Friendly Conditional Text Generator

Advisor :Jia-Ling, Koh

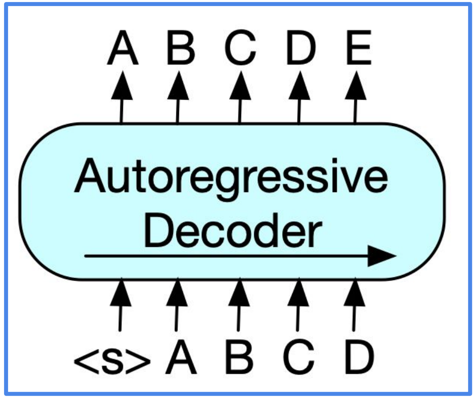
Speaker : Shu-Ming Yu

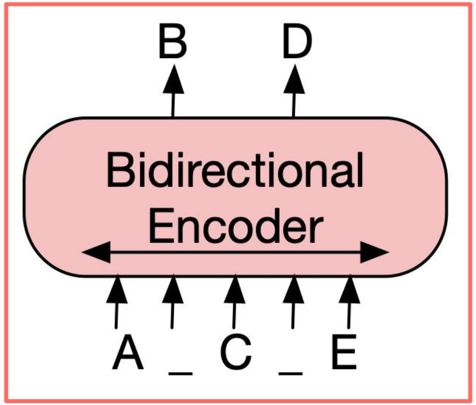
Source : WSDM’ 23

Date : 2024/01/12

Introduction

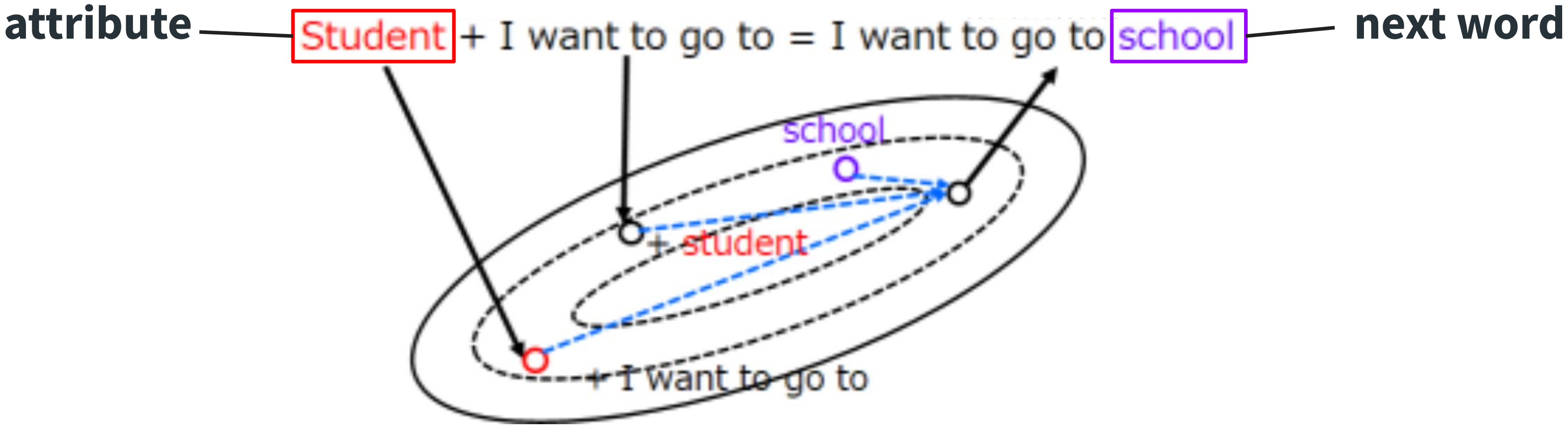
- Text generation
  - Autoregressive(GPT)
  - Encoder-Decoder(BERT)





Introduction

- Conditional text generation



Introduction

- prefix-tuning
  - Fine-tuning needs to updates all parameters which is computationally expensive

