Controllable Text Generation

jcykcai, 20220901

Problem Statement

- We have a pre-trained general LM *p(x)* and we want to generate text with a desirable attribute *a p(x|a)* (or multiple attributes)
- formality, topic, style, sentiment, detoxification, etc
- The most basic baseline: fine-tuning a class-conditional language model
    - Fine-tuning large LMs can be expensive
    - Difficult to preserve the desirable quality of *p(x)*
    - Need a separate LM for each attribute

GeDi: Generative Discriminator Guided Sequence Generation

WARNING: This paper contains GPT-3 outputs which are offensive in nature.

Ben Krause[*], **Akhilesh Deepak Gotmare**[*], **Bryan McCann**[†], **Nitish Shirish Keskar**
Shafiq Joty, Richard Socher[†], **Nazneen Fatema Rajani**

Salesforce Research
{bkrause,akhilesh.gotmare}@salesforce.com

EMNLP2021 Findings

Motivation

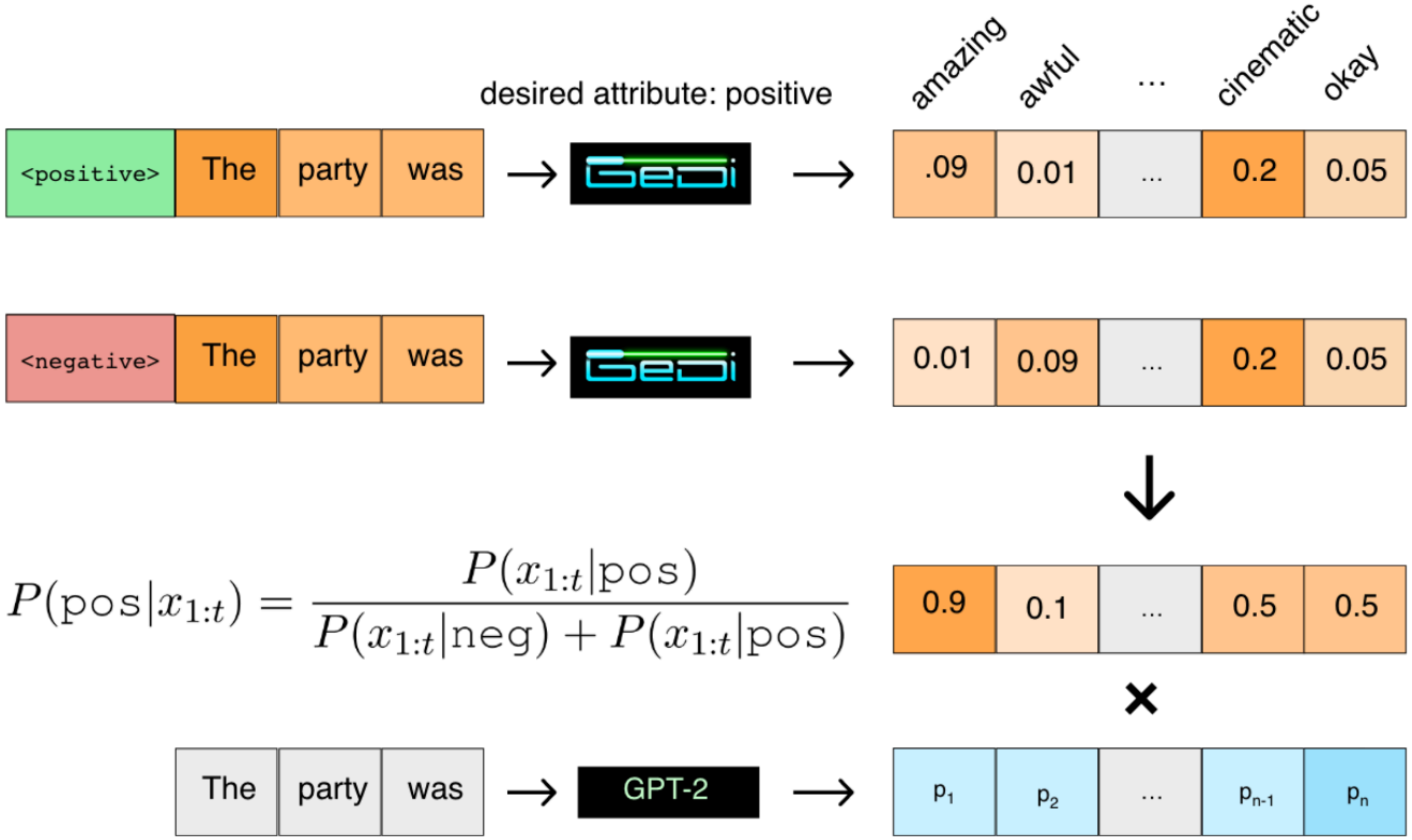
- Fine-tuning large LMs can be expensive
- Difficult to preserve the desirable quality of *p(x)*
- Need a separate LM for each attribute

- Train smaller-sized LMs as discriminators
- Apply Bayes rule

$$P_w(x_t|x_{<t}, c) \propto \ P_{LM}(x_t|x_{<t})P_\theta(c|x_t, x_{<t})^\omega,$$

Methods

$$P_\theta(c|x_{1:t}) = \frac{P(c)\prod_{j=1}^{t} P_\theta(x_j|x_{<j},c)}{\sum_{c'\in\{c,\bar{c}\}}\prod_{j=1}^{t} P(c')P_\theta(x_j|x_{<j},c')}$$



desired attribute: positive

amazing awful … cinematic okay

<positive> The party was → GeDi → .09 0.01 ... 0.2 0.05

<negative> The party was → GeDi → 0.01 0.09 ... 0.2 0.05

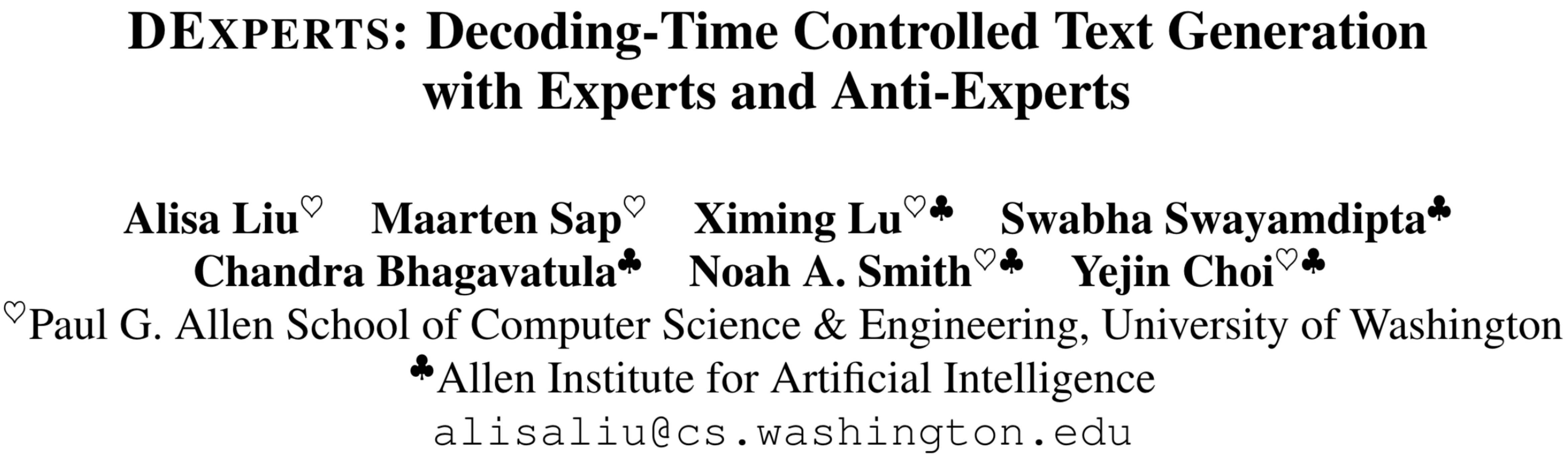
↓

$$P(\text{pos}|x_{1:t}) = \frac{P(x_{1:t}|\text{pos})}{P(x_{1:t}|\text{neg}) + P(x_{1:t}|\text{pos})}$$

