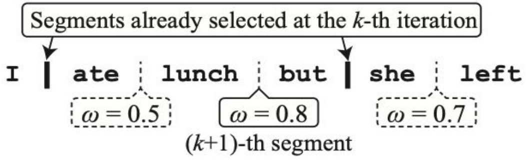


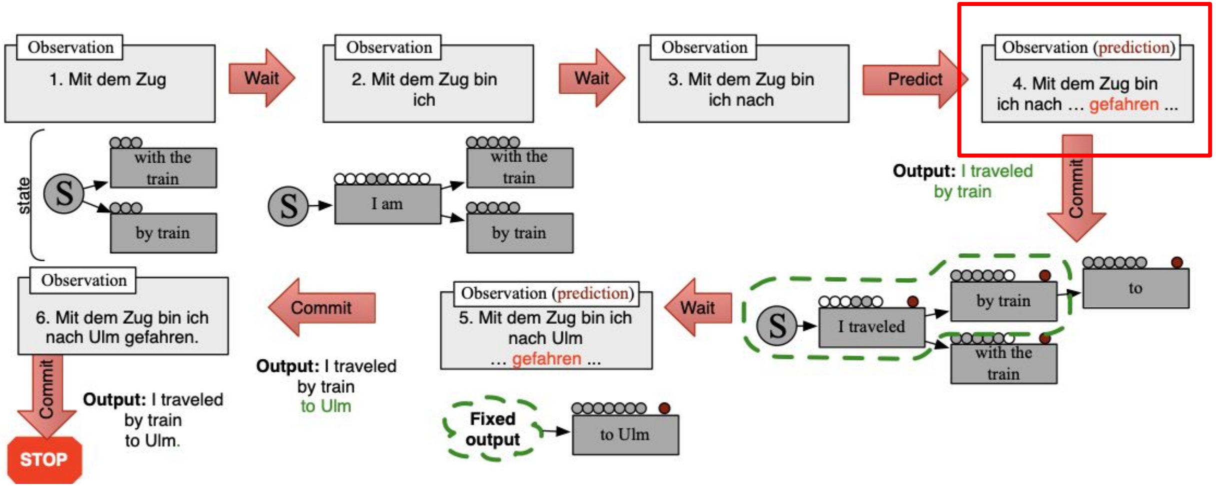
Figure 2: Example of greedy search.

Prediction/Anticipation

- Predict or anticipate future words in the sentence

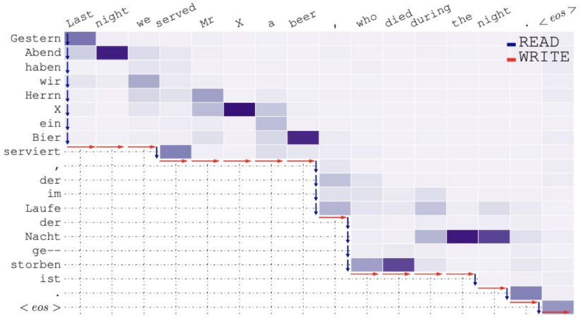
- Don't Until the Final Verb Wait: Reinforcement Learning for Simultaneous Machine Translation (Grissom II et al., 2014)

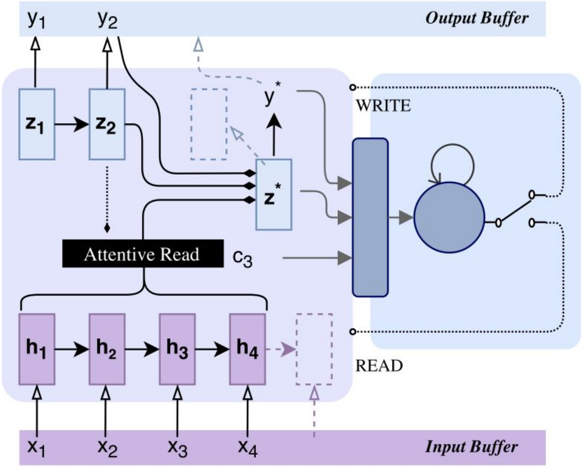
  Predicts source word



Reading + Writing

- Can neural machine translation do simultaneous translation? (Cho et al. 2016)
  - Introduces the notion of Wait Criteria
- Learning to Translate in Real-time with Neural Machine Translation (Gu et al. 2016)
  - Uses RL to learn Read/Write actions



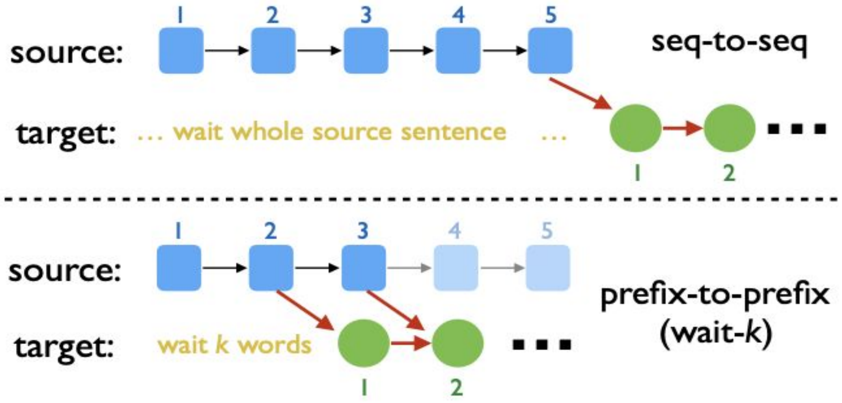


Drawbacks

- Can only "encourage" latency, not control latency
- RL is complicated and slow to train
- Use base models trained on full sentences

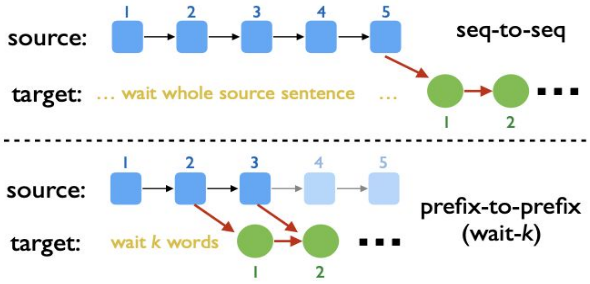
Contributions

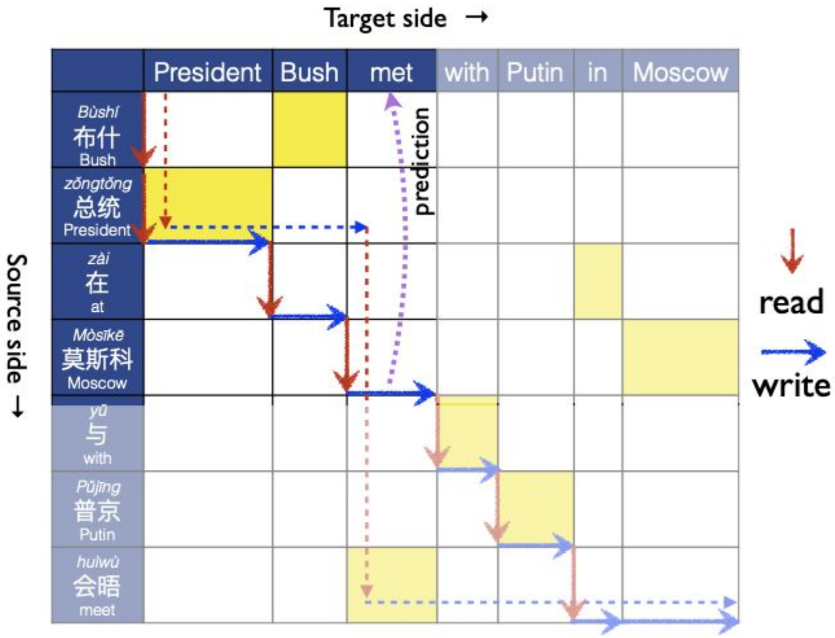
- Prefix-to-prefix model
  - Achieves arbitrary fixed latency
  - Does not use full sentence models
  - Implicitly anticipates future words
- Average Latency Metric
  - Better metric for measuring source word latency in simultaneous MT