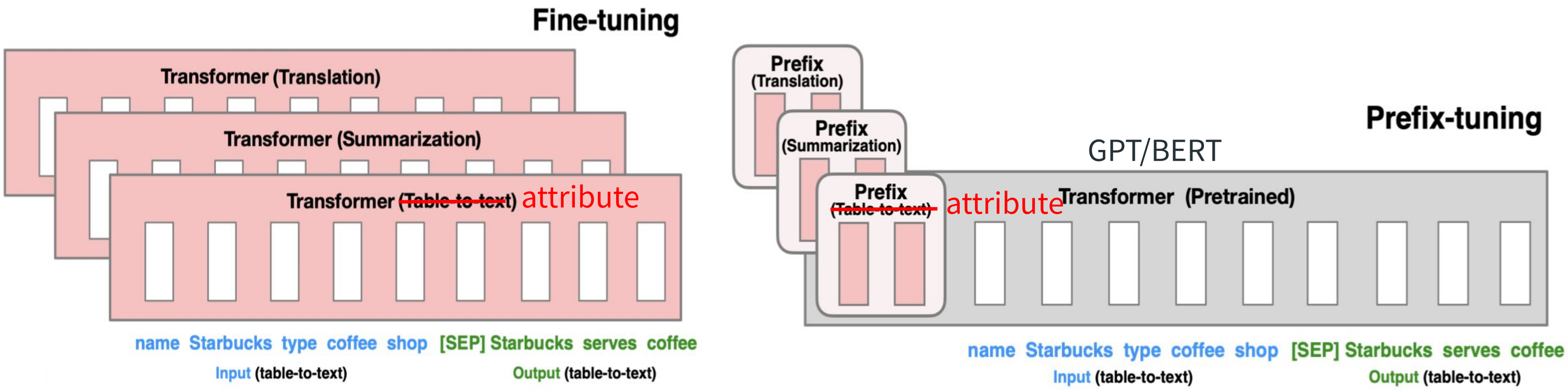
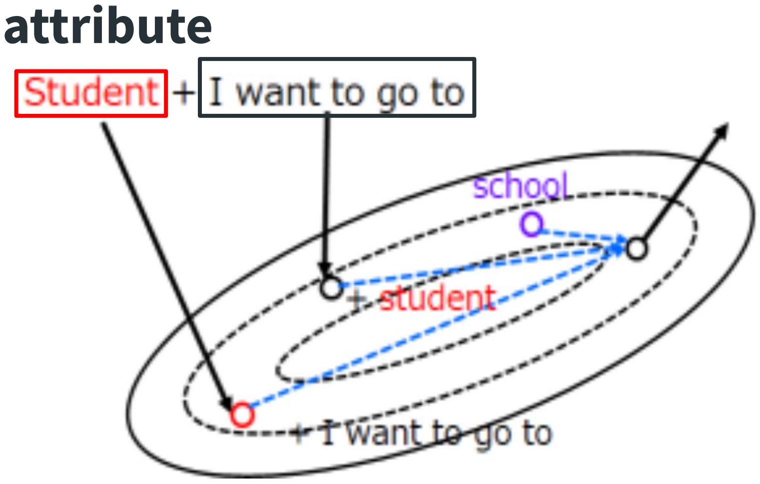
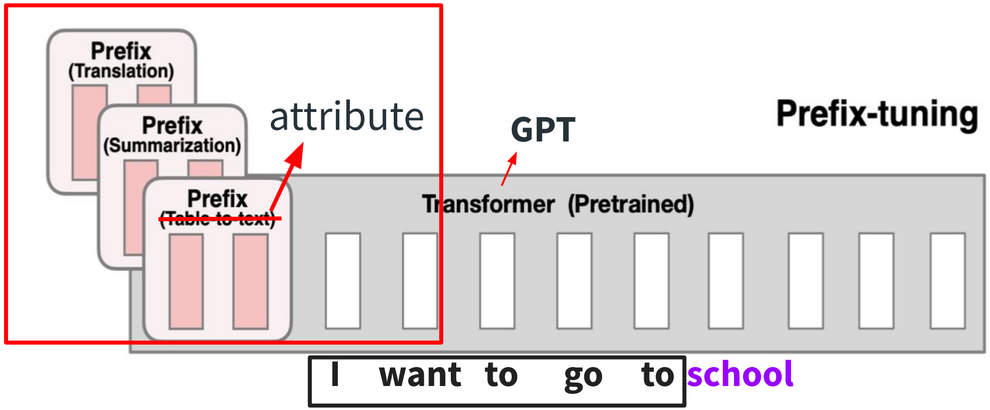


Table-to-text: 對應輸入的table生成相關文字

Introduction

- prefix-tuning





Outline

- Introduction
- **Method**
  - Multi-view attention(MVA)
  - Masked Attribute Modeling(MAM)
  - Attribute Linguistic Matching(ALM)
- Experiment
- Conclusion

Training Input / Output

Training Input:

- Attribute part: attributes
- Linguistic part: words(review/abstract)