0.9 0.1 ... 0.5 0.5

×

The party was → GPT-2 → p_1 p_2 ... p_{n-1} p_n



DExperts: Decoding-Time Controlled Text Generation with Experts and Anti-Experts

Alisa Liu[♡] Maarten Sap[♡] Ximing Lu[♡♣] Swabha Swayamdipta[♣]
Chandra Bhagavatula[♣] Noah A. Smith[♡♣] Yejin Choi[♡♣]
[♡]Paul G. Allen School of Computer Science & Engineering, University of Washington
[♣]Allen Institute for Artificial Intelligence
alisaliu@cs.washington.edu

ACL2021

Motivation

- Fine-tuning large LMs can be expensive
- Difficult to preserve the desirable quality of *p(x)*
- Need a separate LM for each attribute

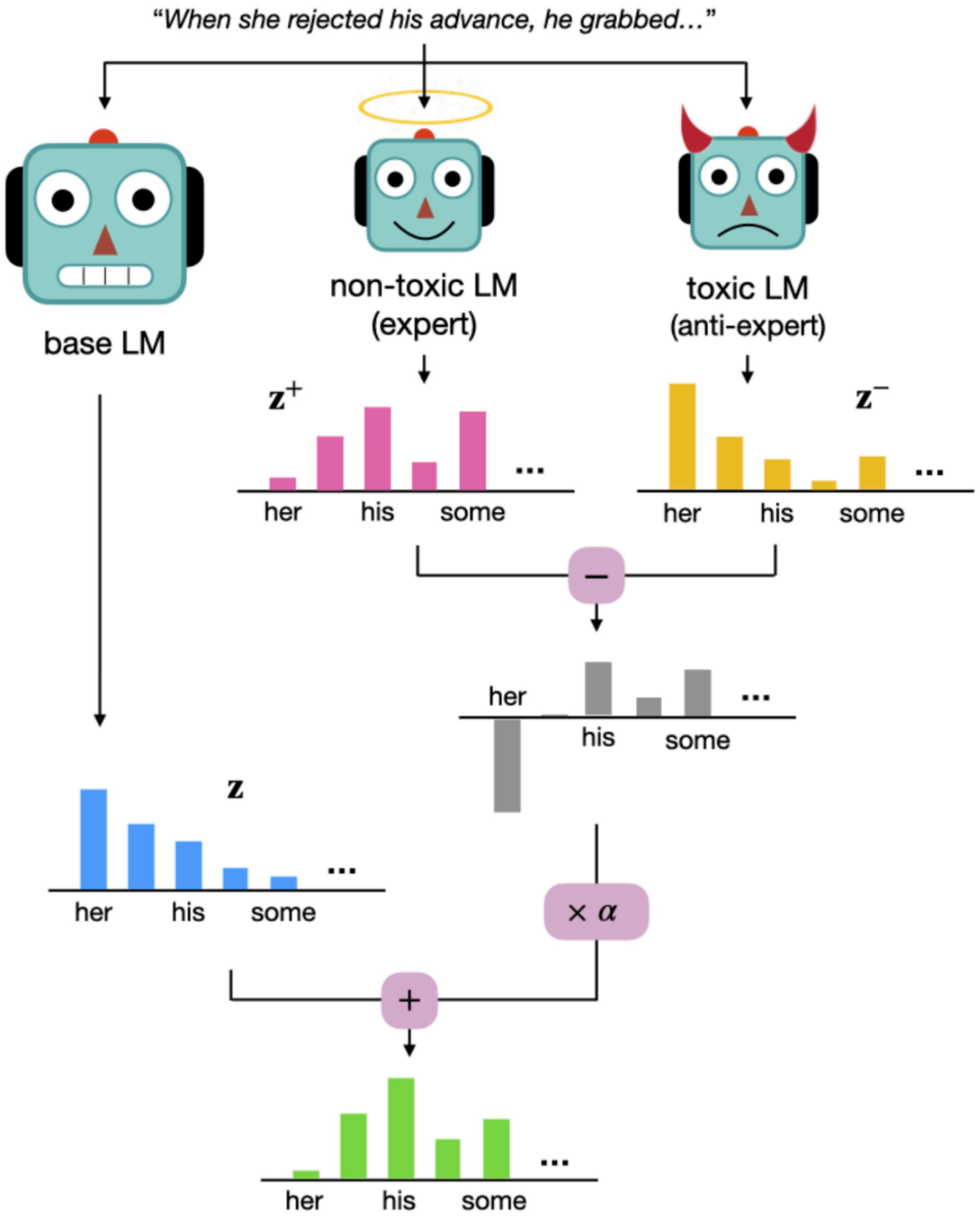
- Train smaller-sized LMs on text with desirable and undesirable attributes (experts and anti-experts)
- Combine base LM with experts and anti-experts

Methods

$$P(X_t \mid \boldsymbol{x}_{<t}) = \text{softmax}(\mathbf{z}_t)$$

$$\downarrow$$

$$\tilde{P}(X_t \mid \boldsymbol{x}_{<t}) = \text{softmax}\left(\mathbf{z}_t + \alpha\left(\mathbf{z}_t^+ - \mathbf{z}_t^-\right)\right)$$



Summary

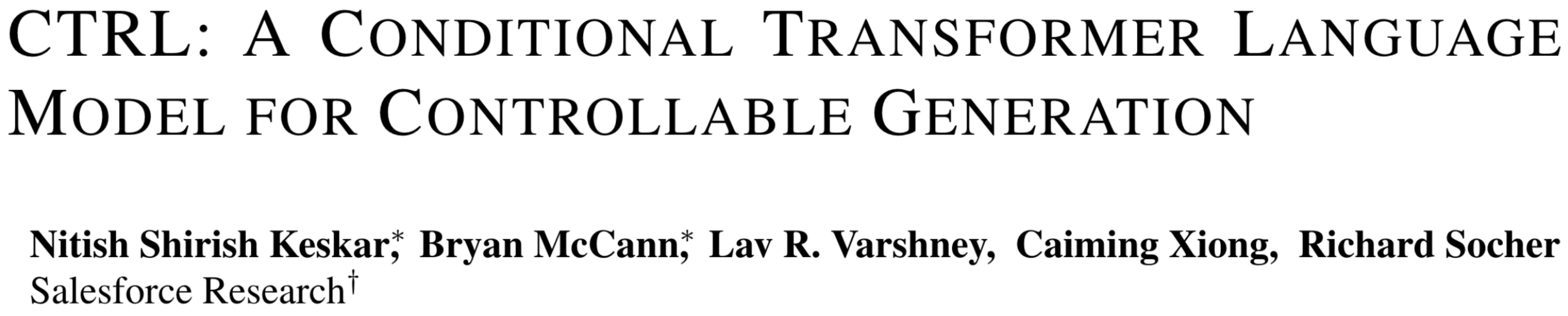
- Fine-tuning large LMs can be expensive
- Difficult to preserve the desirable quality of *p(x)*
- Need a separate LM for each attribute