Prefix-to-Prefix Model

1. Read up to *g(x)* words
2. Write a word
   a. If all source words are read
      use beam search
   b. Otherwise greedily choose





Prefix-to-Prefix Model (Cont)

Cut-off step

$$\tau_g(|\mathbf{x}|) = \min\{t \mid g(t) = |\mathbf{x}|\}$$

Monotonic, non-decreasing “wait” function

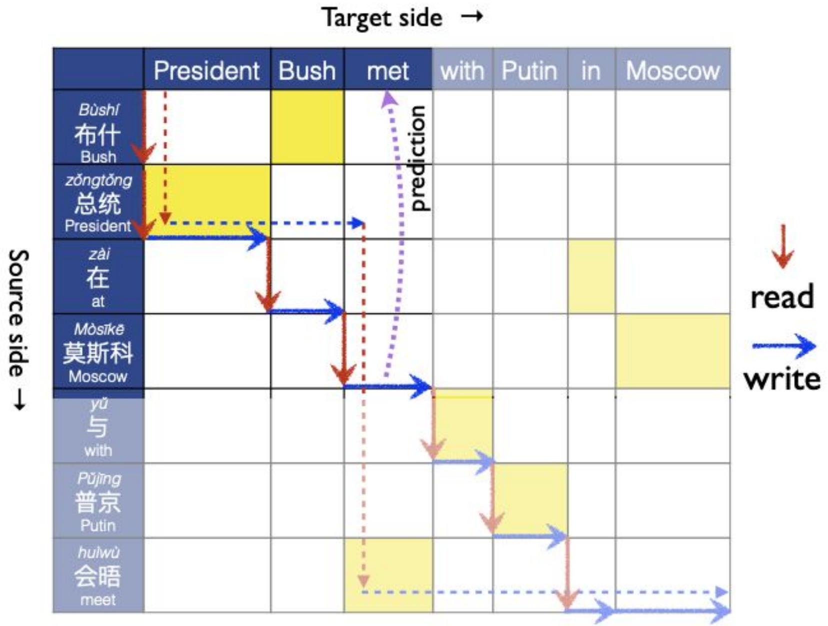
$$g_{\text{wait-}k}(t) = \min\{k + t - 1,\ |\mathbf{x}|\}$$

Probability

$$p_g(\mathbf{y} \mid \mathbf{x}) = \prod_{t=1}^{|\mathbf{y}|} p(y_t \mid \mathbf{x}_{\leq g(t)}, \mathbf{y}_{<t})$$

Training Objective

$$\ell_g(D) = -\sum_{(\mathbf{x},\mathbf{y}^\star)\in D} \log p_g(\mathbf{y}^\star \mid \mathbf{x})$$



Prefix-to-Prefix (Cont.) Training

Encoder only attends to previous words

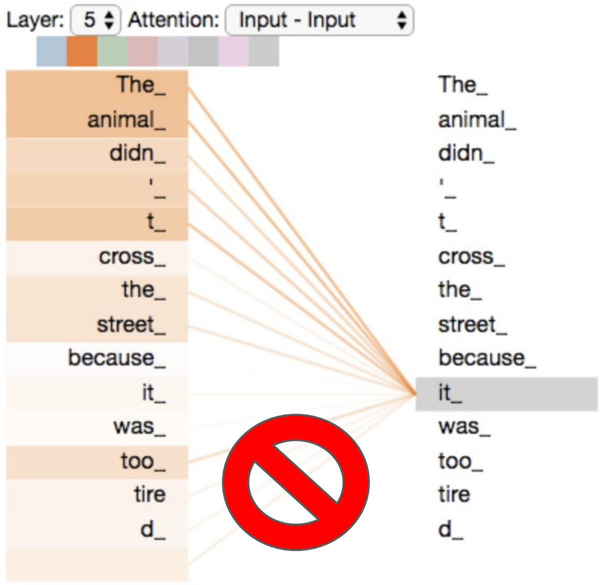


Image from jalammar's page on Transformers

$$\alpha_{ij}^{(t)} = \begin{cases} \frac{\exp e_{ij}^{(t)}}{\sum_{l=1}^{g(t)} \exp e_{il}^{(t)}} & \text{if } i, j \leq g(t) \\ 0 & \text{otherwise} \end{cases}$$

$$e_{ij}^{(t)} = \begin{cases} \frac{P_{W_Q}(x_i)\, P_{W_K}(x_j)^T}{\sqrt{d_x}} & \text{if } i, j \leq g(t) \\ -\infty & \text{otherwise} \end{cases}$$

Latency Metrics

- Consecutive Wait (CW)

  Measures source segment lengths

  Local to source segments

$$\mathrm{CW}_g(\mathbf{x}, \mathbf{y}) = \frac{\sum_{t=1}^{|\mathbf{y}|} \mathrm{CW}_g(t)}{\sum_{t=1}^{|\mathbf{y}|} \mathbb{1}_{\mathrm{CW}_g(t)>0}} = \frac{|\mathbf{x}|}{\sum_{t=1}^{|\mathbf{y}|} \mathbb{1}_{\mathrm{CW}_g(t)>0}}$$

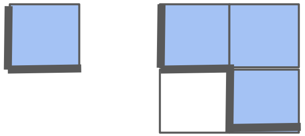
- Average Proportion

  Area above a policy path

  Sensitive to input length

  Proportion is not always clear

$$\mathrm{AP}_g(\mathbf{x}, \mathbf{y}) = \frac{1}{|\mathbf{x}|\ |\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} g(t)$$



Average Lagging

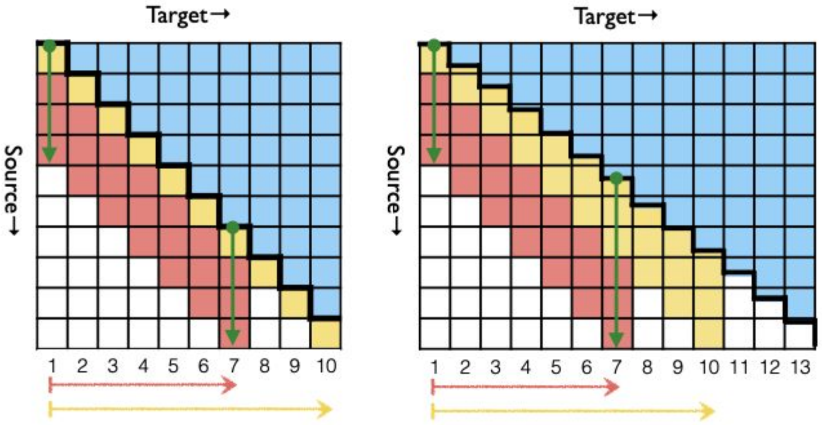
- Average Lagging

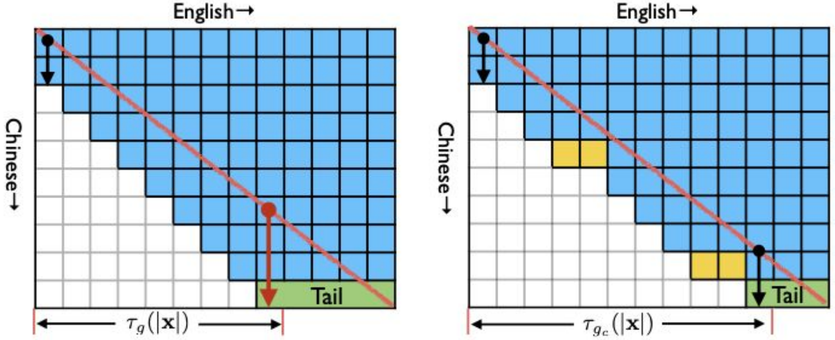
  number of source words the

  target is “lagging” behind source

$$\mathrm{AL}_g(\mathbf{x},\mathbf{y}) = \frac{1}{\tau_g(|\mathbf{x}|)} \sum_{t=1}^{\tau_g(|\mathbf{x}|)} g(t) - \frac{t-1}{r}$$