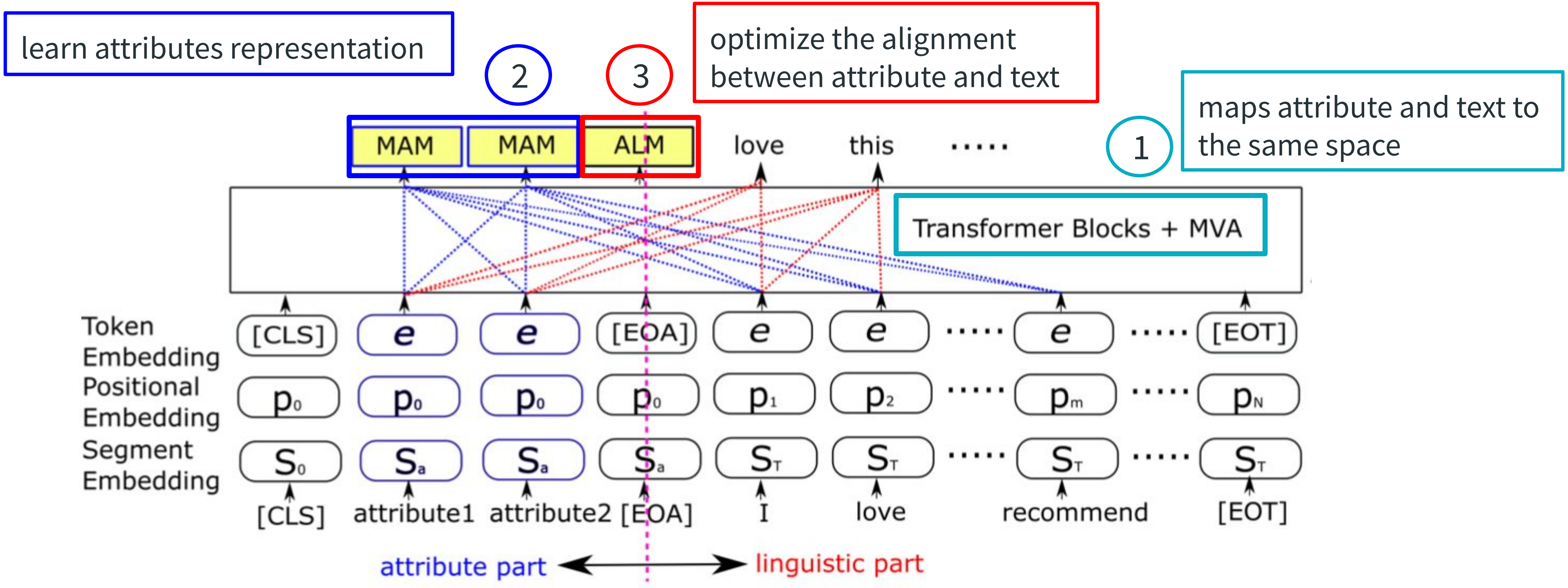
Input to transformer

[CLS] attribute1 attribute2… [EOA] sent1 [SEP] sent2… [EOT]

Output:

- Generated sentences

Method

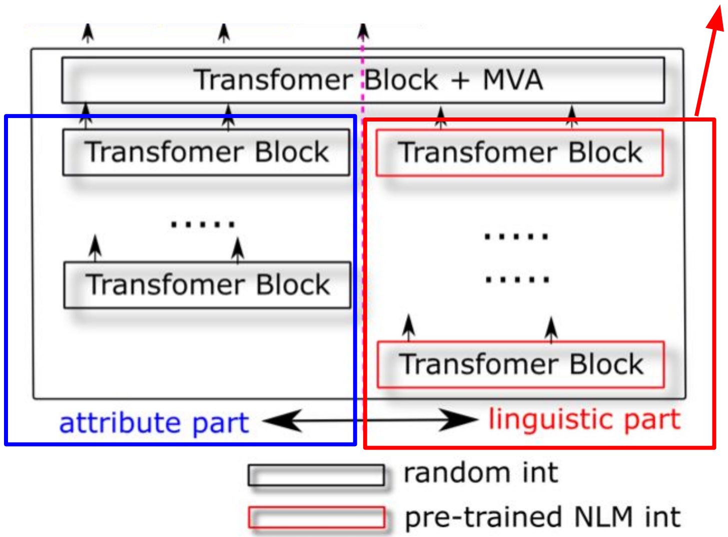

learn attributes representation
2
3
optimize the alignment between attribute and text
maps attribute and text to the same space
1
MAM
MAM
ALM
love
this
.....
Transformer Blocks + MVA
Token Embedding
Positional Embedding
Segment Embedding
[CLS]
e
e
[EOA]
e
e
.....
e
.....
[EOT]
p_0
p_0
p_0
p_0
p_1
p_2
p_m
p_N
S_0
S_a
S_a
S_a
S_T
S_T
S_T
S_T
[CLS]
attribute1
attribute2
[EOA]
I
love
recommend
[EOT]
attribute part
linguistic part

Multi-view attention(MVA)

Parameters will not be optimized

$$\mathbf{Q} = \mathbf{H}_{l-1}\mathbf{W}_l^Q, \mathbf{K} = \mathbf{H}_{l-1}\mathbf{W}_l^K, \mathbf{V} = \mathbf{H}_{l-1}\mathbf{W}_l^V,$$

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V},$$

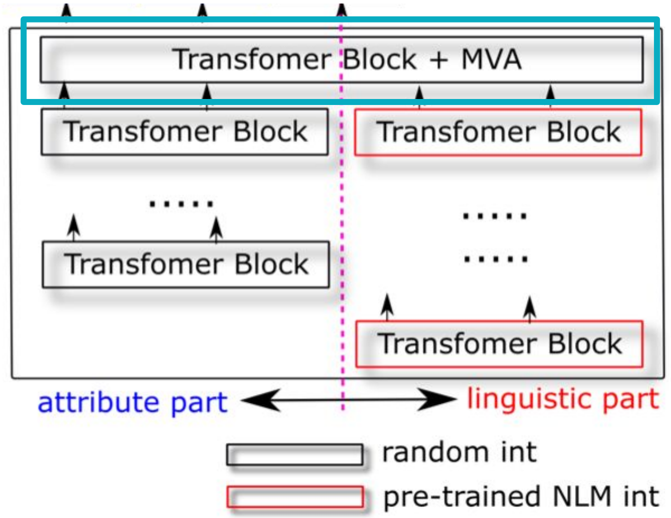


Multi-view attention(MVA)

$$MVA(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{B}_c \otimes softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} + \boxed{\mathbf{M}_c})\mathbf{V},$$

$$+ (1 - \mathbf{B}_c) \otimes softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} + \boxed{\mathbf{M}_s})\mathbf{V},$$