- smaller-sized class-conditional LMs
    - Used in a weighted combination of generators
    - Used as generative discriminators

Summary

- Fine-tuning large LMs can be expensive
- Difficult to preserve the desirable quality of *p(x)*
- Need a separate LM for each attribute

- smaller-sized class-conditional LMs
  - Used in a weighted combination of generators
  - Used as generative discriminators



CTRL: A Conditional Transformer Language Model for Controllable Generation

Nitish Shirish Keskar,* Bryan McCann,* Lav R. Varshney, Caiming Xiong, Richard Socher
Salesforce Research†

ArXiv2019

Motivation

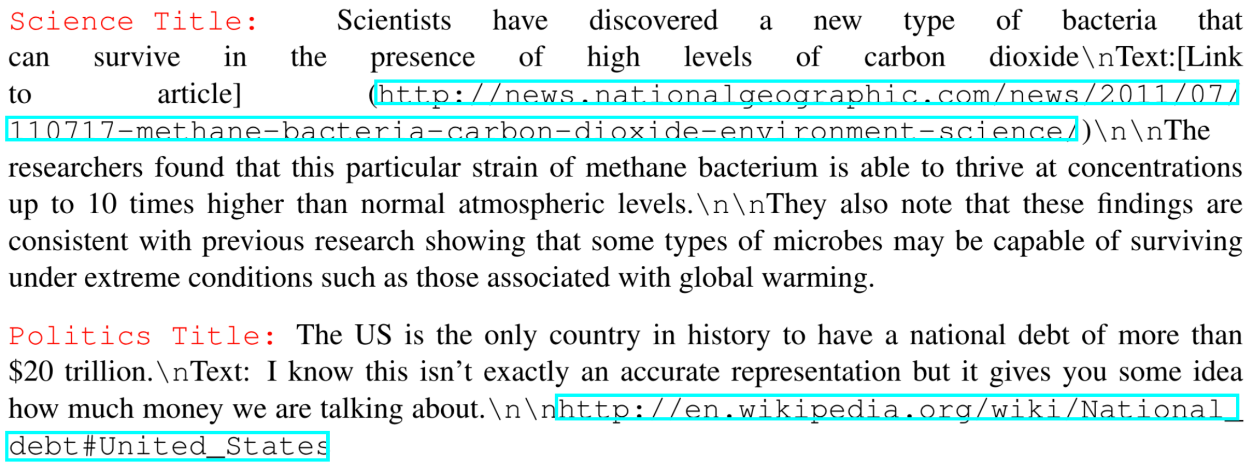
- Many fixed class-conditional language model => a single variable class-conditional language model

- Add a class variable that describes the attribute of the text to generate

Methods

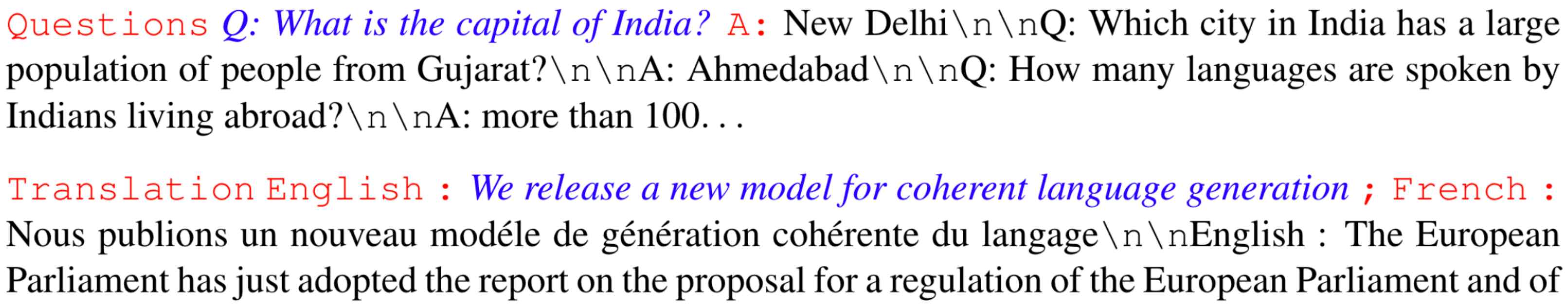
- CTRL is a conditional language model that is always conditioned on a control code *c*

- CTRL is trained with natural control codes
  - Domain, subdomain, entities, entity relations, and even dates
  - Task



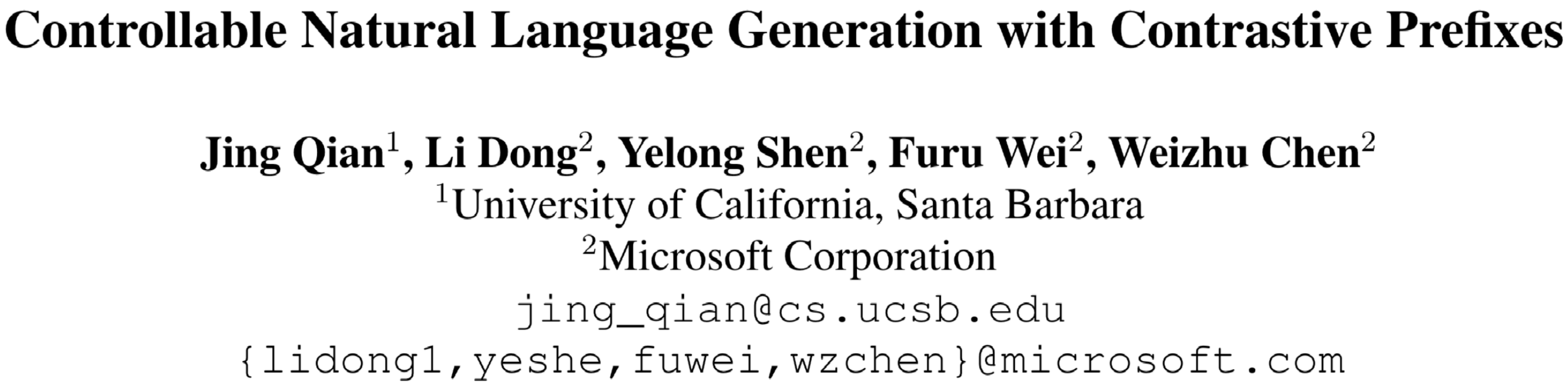
Science Title: Scientists have discovered a new type of bacteria that can survive in the presence of high levels of carbon dioxide\nText:[Link to article] (http://news.nationalgeographic.com/news/2011/07/110717-methane-bacteria-carbon-dioxide-environment-science/)\n\nThe researchers found that this particular strain of methane bacterium is able to thrive at concentrations up to 10 times higher than normal atmospheric levels.\n\nThey also note that these findings are consistent with previous research showing that some types of microbes may be capable of surviving under extreme conditions such as those associated with global warming.