$$r = |y|/|x|$$



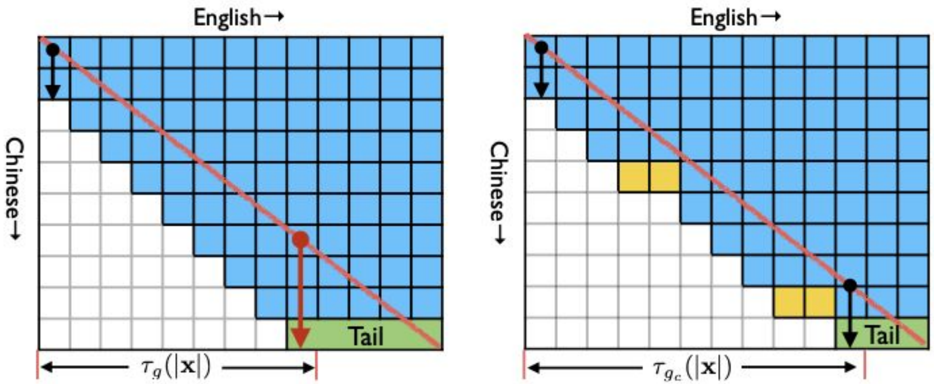


Prefix-to-Prefix Catch-up

- Produce more target side words per source word

$$g_{\text{wait-}k,\,c}(t) = \min\{k + t - 1 - \lfloor ct \rfloor,\ |\mathbf{x}|\}$$

c = |y*|/|x| - 1

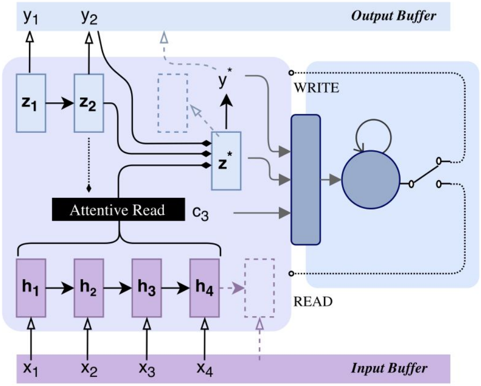


Experimental Setup

- BPE on all texts
- Data sets
  - German-English: *Training* - WMT15, *Dev* - newstest-2013 (dev), *Test* - newstest-2015 (test)
  - Chinese-English: *Training* - NIST corpus, *Dev*- NIST 2006, *Test* - NIST 2008
- Transformer (Vaswani et al., 2017)

Models

- Train Time wait-k
  - Model proposed by the paper
- Test Time wait-k
  - Model trained on full sentences
  - Does wait-k at test time
- Baseline (Gu et al. 2017)



Experimental Results

Train \ Test	k=1	k=3	k=5	k=7	k=9	k=∞
k'=1	*34.1*	33.3	31.8	31.2	30.0	15.4
k'=3	**34.7**	36.7	*37.1*	36.7	36.7	18.3
k'=5	30.7	36.7	37.8	38.4	*38.6*	22.4
k'=7	31.0	**37.0**	**39.4**	*40.0*	39.8	23.7
k'=9	26.4	35.6	39.1	**40.1**	***41.0***	28.6
k'=∞	21.8	30.2	36.0	38.9	39.9	***43.2***

4-ref BLEU, zh->en dev set

	k=3	k=5	k=7	k=3	k=5	k=7
	zh→en			en→zh		
sent-level %	33	21	9	52	27	17
word-level %	2.5	1.5	0.6	5.8	3.4	1.4
accuracy	55.4	56.3	66.7	18.6	20.9	22.2
	de→en			en→de		
sent-level %	44	27	8	28	2	0
word-level %	4.5	1.5	0.6	1.4	0.1	0.0
accuracy	26.0	56.0	60.0	10.7	50.0	n/a

Human Evaluation of Anticipation

Experimental Results (Qualitative)

	1	2	3	4	5	6	7	8	9	10	
	Měiguó	*dāngjú*	*duì*	*Shātè*	*jìzhě*	*shīzōng*	*yī*	*àn*	*gǎndào*	*dānyōu*	
(a)	美国	当局	对	沙特	记者	失踪	一	案	感到	担忧	
	US	authorities	to	Saudi	reporter	missing	a	case	feel	concern	
k=3				the	us	authorities	are	very	concerned	about	the saudi reporter 's missing case
k=3†				the	us	authorities	have	dis-	appeared	from	saudi reporters
										bùmǎn	
(b)	美国	当局	对	沙特	记者	失踪	一	案	感到	不满	
k=3				the	us	authorities	are	very	concerned	about	the saudi reporter 's missing case
k=5						the	us	authorities	have	expressed	**dissatisfaction** with the incident of saudi arabia 's missing reporters

Figure 12: (a) Chinese-to-English example from more recent news, clearly outside of our data. Both the verb "*gǎndào*" ("feel") and the predicative "*dānyōu*" ("concerned") are correctly anticipated, probably hinted by "missing". (b) If we change the latter to *bùmǎn* ("dissatisfied"), the wait-3 result remains the same (which is wrong) while wait-5 translates conservatively without anticipation. †: test-time wait-k produces nonsense translation.

Experimental Results (Qualitative)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
	it	was	learned	that	this	is	the	largest	fire	accident	in	the	medical	and	health	system	nationwide	since	the	founding	of	new	china

k=3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20				
				jù	*liǎojiě*	,	*zhè*	*shì*	*zhōngguó*	*jìn*	*jǐ*	*nián*	*lái*	*fāshēng*	*de*	*zuì*	*dà*	*yī*	*qǐ*	*yīliáo*	*wèishēng*	*xìtǒng*	*huǒzāi*	*shìgù*
				据	了解	，	这	是	中国	近	几	年	来	发生	的	最	大	一	起	医疗	卫生	系统	火灾	事故
				to	known	,	this	is	China	recent	few	years	since	happen	-	most	big	one	case	medical	health	system	fire	accident

k=3†																			
	yīnwèi	*tā*	*shì*	,	*zhègè*	,	*shì*	*zuì*	*dà*	*de*	*huǒzāi*	*shìgù*	,	*zhè*	*shì*	*xīn*	*zhōngguó*	*chénglì*	*yǐlái*
	因为	它	是	，	这个	，	是	最	大	的	火灾	事故	，	这	是	新	中国	成立	以来
	because	it	is	,	this	,	is	most	big	-	fire	accident	,	this	is	new	China	funding	since

Figure 13: English-to-Chinese example in the dev set with incorrect anticipation due to mandatory long-distance reorderings. The English sentence-final clause "since the founding of new china" is incorrectly predicted in Chinese as "近 几 年 来"("in recent years"). Test-time wait-3 produces translation in the English word order, which sounds odd in Chinese, and misses two other quantifiers ("in the medical and health system" and "nationwide"), though without prediction errors. The full-sentence translation, "据 了解，这 是 新 中国 成立 以来，全国 医疗 卫生 系统 发生 的 最 大 的 一 起 火灾 事故", is perfect.