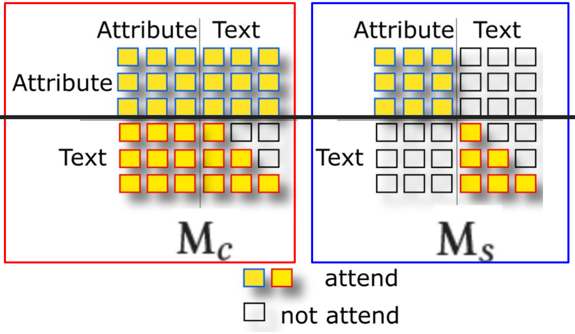
$$\mathbf{B}_c = \sigma(\mathbf{A})\mathbf{I}(\sigma(\mathbf{A}) > \mu), \mathbf{A} = [\mathbf{H}_a \oplus \mathbf{H}_L]\mathbf{W}_b + \mathbf{C}_b,$$



Multi-view attention(MVA)

attribute同時受attribute、text影響

text受attribute及已有的text影響



attribute之間可以互相影響

text只會受到已有的text影響

$$MVA(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{B}_c \otimes softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} + \boxed{\mathbf{M}_c})\mathbf{V},$$

$$+ (\mathbf{1} - \mathbf{B}_c) \otimes softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} + \boxed{\mathbf{M}_s})\mathbf{V},$$

$$\mathbf{B}_c = \sigma(\mathbf{A})\mathbf{I}(\sigma(\mathbf{A}) > \mu), \mathbf{A} = [\mathbf{H}_a \oplus \mathbf{H}_L]\mathbf{W}_b + \mathbf{C}_b,$$

Multi-view attention(MVA)

$$MVA(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{B}_c \otimes softmax(\frac{\mathbf{QK}^T}{\sqrt{d_k}} + \mathbf{M}_c)\mathbf{V},$$

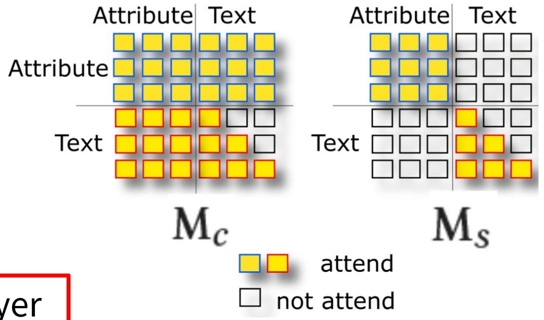
$$+ (\mathbf{1} - \mathbf{B}_c) \otimes softmax(\frac{\mathbf{QK}^T}{\sqrt{d_k}} + \mathbf{M}_s)\mathbf{V},$$

$$\mathbf{B}_c = \sigma(\mathbf{A})\mathbf{I}(\sigma(\mathbf{A}) > \mu), \mathbf{A} = [\mathbf{H}_a \oplus \mathbf{H}_L]\mathbf{W}_b + \mathbf{C}_b,$$

attribute part's top layer

concat

linguistic part's top layer



If $\sigma(\mathbf{A}) > \mu$: $\mathbf{B}_c = \sigma(\mathbf{A})$

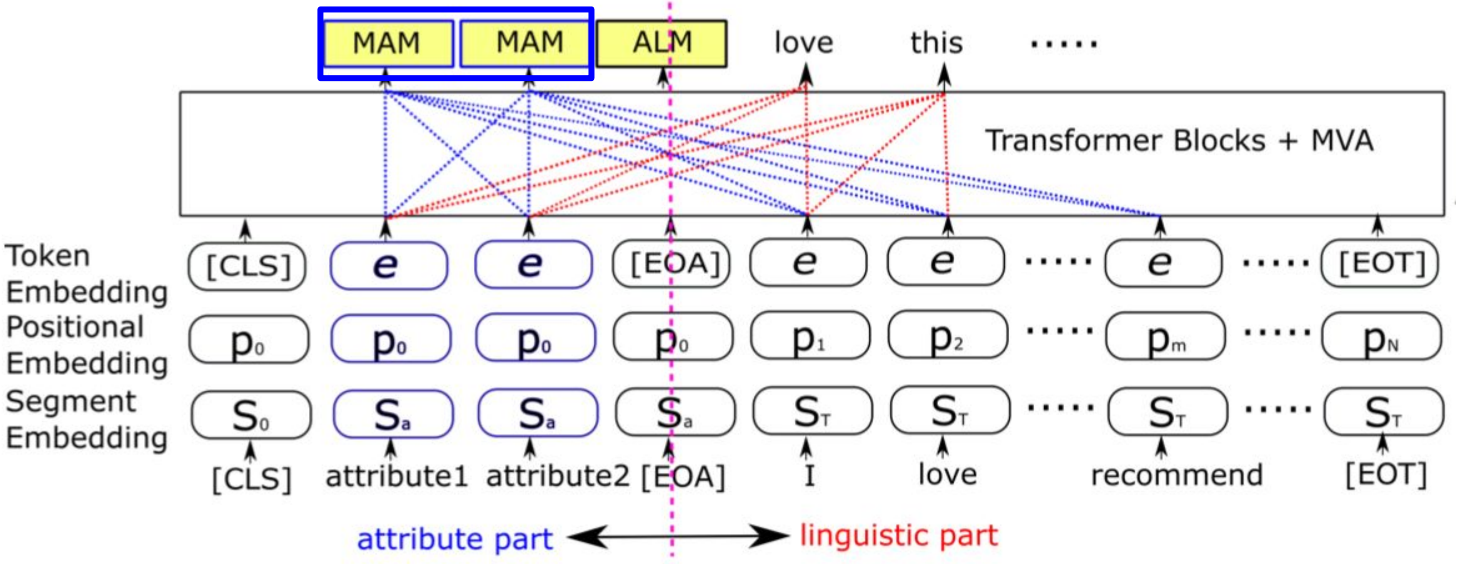
else : $\mathbf{B}_c = \mathbf{0}$

=> only $\mathbf{M}_s$

Method

learn attributes representation

2



Masked Attribute Modeling(MAM)