Politics Title: The US is the only country in history to have a national debt of more than $20 trillion.\nText: I know this isn't exactly an accurate representation but it gives you some idea how much money we are talking about.\n\nhttp://en.wikipedia.org/wiki/National_debt#United_States



Questions *Q: What is the capital of India?* A: New Delhi\n\nQ: Which city in India has a large population of people from Gujarat?\n\nA: Ahmedabad\n\nQ: How many languages are spoken by Indians living abroad?\n\nA: more than 100. . .

Translation English : *We release a new model for coherent language generation* ; French : Nous publions un nouveau modéle de génération cohérente du langage\n\nEnglish : The European Parliament has just adopted the report on the proposal for a regulation of the European Parliament and of



Controllable Natural Language Generation with Contrastive Prefixes

Jing Qian[1], Li Dong[2], Yelong Shen[2], Furu Wei[2], Weizhu Chen[2]
[1]University of California, Santa Barbara
[2]Microsoft Corporation
jing_qian@cs.ucsb.edu
{lidong1,yeshe,fuwei,wzchen}@microsoft.com

ACL2022

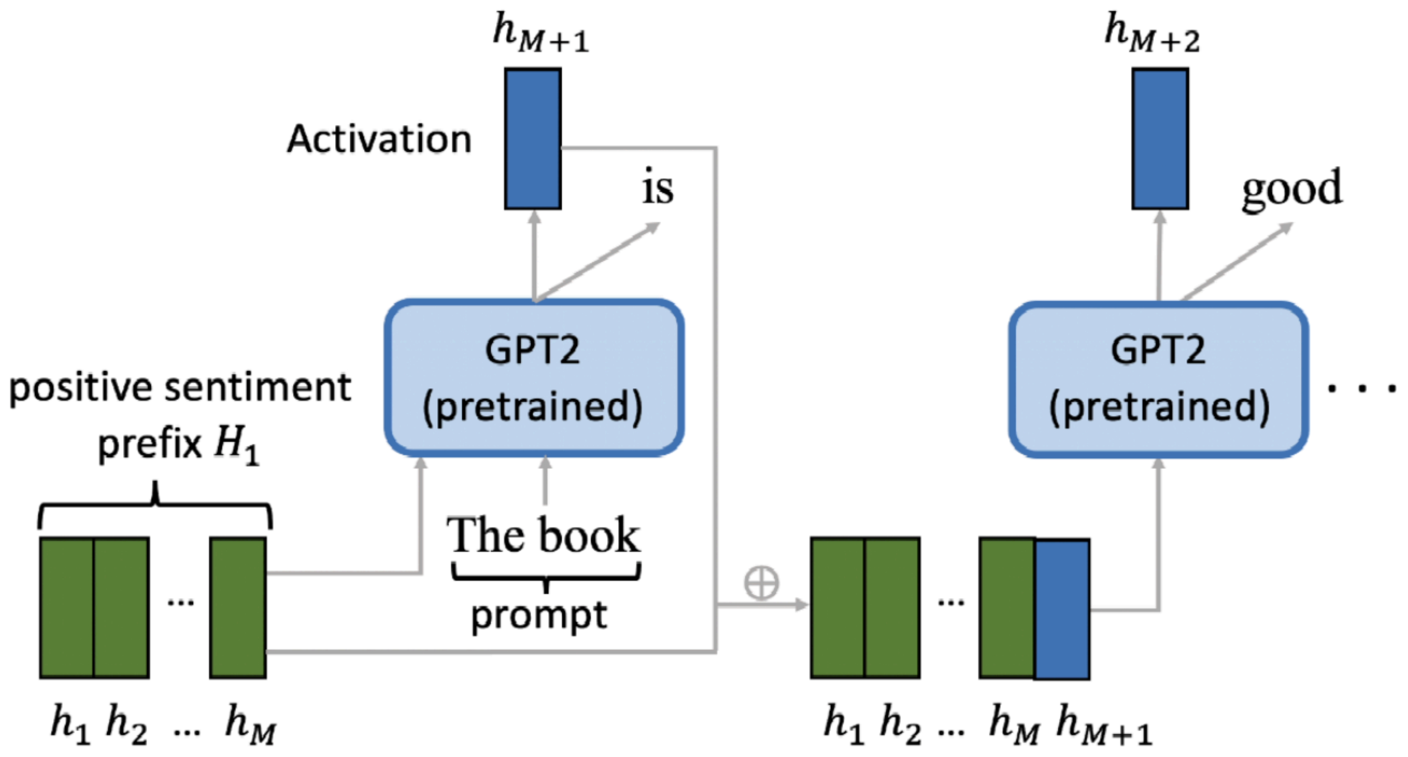
Motivation

- CTRL is expensive (1.63B parameters) and lacks flexibility since the control codes are fixed.
- Lightweight and flexible fine-tuning:
  - introduce a fewer additional parameters
  - Easy to add a new attribute control

Methods

- Prefix-tuning: optimize a a set of small continuous attribute-specific vectors for steer text generation.

    - The original parameters of GPT2 is fixed



Methods

- Supervised Training (text with annotated attributes)

$$\mathcal{L}_{sup} = \omega_1 \mathcal{L}_{LM} + \omega_2 \mathcal{L}_d$$

$$\mathcal{L}_{LM} = -\sum_{t=1}^{T} \log p(x_t | x_{<t}, y)$$

$$\mathcal{L}_d = -\log \frac{p(y)p(x|y)}{\sum_{y' \in Y} p(y')p(x|y')}$$

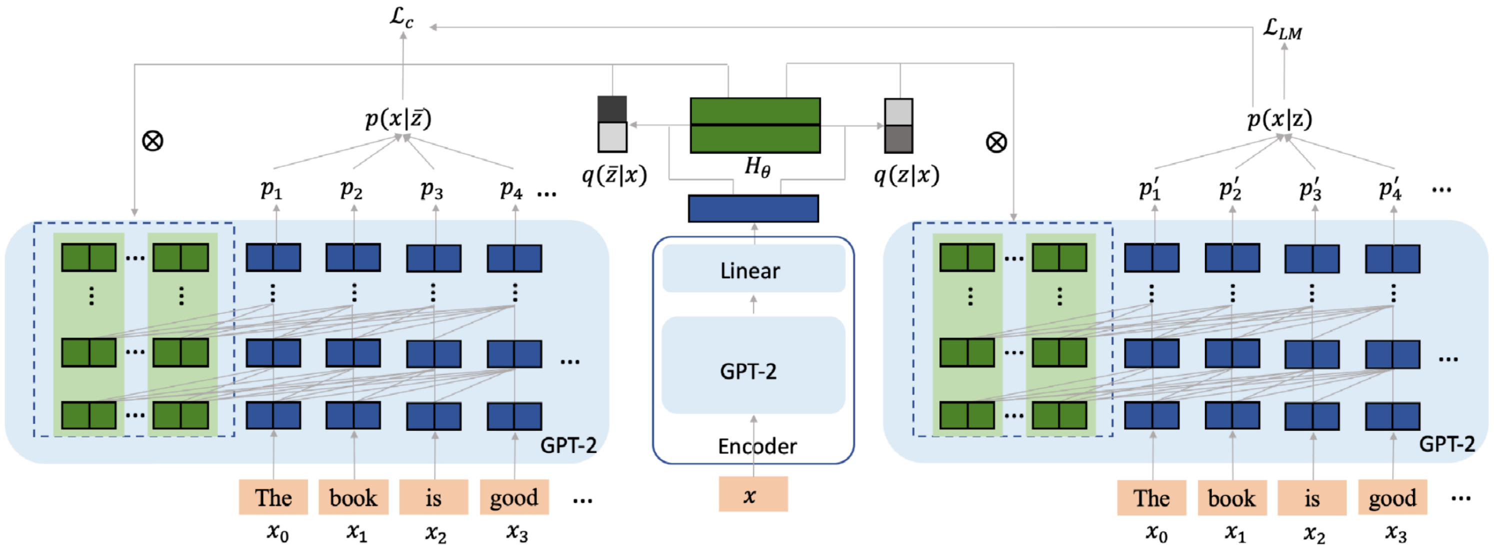
- Unsupervised Training (attribute as latent variable)