Experimental Results (Latency)

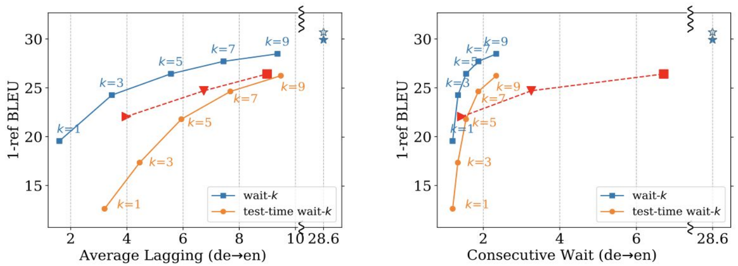


Figure 5: Translation quality against latency metrics (AL and CW) on German-to-English simultaneous translation, showing wait-k and test-time wait-k results, full-sentence baselines, and our adaptation of Gu et al. (2017) (▶:CW=2; ▼:CW=5; ■:CW=8), all based on the same Transformer. ★☆:full-sentence (greedy and beam-search).

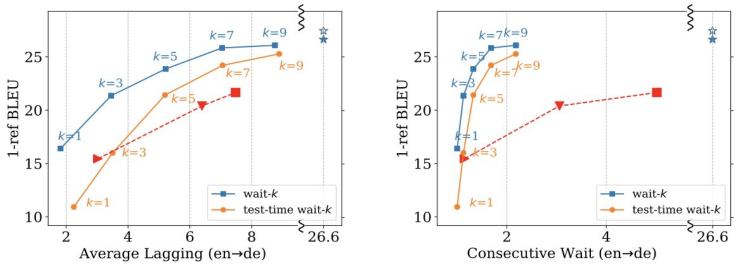


Figure 6: Translation quality against latency metrics on English-to-German simultaneous translation.

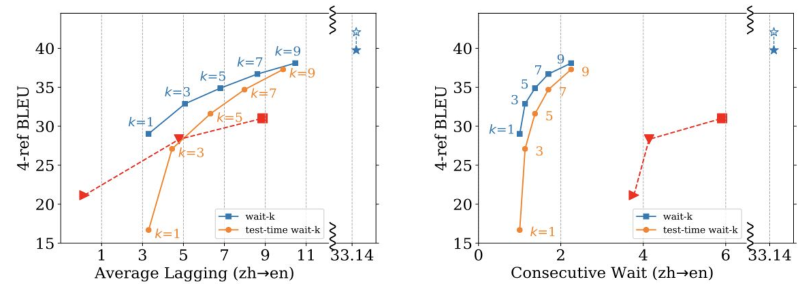


Figure 7: Translation quality against latency on Chinese-to-English simultaneous translation.

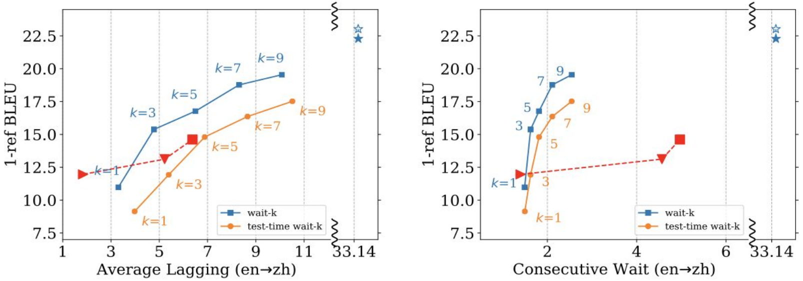


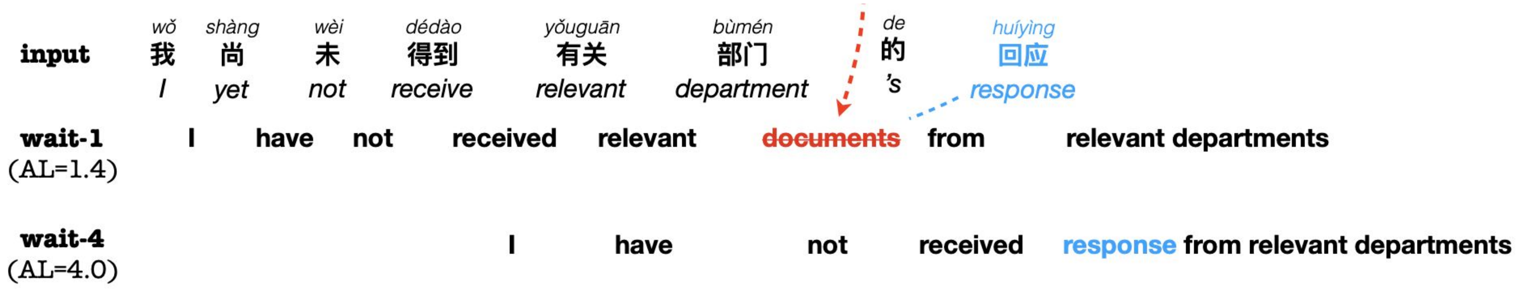
Figure 8: Translation quality against latency on English-to-Chinese, with encoder catchup (see Appendix A).

Results

- New Model has better Performance/Latency
- Can force a fixed latency
- Qualitative Analysis shows anticipation is learned
- Model can be trained prefix-to-prefix
- Existing sentence models can be adapted easily

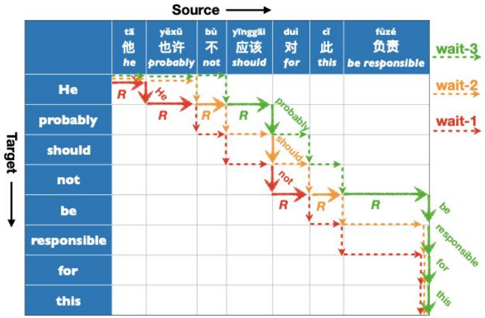
Issues

- Not all results are given for all language pairs
  - English-to-Chinese latency, encoder catch-up, BLEU
- Practical Latency (sec)
- Word ordering is not solved by anticipation



From Huang Liang's presentation

Recent Work



Adaptive Policies

- Simultaneous Translation with Flexible Policy via Restricted Imitation Learning (Zheng et al. 2019)
    - Add READ as a target language token, to simulate READ/WRITE capabilities
    - Imitation training using oracle for expert policy
- Simpler and Faster Learning of Adaptive Policies for Simultaneous Translation (Zheng et al. 2019)
    - Do not retrain model
    - Write if confident, read if unconfident
- Simultaneous Translation Policies: From Fixed to Adaptive (Zheng et al. 2020)
    - Use ensemble of wait-k models, and use best policy dynamically

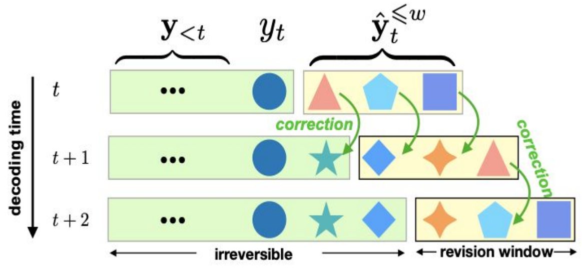
Recent Work

Corrections

- Re-translation versus Streaming for Simultaneous Translation (Arivazhagan et al. 2020)
  - Explores re-translating text against popular "streaming" approaches