- To optimize attribute representation, utilize the idea of predicting the randomly masked 15% attributes $\mathbf{a_j}$

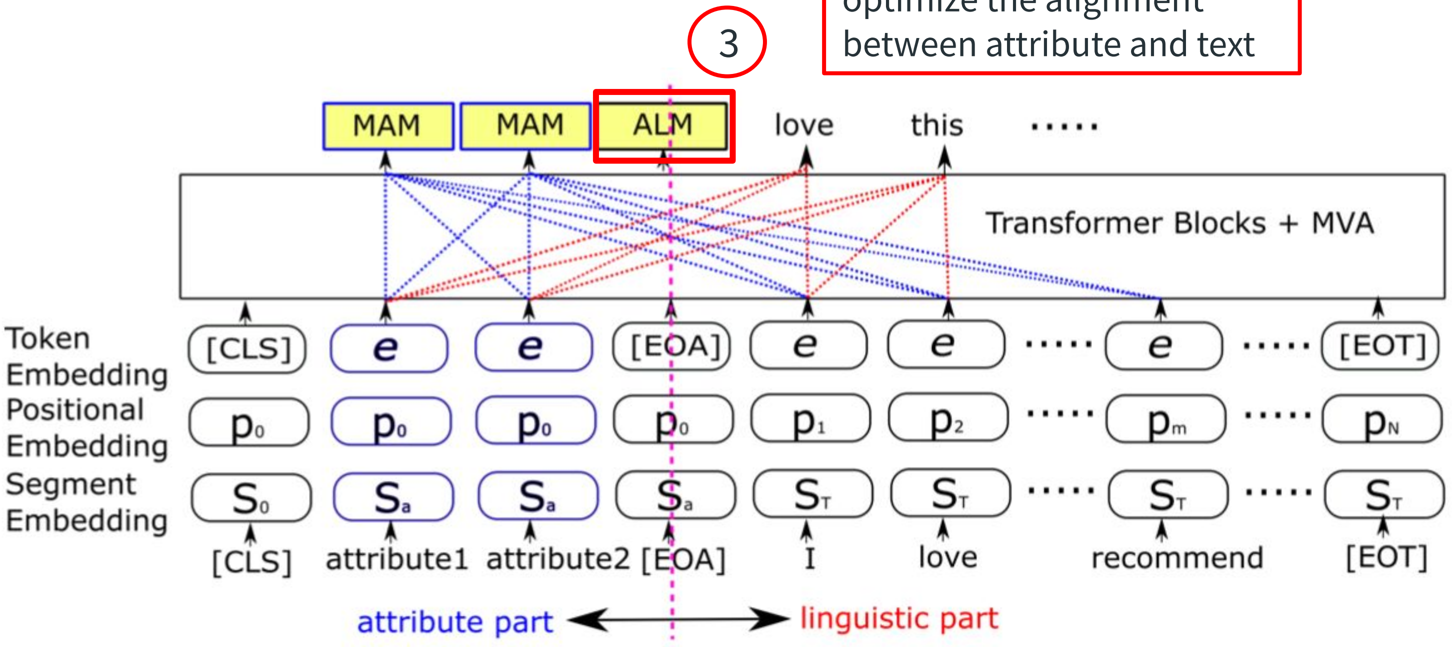
attribute input: $\mathbf{a_j} = \{\mathbf{a_{j,1}}, \cdots, \mathbf{a_{j,|A|}}\}$ **linguistic input:** $\mathbf{w_j} = \{\mathbf{w_{j,1}}, \cdots, \mathbf{w_{j,|X|}}\}$

$$\mathcal{L}_{MAM}(\zeta) = -E_{(a_{j,m},\mathbf{w_j})\sim \mathrm{D}} \log P_{\zeta}(a_{j,m}|\mathbf{a_{j,\backslash m}}, \mathbf{w_j})$$

whole training set

Method

optimize the alignment
between attribute and text



Attribute Linguistic Matching(ALM)

- Text should be semantically aligned with the corresponding attributes

the attribute and text vector pairs $< \mathbf{v}_{|A|,i}, \mathbf{v}_{|T|,i} >$

contrastive(CN)

$$\mathcal{L}_{ALM}(\zeta) = -\sum_{b=1}^{\mathbf{B}} \sum_{i \in b} \log \frac{\exp(\mathbf{v}_{|A|,i} \cdot \mathbf{v}_{|T|,i} / \tau)}{\sum_{k \in b \setminus j} \exp(\mathbf{v}_{|A|,i} \cdot \mathbf{v}_{|T|,k} / \tau)},$$

triplet(TP)

$$\mathcal{L}_{ALM}(\zeta) = \sum_{b=1}^{\mathbf{B}} \max_{(\mathbf{v}_a, \mathbf{v}_p, \mathbf{v}_n) \sim b} (||\mathbf{v}_a - \mathbf{v}_p|| - ||\mathbf{v}_a - \mathbf{v}_n|| + \epsilon, 0),$$

minimum triplet(MTP)

$$\mathcal{L}_{ALM}(\zeta) = \sum_{b=1}^{\mathbf{B}} \max_{(\mathbf{v}_a, \mathbf{v}_p, \mathbf{v}_n) \sim b} (||\mathbf{v}_a - \mathbf{v}_p|| - \boxed{||\mathbf{v}_a - \mathbf{v}_n||} + \epsilon, 0),$$