$$\mathcal{L}_{uns} = \omega_1 \mathcal{L}_{LM} + \omega_2 \mathcal{L}_{KL} + \omega_3 \mathcal{L}_c$$

$$\mathcal{L}_{LM} = -\sum_{t=1}^{T} \log p(x_t | x_{<t}, z)$$

$$\mathcal{L}_{KL} = KL[q(z|x) || p(z)]$$

$$\mathcal{L}_c = \max(m - \|p(z|x) - p(\bar{z}|x)\|_2, 0)^2$$



Fine-Grained Controllable Text Generation Using Non-Residual Prompting

Fredrik Carlsson*[*] **Joey Öhman[†]** **Fangyu Liu[‡]**
Severine Verlinden[†] **Joakim Nivre[*]** **Magnus Sahlgren[†]**

[*]Research Institutes of Sweden
fredrik.carlsson@ri.se
joakim.nivre@ri.se

[†]AI Sweden
joey.ohman@ai.se
severine.verlinden@ai.se
magnus.sahlgren@ai.se

[‡]University of Cambridge
fl339@cam.ac.uk

ACL2022

Motivation

- The prompt’s influence is negatively correlated with the distance from the prompt to the next predicted token.

- Different to the previous work (Qian et al, 2022), it uses textual prompts.

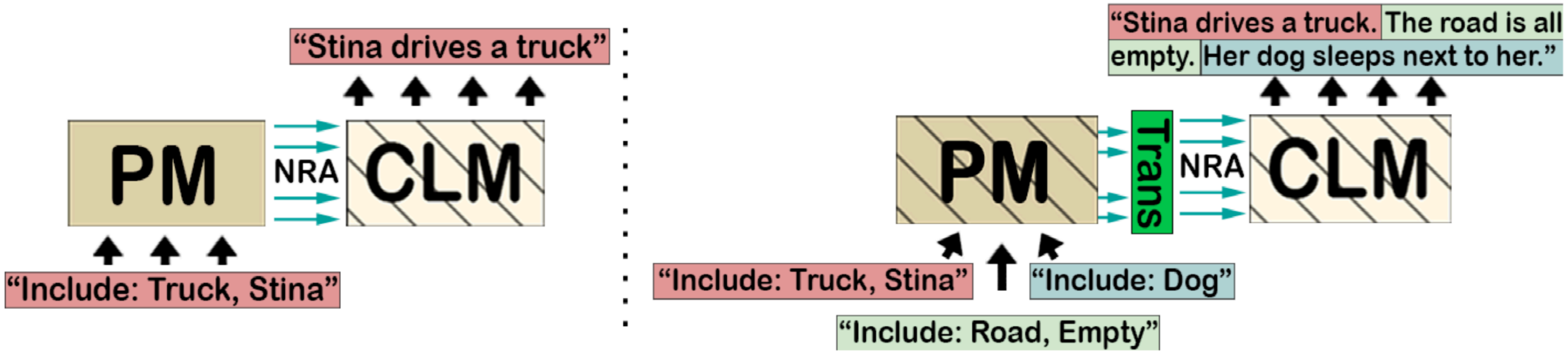
Methods

- A separate model for prompt instructions (PromptModel)

$$KV_P = \text{PromptModel}(S_P)$$

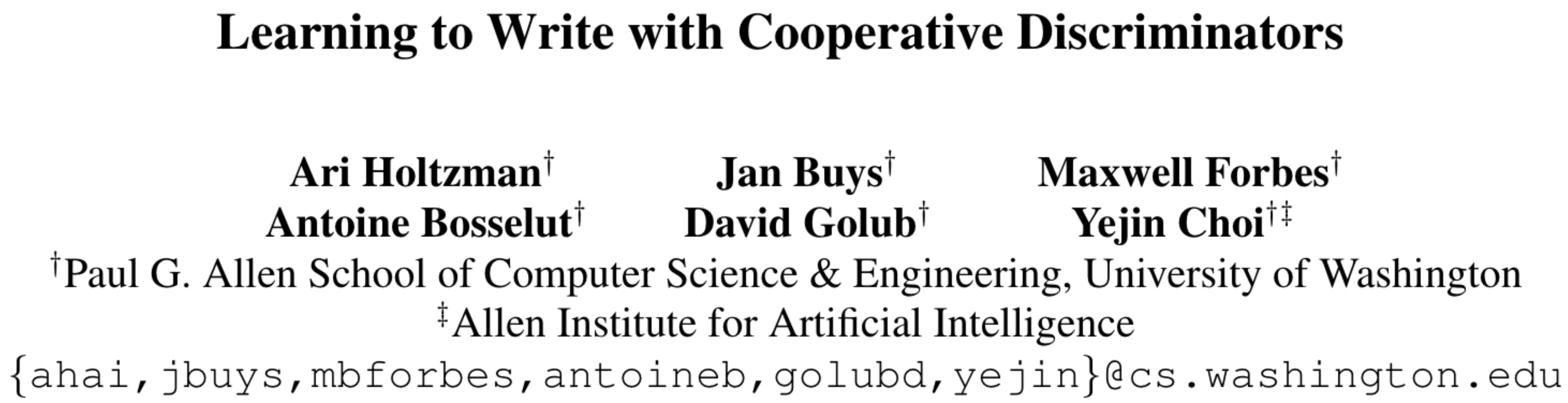
$$\boxed{\begin{aligned} KV_T^n &= \text{CLM}(w_n \mid KV_T^{i<n}) \\ P(w_{n+1}) &= \text{CLM}(w_n \mid KV_p,\ KV_T^{i<n}) \end{aligned}}$$

- Non-Residual Attention (NRA)
  - allow independent prompts at different steps



summary

- Previous work assume the access to attribute-specific data / LMs
    - Can be impractical in scenarios with privacy concerns
- Let's assume access only to the general LM (no class-conditional LM)
    - and pre-trained attribute discriminators



Learning to Write with Cooperative Discriminators

Ari Holtzman[†] Jan Buys[†] Maxwell Forbes[†]
Antoine Bosselut[†] David Golub[†] Yejin Choi[†‡]
[†]Paul G. Allen School of Computer Science & Engineering, University of Washington
[‡]Allen Institute for Artificial Intelligence
{ahai,jbuys,mbforbes,antoineb,golubd,yejin}@cs.washington.edu