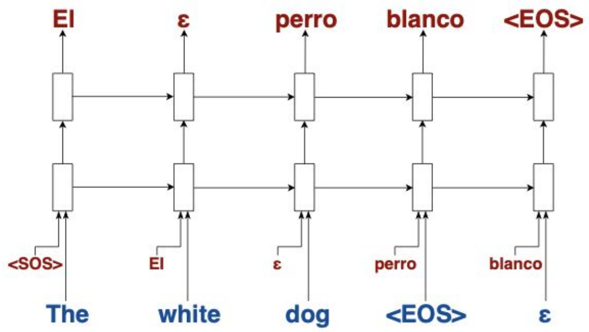
Source	Output								Erasure
1: Neue	New								-
2: Arzneimittel	New	Medicines							0
3: könnten	New	Medicines							0
4: Lungen-	New	drugs	may	be	lung				1
5: und	New	drugs	could	be	lung	and			3
6: Eierstockkrebs	New	drugs	may	be	lung	and	ovarian	cancer	4
7: verlangsamen	New	drugs	may	slow	lung	and	ovarian	cancer	5
Content Delay	1	4	6	7	7	7	7	7	

- Opportunistic Decoding with Timely Correction for Simultaneous Translation (Zheng et al. 2020)
  - Corrects previous words outputted from the model
  - Has a correction window, where only words in the window can be corrected

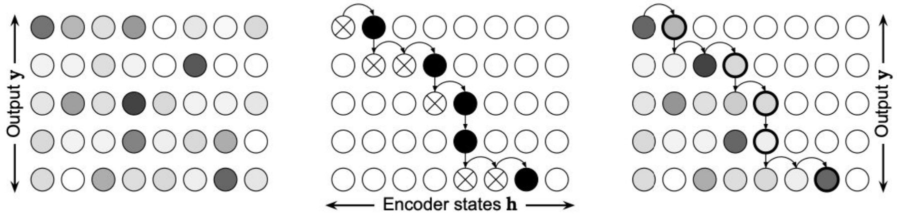


Recent Work

- You May Not Need Attention (Press et al. 2018)
  - Single encoder-decoder model
  - Eager translation, word-by-word



- Monotonic Infinite Lookback Attention for Simultaneous Machine Translation (Arivazhagan et al. 2019)
  - New attention mechanism
  - Attend from left-to-right and from beginning of sentence
  - Adds latency training goal



(a) Soft attention. (b) Monotonic attention. (c) MILk attention.

Q&A

Plug and Play Language Models: A Simple Approach to Controlled Text Generation

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, Rosanne Liu

ICLR 2020

Motivation

- Language Models (LMs) model p(x)

$$p(X) = \prod_{i=1}^{n} p(x_i | x_0, \cdots, x_{i-1}) \quad (1)$$

- Leads to fluent, grammatical text, but it lacks **controllability**

- For example, GPT-2-medium, given “The food is awful” generates:

```
"The food is awful. The staff are rude and lazy. The food is
disgusting - even by my standards."
```

Controlled Text Generation

- Perform **controlled** generation via conditioning generation on attribute, ***a***
  - Sample from p(x | a) instead of p(x)
- Given "The food is awful" with ***a*** = **positive** might generate:

"The food is awful, but there is also the music, the story and the magic! The "Avenged Sevenfold" is a masterfully performed rock musical that will have a strong presence all over the world."

- Given "The potato" with ***a*** = **negative** sentiment might generate:

"The potato is a pretty bad idea. It can make you fat, it can cause you to have a terrible immune system, and it can even kill you..."

Related Work - Controlled Text Generation

There have been some prior attempts at controlled text generation architectures:

- Learning to Write with Cooperative Discriminators Holtzman et al. 2018
  - Use discriminators/attribute models to rank for decoding, may lead to less coherence
  - Referred to as “Weighted Decoding (WD)” in this work
- Fine-Tuning Language Models from Human Preferences, Ziegler et al 2020
  - Start with a pretrained LM, finetune to produce positive outputs - learn p(x | a) by fine tuning
  - Data collected in online fashion and RL objective trained from human evaluators
- CTRL: A Conditional Transformer Language Model for Controllable Generation, Keskar et al 2019
  - Train a conditional model from scratch - learn p(x | a) from scratch
  - 1.6 billion parameters, ~50 control codes form URL and subreddits
- Delete, Retrieve, Generate: A Simple Approach to Sentiment and Style Transfer, Li et al 2018
  - Retrieval based approaches, relies on transforming text
  - Use neural methods for extracting attribute markers

Main Problem

- Generating on an attribute ***a*** lets us direct output, making it more…
  - human like through a coherent/consistent direction
  - topical through specifying ***a*** to be a particular topic
  - like anything which can be modeled with an attribute ***a***

- Existing approaches require fine-tuning existing models, or training from scratch with control codes. This is:
  - costly due to the need to collect data with attribute ***a***
  - data inefficient as RL/DL from scratch needs millions of training episodes
  - not flexible as codes are fixed or models are fine tuned for only one attribute

Problem Setting

- Given an attribute, ***a***, we want to generate text conditioned on this attribute from a language model
  - Generate from p(x | a)

- Desirable properties are that it requires:
  - few computational resources
  - little to no training
  - high adaptability

Related Work - Alternative Controlled Schema

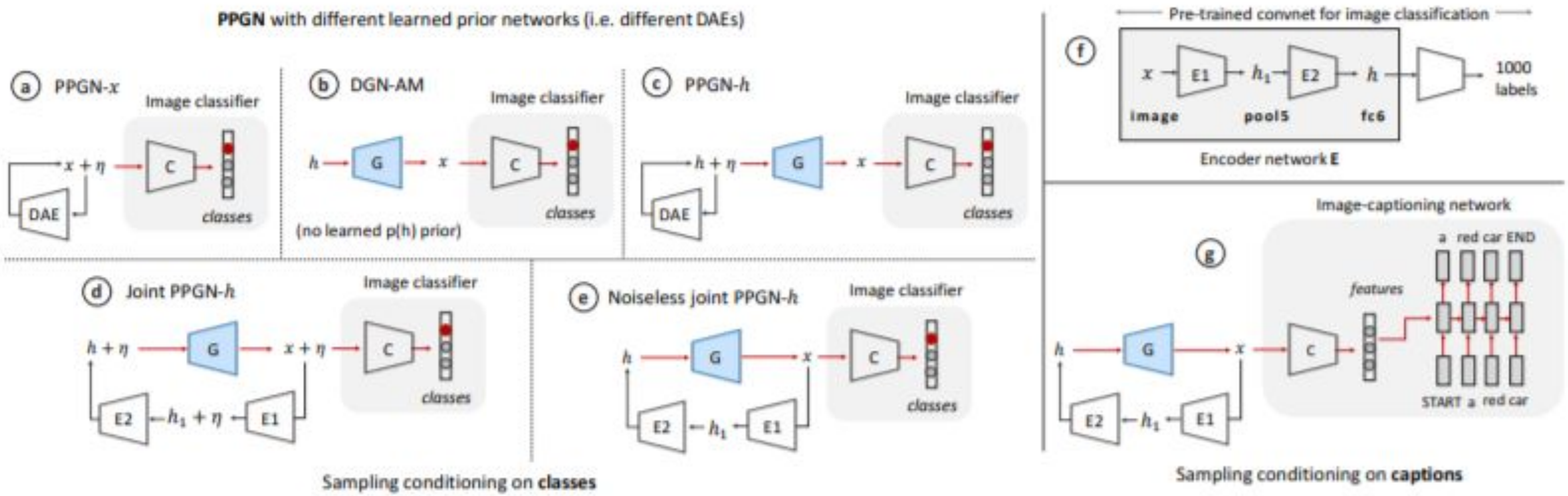
- Simple and Effective Noisy Channeling Yee et al. 2019
  - Use similar application of Bayes rule: $p(y|x) = p(x|y)p(y)/p(x)$ for NMT

- SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient , Yu et al 2017
  - Train a GAN, treating the sequence generation as sequential decision problem (RL)
  - Use a discriminator to guide training

- Multiple Attribute Text Style Transfer Subramanian et al. 2019
  - Uses denoised autoencoding to style transfer, makes use of back translation similar to this work

Related Work - Plug and Play

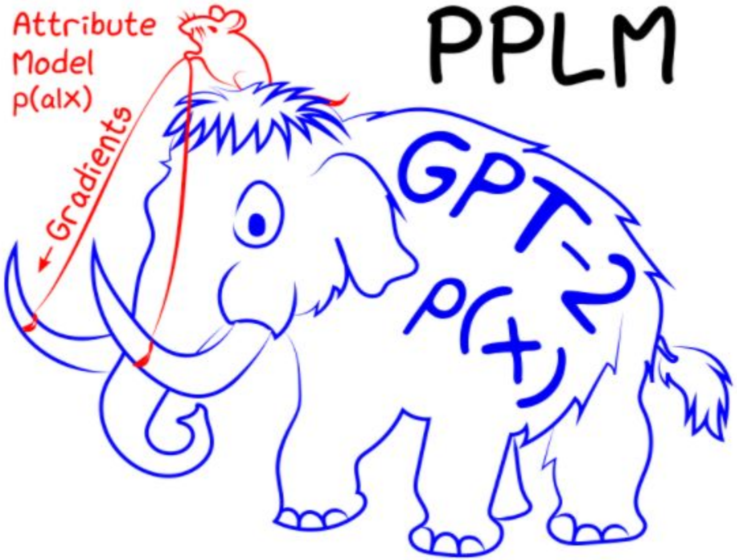
- Plug and Play Generative Networks Conditional Iterative Generation of Images in Latent Space Nguyen et al 2017
  - Introduces plug and play in vision, similar motivation to manipulate latent space



Compare and Contrast: PPGN and PPLM

- PPGN
  - h -> x -> y where h is latent code, x is image, y is attribute
  - Noise added in h space for image diversity
  - Markov chain in h space to sample probability distribution
- PPLM:
  - [x1 -> (h1, x2) -> …] -> y where h_t is latent, x_t is byte-pairs, and y is attribute
  - Noise is naturally introduced by sampling of each x to obtain sentence diversity
  - No Markov chain - instead, sliding window of h's history is used to sample words one at a time

PPLM Overview



- Combine:
  - Pretrained LM which models p(x)
  - Discriminator/attribute model, which models p(a | x)

- Use Bayes rule: $p(x \mid a) \propto p(a \mid x) * p(x)$

- The small attribute model will "steer" the gradients to alter the activation functions to prefer things of desired attribute

- Modularity: the LM and attribute model can be anything modeling p(x) and p(a | x) respectively

How PPLM works

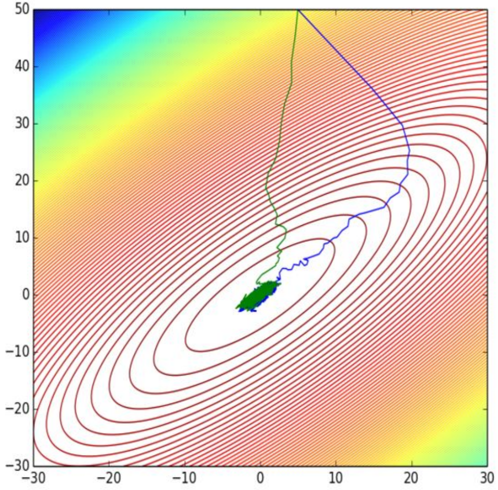
Application of Metropolis-adjusted Langevin sampler (MALA) Roberts and Tweedie 1996 on H_t where

$H_t = [(K_t^0, V_t^0), \ldots, (K_t^l, V_t^l)]$ where (K_t^i, V_t^i) are the transformer key value pairs generated from time 0 to t

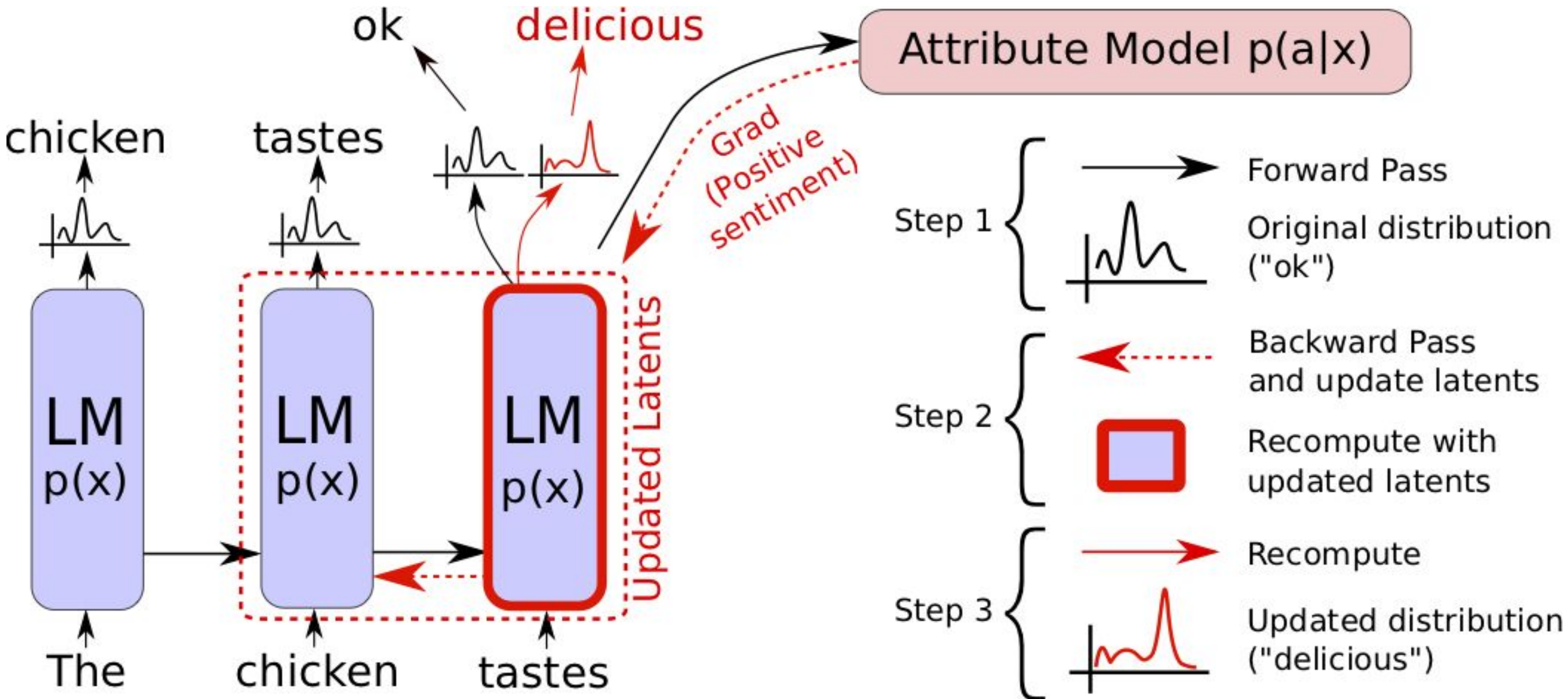
1. From partial sentence ***x*** - compute log(p(x)) and log(p(a|x)) and gradients w.r.t hidden rep H_t