Only select the minimum distance sample

Prediction loss

$$\mathcal{L}_{FCTG}(\theta) = -\sum_{i=1}^{|D|}\sum_{t=1}^{|\mathbf{x}_i|} \log P_\theta(x_{i,t}|\underline{\mathbf{x}_{i,1:t-1}}, \underline{\mathbf{c}_{i,1:a}})$$

已有的text

所有condition(attribute)

$$\mathcal{X} = LayerNorm(\mathbf{H}_{MVA})\mathbf{W}_c, \quad \mathbf{W}_c \in \mathbb{R}^{d_h \times \boxed{V}}$$

size of vocabulary

$$p(w_i) = \frac{exp(x_i/T)}{\sum_i exp(x_i/T)}, \quad x_i \in \mathcal{X}$$

Next text chosen by sampling on a multinomial distribution at the top-k tokens.

Loss

1 0.1 0.1

$$\mathcal{L} = \lambda_F \mathcal{L}_{FCTG}(\theta) + \lambda_M \mathcal{L}_{MAM}(\zeta) + \lambda_A \mathcal{L}_{ALM}(\zeta),$$

prediction loss

Outline

- Introduction
- Method
- **Experiment**
- Conclusion

Experiment

- **Human Evaluation**
  - Fluency(1~5)

Experiment

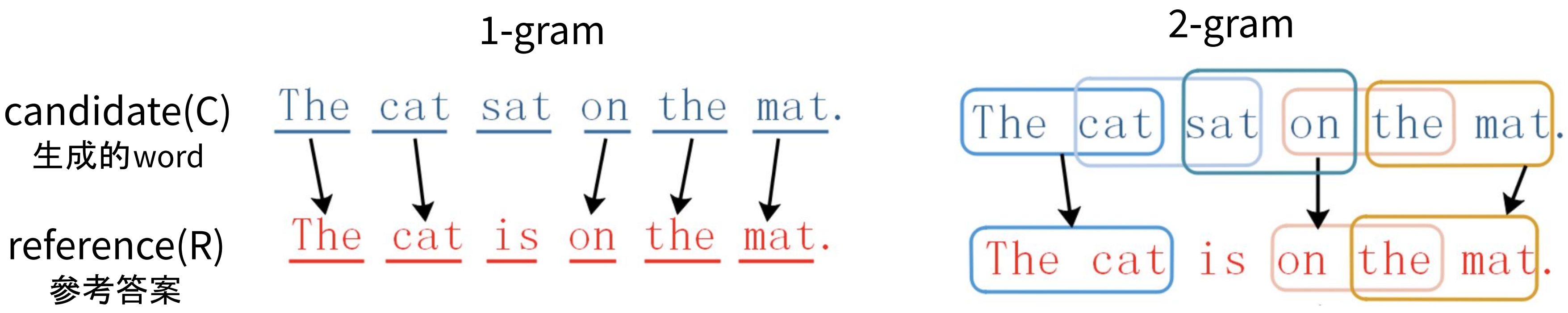
- **Automated Evaluation**
  - BLEU(B-4)
  - ROUGE-L
  - METEOR
  - perplexity

Experiment

- **Automated Evaluation**
  - BLEU(B-4)

Experiment

- **N-gram**



Experiment

- **Evaluation**
  - BLEU as **precision**

$$BLEU = BP \cdot exp(\sum_{n=1}^{N} w_n log \boxed{p_n})$$

Candidate中的uni-gram有幾個

也出現在reference中

$$p_n = \frac{\boxed{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}}{\boxed{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}}.$$

candidate中的uni-gram有幾個

1-gram

candidate(C)
生成的word

The cat sat on the mat.

reference(R)
參考答案

The cat is on the mat.

$$p_n = \frac{5}{6}$$

Experiment

- **Evaluation**
  - BLEU

$$BLEU = BP \cdot exp(\sum_{n=1}^{N} w_n log\, p_n)$$

$$p_n = \frac{\sum_{C\in\{Candidates\}} \sum_{n\text{-}gram\in C} \boxed{Count_{clip}(n\text{-}gram)}}{\sum_{C'\in\{Candidates\}} \sum_{n\text{-}gram'\in C'} Count(n\text{-}gram')}.$$

$$Count_{clip} = \min(\underbrace{Count}_{\text{candidate中這個uni-gram出現的次數}}, \underbrace{Max\_Ref\_Count}_{\text{所有reference 中這個uni-gram出現最多的次數}})$$

candidate中這個uni-gram出現的次數

所有reference 中這個uni-gram出現最多的次數

1-gram

candidate(C) The the the
生成的word

reference(R) The cat is on the mat.
參考答案

$$p_n = \frac{\cancel{3}\ 2}{3}$$

$Count_{clip}$ = min(3, max(2))
= 2

Experiment

- **Evaluation**
  - BLEU

$$BLEU = \boxed{BP} \cdot exp(\sum_{n=1}^{N} w_n log\, p_n)$$

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}.$$

1-gram

candidate(C)
生成的word

The cat sat on the mat.

reference(R)
參考答案

The cat is on the mat.