ACL2018

Motivation

- Long-form Text Generation: repetitive, self-contradictory, and overly generic
- Grice’s Maxims: cooperative discriminators

$$f_\lambda(\mathbf{x}, \mathbf{y}) = \log(P_{\mathrm{lm}}(\mathbf{y}|\mathbf{x})) + \sum_k \lambda_k s_k(\mathbf{x}, \mathbf{y}),$$

Methods

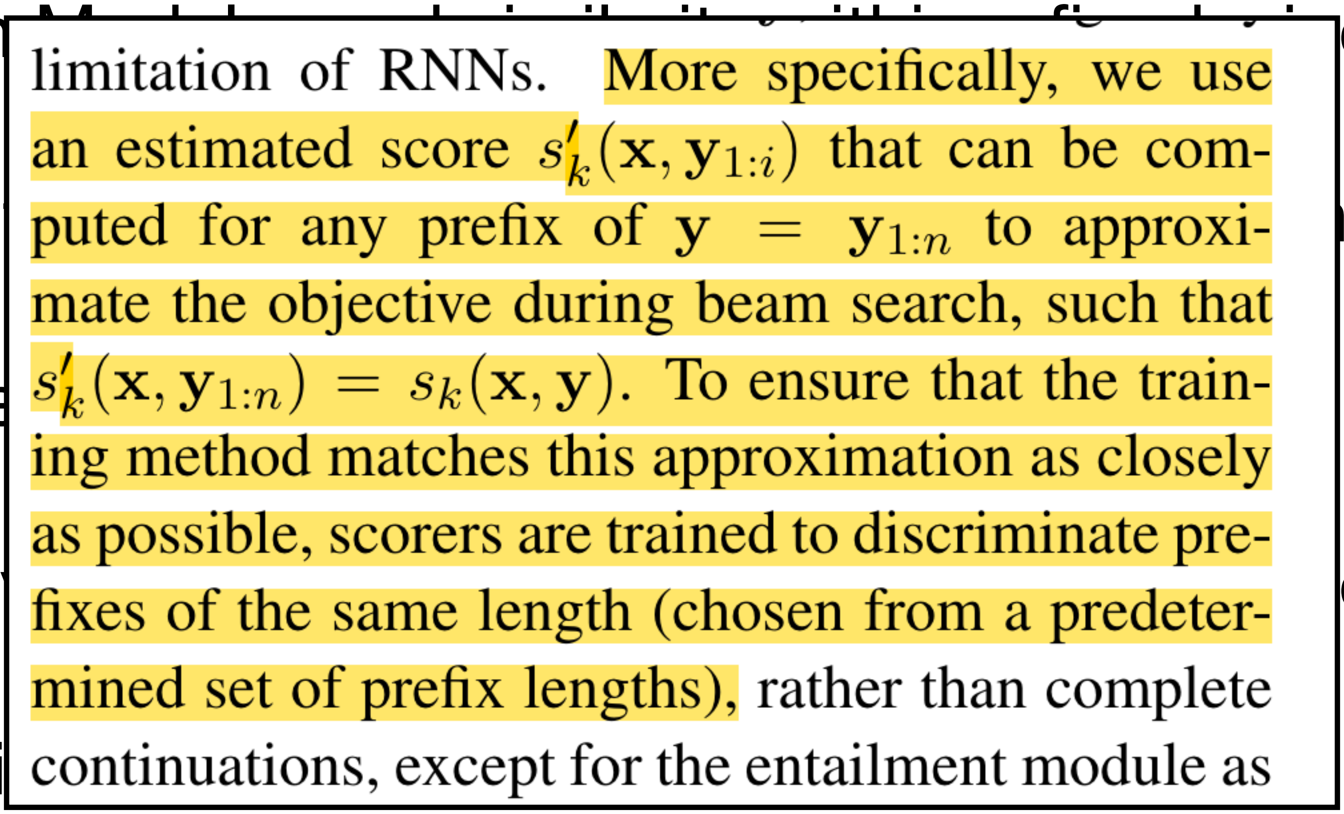
$$f_\lambda(\mathbf{x}, \mathbf{y}) = \log(P_{\mathrm{lm}}(\mathbf{y}|\mathbf{x})) + \sum_k \lambda_k s_k(\mathbf{x}, \mathbf{y}),$$

- 1. Repetition Model: word similarity within a fixed window
- 2. Entailment Model: NLI scores of y against preceding sentences
- 3. Relevance Model: sentence-pair classification
- 4. Lexical Style Model: Bag of words Classification Model
- 1,2,4 are trained using natural sentences as positives and model-generated sentences as negatives, 3 is an off-the-shelf NLI model.

Methods

$$f_\lambda(\mathbf{x}, \mathbf{y}) = \log(P_{\text{lm}}(\mathbf{y}|\mathbf{x})) + \sum_k \lambda_k s_k(\mathbf{x}, \mathbf{y}),$$

- 1. Repetition dow
- 2. Entailmen g sentences
- 3. Relevance
- 4. Lexical St odel
- 1,3,4 are trai and model-generated sentences as negatives, 2 is an off-the-shelf NLI model.



limitation of RNNs. More specifically, we use an estimated score $s'_k(\mathbf{x}, \mathbf{y}_{1:i})$ that can be computed for any prefix of $\mathbf{y} = \mathbf{y}_{1:n}$ to approximate the objective during beam search, such that $s'_k(\mathbf{x}, \mathbf{y}_{1:n}) = s_k(\mathbf{x}, \mathbf{y})$. To ensure that the training method matches this approximation as closely as possible, scorers are trained to discriminate prefixes of the same length (chosen from a predetermined set of prefix lengths), rather than complete continuations, except for the entailment module as

Improving Controllable Text Generation with Position-Aware Weighted Decoding

Yuxuan Gu[†], **Xiaocheng Feng**[†‡], **Sicheng Ma**[†], **Jiaming Wu**[†], **Heng Gong**[†], **Bing Qin**[†‡]
[†]Harbin Institute of Technology [‡] Peng Cheng Laboratory
{yxgu,xcfeng,scma,jmwu,hgong,bqin}@ir.hit.edu.cn

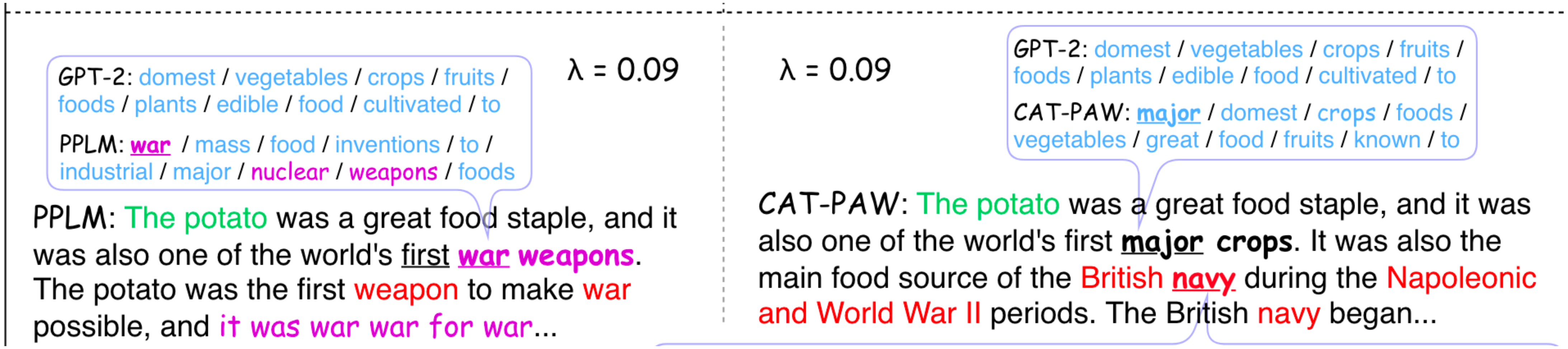
ACL2022

Motivation

$$P(X|a) \propto P(X)P(a|X)^{\lambda} \longrightarrow P(X|a) \propto \prod_{i=1}^{n}\left[P(x_i|x_{<i})P(a|x_{<i})^{\lambda}\right]$$

- Weighted Decoding: λ control the trade-off between control strength and text fluency

- The strength should vary across different positions.