2. Using gradients, move H_t a small step increasing log(p(a|x)) and increasing log(p(x)).

3. Sample the next word and repeat



How PPLM works illustrated



Methodology - Maximizing p(a|x)

- Want to perform an update to the latent space to shift towards higher LL of ***a***
  - ΔH_t starts at 0, updated by gradients from attribute model - is the “reinterpretation”of the past
  - log p(a | $H_t + \Delta H_t$) = log p(a | x) (with the update)
  - **α** is the step size update
  - γ is the per layer normalization term (transformer specific)

$$\Delta H_t \leftarrow \Delta H_t + \alpha \frac{\nabla_{\Delta H_t} \log p(a|H_t + \Delta H_t)}{\|\nabla_{\Delta H_t} \log p(a|H_t + \Delta H_t)\|^{\gamma}} \quad (3)$$

- This update is performed 3 - 10 times, then we use the LM on: $\tilde{H}_t = H_t + \Delta H_t$. to get p(t’)

Methodology - Remembering p(x)

❖ Pushing the model towards higher LL of ***a*** will lead to degeneration

❖ Two fixes for this:

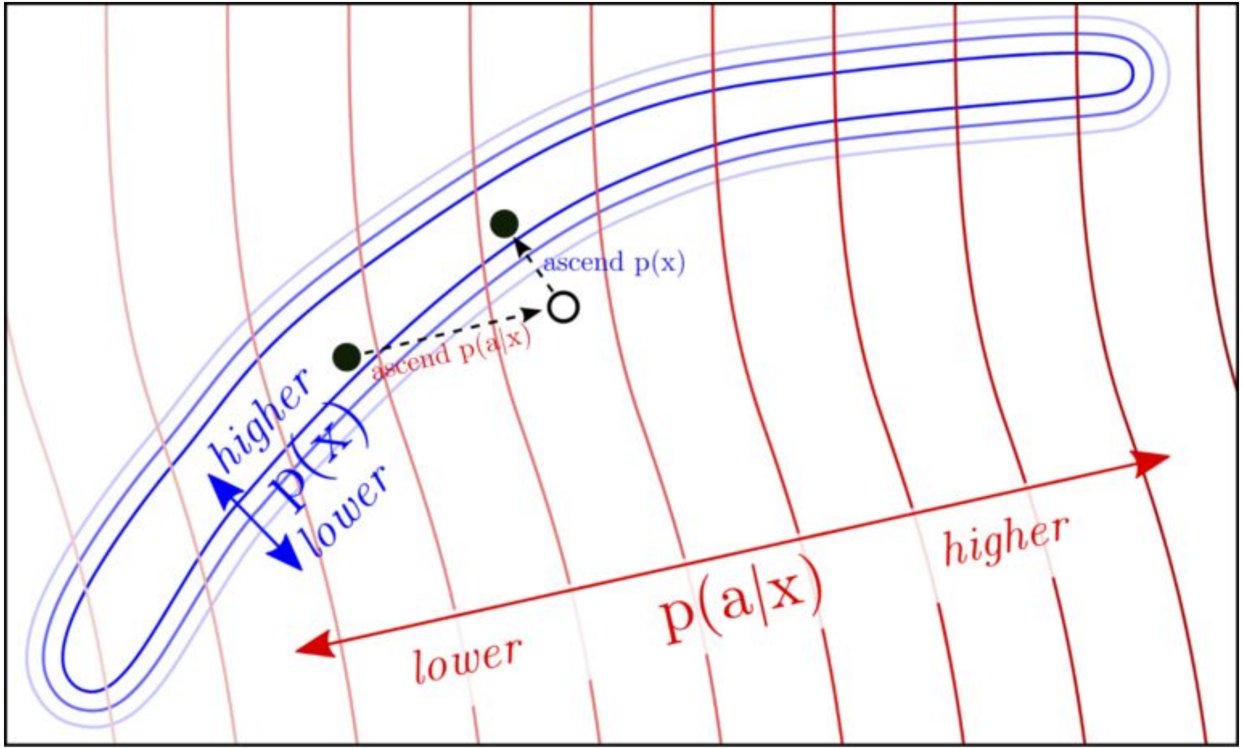
1. Update ΔH_t to minimize KL Divergence - add p(t)'s before taking gradient, scale by λKL:

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

2. Post norm Geometric mean fusion - sample from a combination of the distributions

$$x_{t+1} \sim \frac{1}{\beta}\left(\widetilde{p}_{t+1}^{\gamma_{gm}}\, p_{t+1}^{1-\gamma_{gm}}\right)$$

Method - Putting it all together, illustration



Experimental Setup

- Evaluate controlled text generation using top-10 sampling, ranking with the attribute model p(a | x), and discarding poor quality generation below a threshold for mean of Dist-1, Dist-2, and Dist-3

- Models compared
  - **B** and **BR** - GPT-2 unchanged and sampled once (**B)** or R samples with ranking (**BR**)
  - **BC** and **BCR** - this paper's method sampled once (**BC)** or R samples with ranking (**BCR**)
  - **CTRL, GPT-FT-RL,** and **WD** - alternative approaches introduced in related works

- Authors evaluate with:
  - **Automatic metrics**: Perplexity, Dist-1, Dist-2, and Dist-3
  - **Human metrics**: Fluency (1-5) & A/B testing for attribute (model A, model B, both or neither)

Experimental Results - BOW model

- **Using simple BOW as attribute model:**

$$\log p(a|x) = \log \left(\sum_{i}^{k} p_{t+1}[w_i] \right). \quad (4)$$

  - SCIENCE, MILITARY, LEGAL, COMPUTERS, SPACE, POLITICS, and RELIGION
  - Generated 420 samples from 7 topics, 20 prefixes
  - Even with simple attribute model, performs much higher in topicality with slightly worse automatic metrics (especially perplexity)

Method	Topic % (↑ better) (human)	Perplexity (↓ better)	Dist-1 (↑ better)	Dist-2 (↑ better)	Dist-3 (↑ better)	Fluency (↑ better) (human)
B	11.1	39.85±35.9	0.37	0.79	0.93	3.60±0.82
BR	15.8	38.39±27.14	0.38	0.80	0.94	3.68±0.77
BC	46.9	43.62±26.8	0.36	0.78	0.92	3.39±0.95
BCR	**51.7**	44.04±25.38	0.36	0.80	0.94	3.52±0.83
CTRL	50.0	24.48±11.98	0.40	0.84	0.93	3.63±0.75
BCR	**56.0**	–	–	–	–	3.61±0.69
WD	35.7	32.05±19.07	0.29	0.72	0.89	3.48±0.92
BCR	**47.8**	–	–	–	–	3.87±0.71