P1 = 5/6

$$BP = \begin{cases} 1 & lc > lr \\ \exp(1 - lr/lc) & lc \leq lr \end{cases}$$

> 1

< 1

$lc =$ 生成的word的長度

$lr =$ 最短的參考答案的長度

Experiment

Candidate : 生成的word

Reference : 參考答案

- **Evaluation**
  - ROUGE-L

LCS : longest common subsequence

n: len(candidate)

m : len(reference)

$$R_{lcs} = \frac{LCS(X,Y)}{m} \qquad \frac{5}{7}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \qquad \frac{5}{6}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \qquad \frac{10}{13}$$

$\beta = 1$

1-gram

candidate(C) 生成的word: The cat sat on the mat.

reference(R) 參考答案: The **cute** cat is on the mat.

Candidate : 生成的word

Experiment

Reference : 參考答案

- **Evaluation**
  - METEOR

$$METEOR = (1 - pen) \times F_{means}$$

$$F_{means} = \frac{PR}{\alpha P + (1-\alpha)R}$$

$\alpha = 0.5$

=> F_{means} as F-1

P : $\dfrac{\text{Candidate的uni-gram也出現在reference中的數量}}{\text{Candidate的長度}}$ as **precision**

R : $\dfrac{\text{Candidate的uni-gram也出現在reference中的數量}}{\text{reference的長度}}$ as **recall**

Candidate : 生成的word

Reference : 參考答案

Experiment

- **Evaluation**
  - METEOR

$$Pen = \frac{\#chunks}{m}$$

m : number of match

$$METEOR = (1 - pen) \times F_{means}$$

$$Pen = \frac{2}{5}$$

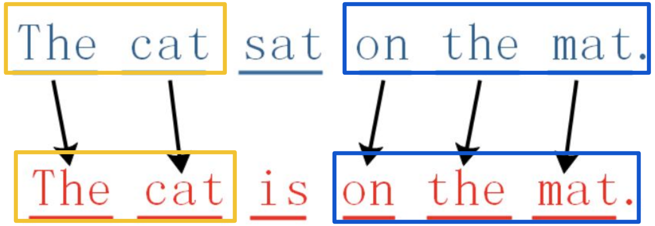
$$F_{means} = \frac{PR}{\alpha P + (1 - \alpha) R}$$

P : precision
R : recall

1-gram

candidate(C)
生成的句子

reference(R)
參考答案



Experiment

- **Automated Evaluation**
  - BLEU(B-4) as **precision**
  - ROUGE-L
  - METEOR as **F1**
  - perplexity

PPL越低代表language model的效果越好，生成出的句子越貼近 training data的用句

$$\mathrm{PPL}\left(\mathbf{W}\right) = \exp\left[-\frac{1}{t}\sum_{i}^{t}\log p_{\theta}\left(w_i | w_{<i}\right)\right]$$

Experiment

- **Dataset**
  - Amazon (At least 20 reviews)

    Input: user, item

    Output: review
  - arXiv

    Input: paper's title

    Output: abstract

		user item	
Dataset	#texts	#attributes	#vocabulary
Amazon	210,000	2,311+2,381	246,534
arXiv	1,506,500	25,112	565,762