GPT-2: domest / vegetables / crops / fruits / foods / plants / edible / food / cultivated / to

PPLM: war / mass / food / inventions / to / industrial / major / nuclear / weapons / foods

λ = 0.09

PPLM: The potato was a great food staple, and it was also one of the world's first war weapons. The potato was the first weapon to make war possible, and it was war war for war...

λ = 0.09

GPT-2: domest / vegetables / crops / fruits / foods / plants / edible / food / cultivated / to

CAT-PAW: major / domest / crops / foods / vegetables / great / food / fruits / known / to

CAT-PAW: The potato was a great food staple, and it was also one of the world's first major crops. It was also the main food source of the British navy during the Napoleonic and World War II periods. The British navy began...

Method

- Regulator: adjust control strength properly at different positions

$$P(X|a) \propto \prod_{i=1}^{n} \left[P(x_i|x_{<i}) P(a|x_{<i})^{\boxed{\lambda f(a, P(x_{\leq i}))}} \right]$$

- 1. Heuristic Regulator: Amply the signal when it is more likely to generate attribute-relevant words. W^a is a set of keywords for the attribute a

$$t_H = \sum_{w \in W^a} P(x_i = w|x_{<i})$$

$$\begin{aligned} f &= f_H(W^a, P(x_i|x_{<i})) \\ &= t_H/\tau_H, \end{aligned}$$

- 2. Trainable Regulator: train a classifier to estimates the probability of the next token being relevant to attribute a. Supervision is from masking methods for unsupervised style transfer.

$$\begin{aligned} t_T &= \sum_{k=1}^{N} n_k \times P(k|x_{\leq i}) \\ &= \mathbf{n} \cdot \mathrm{softmax}[\mathbf{W} \cdot \mathrm{Attn}(\mathbf{h}_{[1..i]})] \\ f &= f_T(a, P(x_{\leq i})) \\ &= t_T/\tau_T, \end{aligned}$$

FUDGE: Controlled Text Generation With Future Discriminators

Kevin Yang
UC Berkeley
yangk@berkeley.edu

Dan Klein
UC Berkeley
klein@berkeley.edu

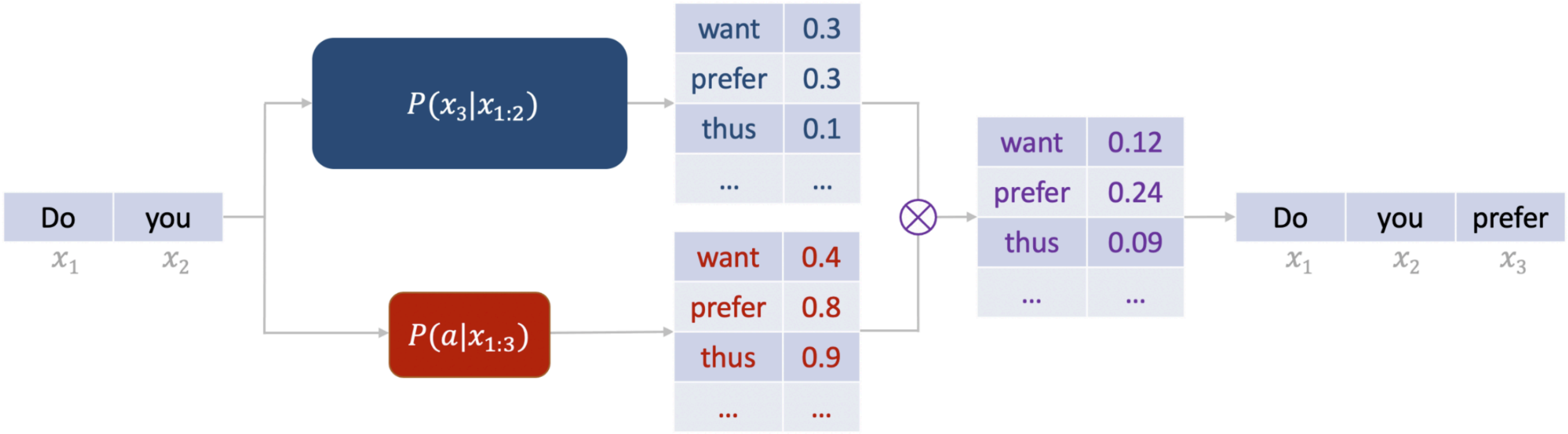
ACL2021

Motivation

$$P(x_i|x_{1:i-1}, a) \propto P(a|x_{1:i})P(x_i|x_{1:i-1})$$

- Weighted Decoding: although the classifier takes a prefix $x_{1:i}$ as input, it should predict whether attribute *a* will in the **future** be satisfied for the completed generation.

Method



want	0.3
prefer	0.3
thus	0.1
…	…

want	0.4
prefer	0.8
thus	0.9
…	…

want	0.12
prefer	0.24
thus	0.09
…	…

Mix and Match: Learning-free Controllable Text Generation using Energy Language Models

Fatemehsadat Mireshghallah[1], **Kartik Goyal**[2], **Taylor Berg-Kirkpatrick**[1]
[1] University of California San Diego, [2] Toyota Technological Institute at Chicago (TTIC)
[fatemeh, tberg]@ucsd.edu, kartikgo@ttic.edu

ACL2022

Motivation

- Discriminators need to be trained on partial generations in order to be operationalized with step-wise autoregressive models