Experimental Results - Discriminator

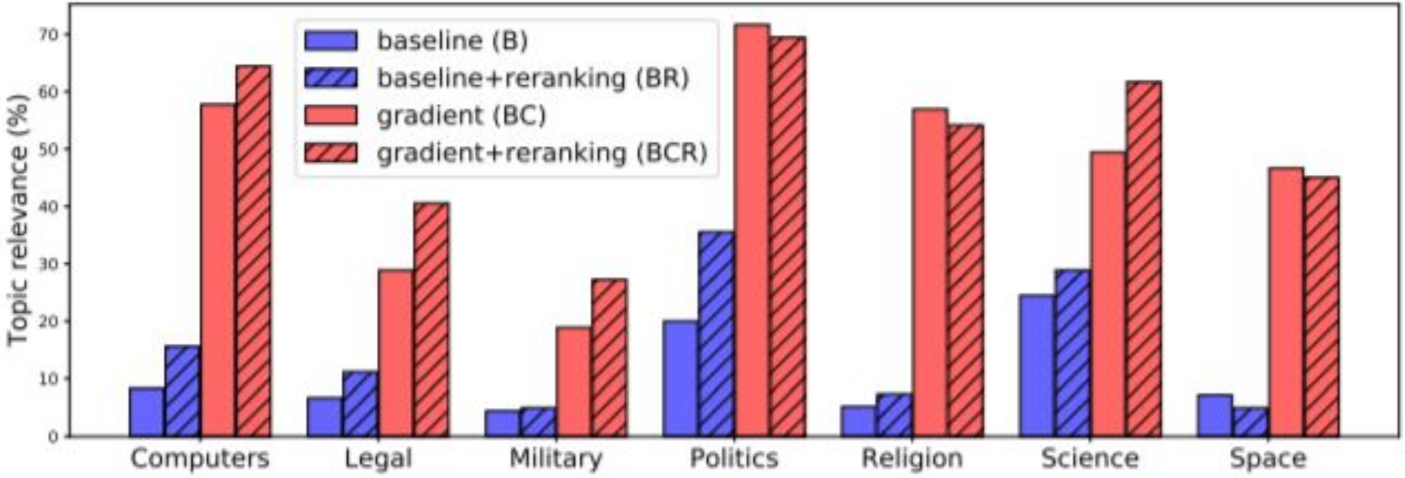
- **Train a single layer classifier for sentiment extraction**

$$\log p(a|x) = \log f(o_{:t+1}, o_{t+2}) \quad (5)$$

  - Trained on the SST-5 dataset (movie reviews)
  - Use 15 prefixes to generate 45 samples, VERY POS and VERY NEG
  - BR much more effective because topics, but BCR still performs quite well in sentiment

Method	Sentiment Acc. (%) (human)	Sentiment Acc. (%) (external classifer)	Perplexity (↓ better)	Dist-1 (↑ better)	Dist-2 (↑ better)	Dist-3 (↑ better)	Human Evaluation Fluency (↑ better)
B	19.3	52.2	42.1±33.14	0.37	0.75	0.86	3.54±1.08
BR	41.5	62.2	44.6±34.72	0.37	0.76	0.87	3.65±1.07
BC	39.6	64.4	41.8±34.87	0.33	0.70	0.86	2.79±1.17
BCR	**73.7**	**78.8**	46.6±40.24	0.36	0.77	0.91	3.29±1.07
CTRL	**76.7**	96.6	37.4±16.89	0.35	0.78	0.89	3.54±0.77
BCR	70.0	–	–	–	–	–	3.36±0.82
GPT2-FT-RL*	13.3	77.8	217.3±176.4	0.54	0.91	0.94	3.31±0.84
BCR	**84.4**	–	–	–	–	–	3.68±0.83
WD	18.9	52.2	31.7±28.0	0.33	0.69	0.83	3.67±0.89
BCR	**61.1**	–	–	–	–	–	3.75±0.66

Ablation Study - BOWS



Qualitative - BOW with odd topics

BOW using Military:

“**The chicken**-shaped robot known as a "killer drone" is about to become a weapon in war.\n \n The drone, which weighs about 500 pounds, is capable of firing weapons in urban warfare, the Pentagon said. It can fly at speeds of 30 miles an hour.\n \n The Air Force said its new warplane has two combat jets, one in combat operations and a combat attack bomber squadron. Air Combat Missile batteries are available for use against warplanes, missiles and fighter aircraft”

Qualitative - Discriminator with odd topics

Discriminator with positive setting

"**My dog died** at the age of 92 years this year. He was a legend in our home state of Virginia. I have a tremendous heart, my soul, my spirit, my love. He is in heaven and I love him very much. We have a great bond. He is gone and I am so thankful for him!\n \n He had been in the hospital for so many months. My husband of over 60 years, Dr. Paul, was in charge of his care. He took his place as the. . . "

Qualitative - Multiple Knobs

Trained on a clickbait discriminator

[Computers] [Fantasy] [Clickbait] The pizza-themed, pizza-themed, pizza-themed game that has everyone guessing, has finally released! The game, known as Pizza-Pocalypse, has released on Steam and on GOG. Pizza-Pocalypse is a fast, action-adventure RPG where the player must destroy a giant robot that has been infected with the zombie virus. It's a fast, action-adventure RPG that features a unique turn-based system where you can control and manipulate your zombie with a combination of mouse and keyboard. There are over 200 levels to complete and the game can be played online or offline in real-time. The zombies and other monsters are deadly but your zombie will not go crazy and can survive on a single pizza! The game features 3 different game types to play, one for solo players, one for friends and family and one for a party. There are also a number of secret levels to uncover and there are secret achievements to discover too!...

Contributions

- Performance is slightly lower than 1.6 billion parameter, trained from scratch CTRL but beats other efforts and is comparable in human evaluation
- PPLM is very simple solution for learning conditional text generation, or p(x | a)
- PPLM is an incredibly flexible system - anything that can be modeled with p(a | x) can be usedfor conditional text generation
- Can be applied to story generation or language detoxification

Limitations of the Work

- Getting this system to work requires lots of tuning and “tricks” (despite the name)
  - Hyperparameters for MALA, KL divergence, and geometric fusion need tuning
  - Only modify a finite horizon of H (5 found to be best setting), 3-10 passes for H
- Different topics and/or attribute models require different hyperparameter settings
  - See Table S18 in appendix
- Highly dependent on the attribute model (errors could compound!)
- PPLM is high variance, only operates on the transformer latent space

Future/Related Work

- Towards Controllable Biases in Language Generation Sheng et al 2020
  - Use gradients to form a bias trigger instead of latent space updates
- Few Shot Natural Language Generation for Task Oriented Dialog Peng et al 2020
  - Build a NLG benchmark for few shot controllable text generation
- Conditional Rap Lyrics Generation with Denoising Autoencoders Nikolov et al 2020
  - Use conditional text generation to generate topical rap lyrics
- You are right. I am ALARMED – But by Climate Change Counter Movement Bhatia et al 2020
  - Point to the dangers of this paper (used for climate deniers/fake news)
- GEDI: Generative Discriminator Guided Sequence Generation Krause et al 2020
  - Bake discriminator into training from scratch (combination of CTRL and this work). Leads to generalization to new attributes

Summary

- Conditional text generation allows for controlled generation while retaining fluency
- Requires directly training a model for p(x | a), or fine-tuning on existing models
  - This process is costly, data inefficient, and inflexible
- This paper proposes a flexible attribute model to steer language model's activations
  - Improves likelihood of p(a | x) for control while maintaining likelihood of p(x) for fluency
- Produces performance near CTRL while being more lightweight and flexible and requiring no fine tuning
- Able to handle odd topic combinations and multiple attributes with grace
- Check out blogpost at: https://eng.uber.com/pplm/