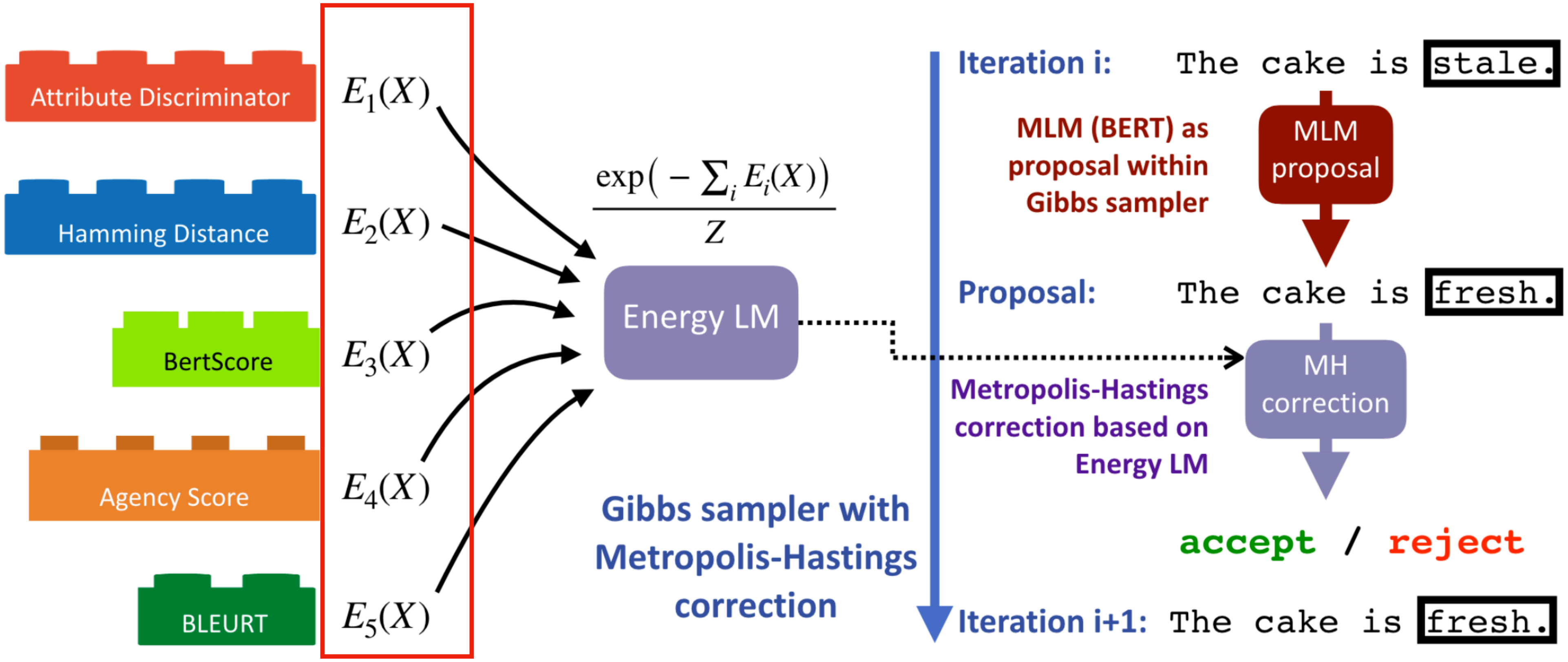
- Many attributes are essentially global.

Methods

- Product of experts as a probabilistic energy model (i.e., **non-autoregressive, globally normalized** LM)

- Gibbs-Metropilis-Hastings sampling

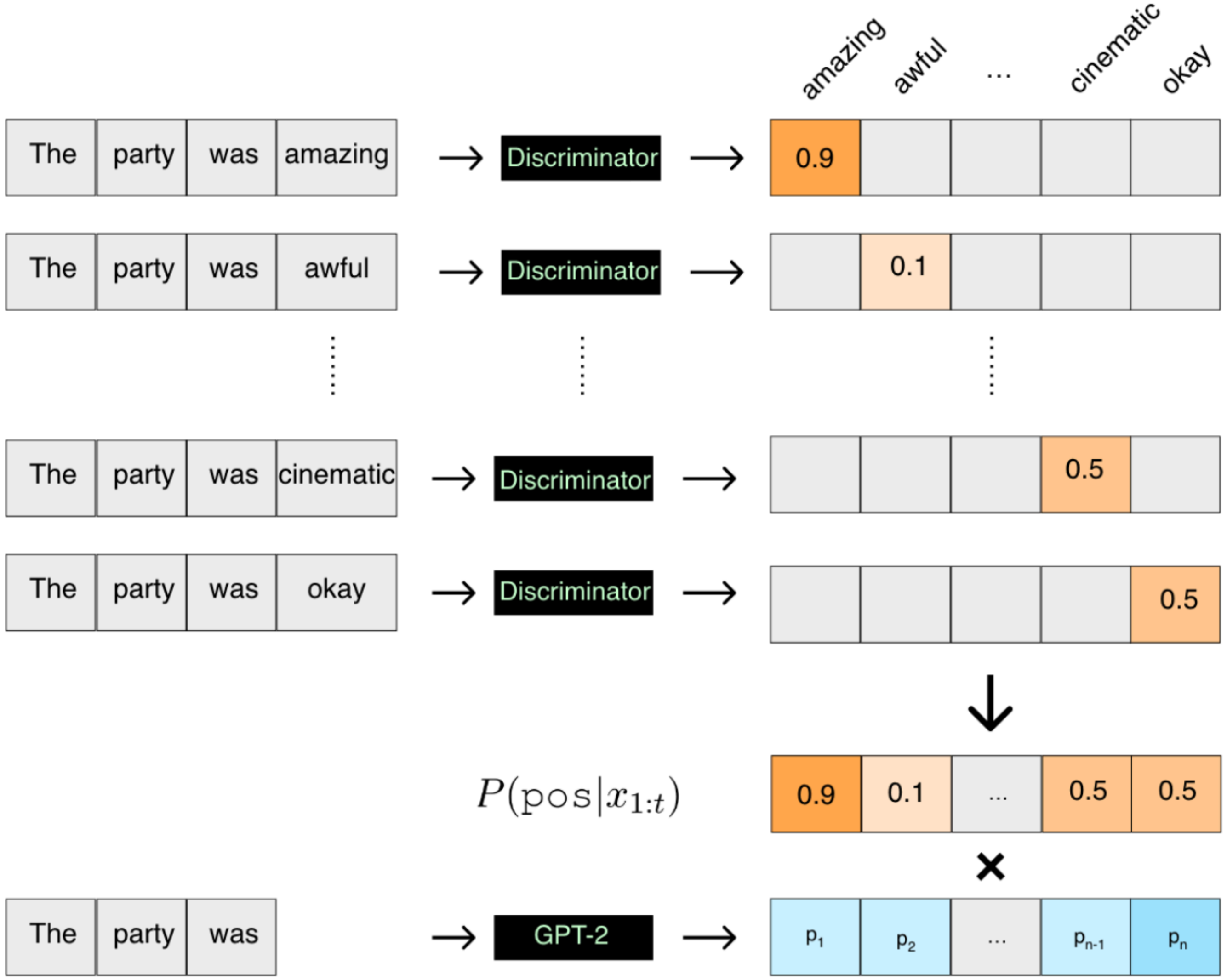
Logits from different models taking the whole sequence as input



Attribute Discriminator
Hamming Distance
BertScore
Agency Score
BLEURT
$E_1(X)$
$E_2(X)$
$E_3(X)$
$E_4(X)$
$E_5(X)$
$\frac{\exp(-\sum_i E_i(X))}{Z}$
Energy LM
Gibbs sampler with Metropolis-Hastings correction
Iteration i: The cake is stale.
MLM (BERT) as proposal within Gibbs sampler
MLM proposal
Proposal: The cake is fresh.
Metropolis-Hastings correction based on Energy LM
MH correction
accept / reject
Iteration i+1: The cake is fresh.

Summary

- Weighted decoding is slow.
  - Feeding candidate next tokens into a discriminator scales **linearly** with the number of tokens to be re-weighted


amazing
awful
...
cinematic
okay
The party was amazing
Discriminator
0.9
The party was awful
Discriminator
0.1
The party was cinematic
Discriminator
0.5
The party was okay
Discriminator
0.5
$P(\text{pos}|x_{1:t})$
0.9
0.1
...
0.5
0.5
×
The party was
GPT-2
p_1
p_2
...
p_{n-1}
p_n