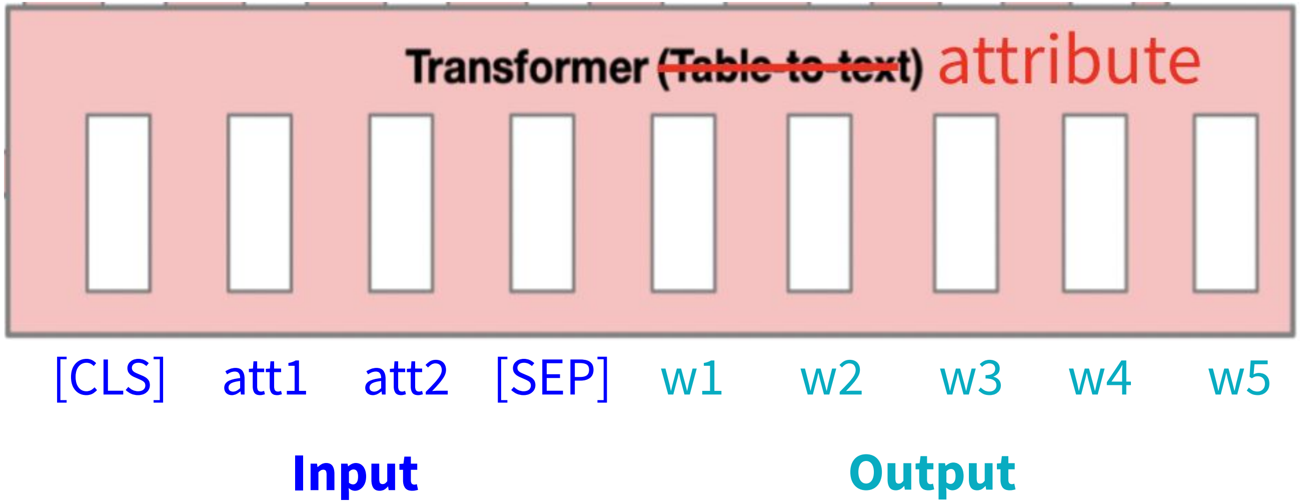
number of word in title

Experiment

- **Baseline**
  - GPT-2
  - Prefix
  - NRP

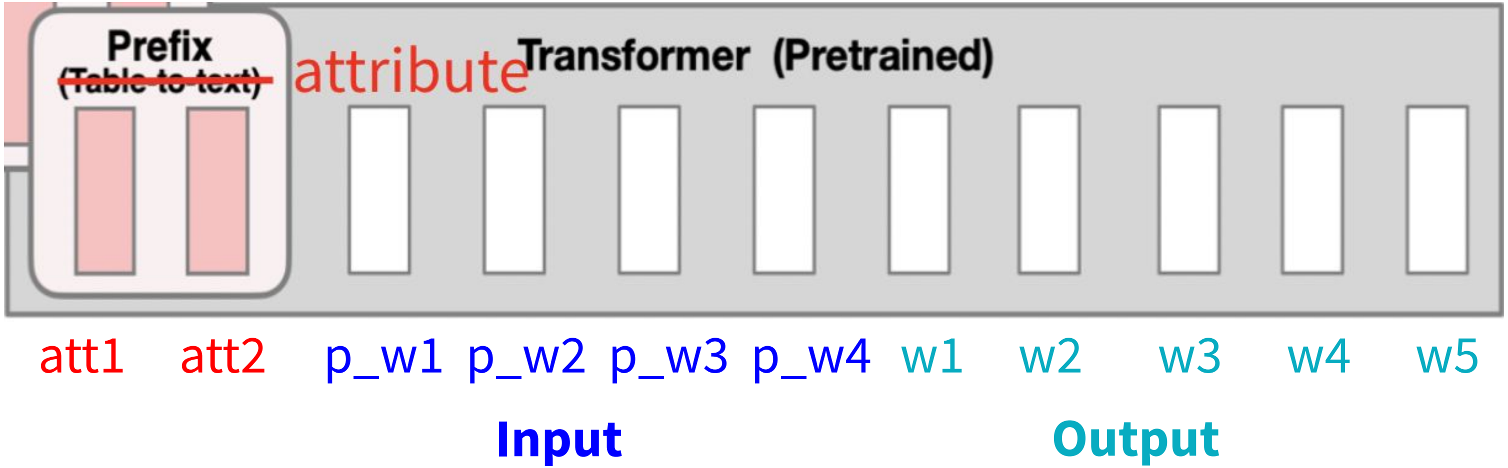
Experiment

- **Baseline**
  - GPT-2



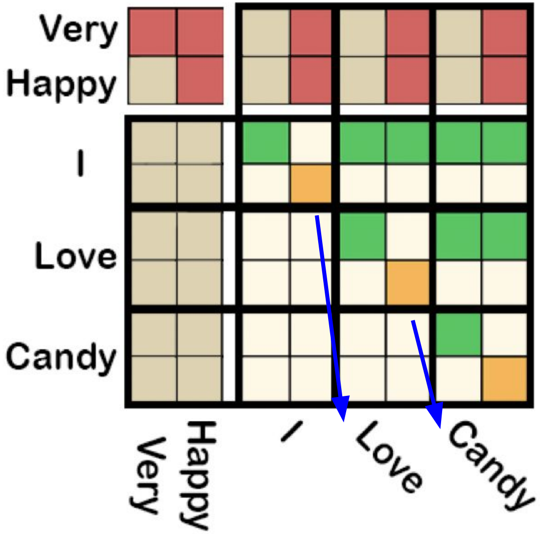
Experiment

- **Baseline**
  - Prefix



Experiment

- **Baseline**
  - NRP



Green : only textual states

Red : prompt model's states

Yellow : textual+prompt states

Experiment

Model	Amazon							arXiv						
	Flu	PPL	B-4	Meteor	Rouge-L			Flu	PPL	B-4	Meteor	Rouge-L		
					P	R	F1					P	R	F1
	↑	↓	↑	↑	↑	↑	↑	↑	↓	↑	↑	↑	↑	↑
GPT-2	3.16	16.67	0.12	0.19	0.09	0.08	0.08	3.16	7.24	0.10	0.28	0.19	0.29	0.23
Prefix [26]	3.19	16.21	0.14	0.20	0.09	0.09	0.09	3.03	7.12	0.11	0.29	0.21	0.22	0.21
NRP [6]	3.19	15.86	0.13	0.21	0.09	0.09	0.09	3.03	7.02	0.11	0.30	0.23	0.22	0.22
FCTG	**3.95**	**13.83**	**0.26**	**0.28**	**0.14**	**0.13**	**0.13**	**3.78**	**2.70**	**0.14**	**0.31**	**0.31**	**0.28**	**0.29**

Experiment

CE: cross-entropy
CN: constructive
TR: triplet
MTR: minimum triplet

Data components			Amazon						arXiv					
			PPL	B-4	Meteor	Rouge-L			PPL	B-4	Meteor	Rouge-L		
MVA	MAM	ALM	↓	↑	↑	P ↑	R ↑	F1 ↑	↓	↑	↑	P ↑	R ↑	F1 ↑
$\mu = 0.8$	CE	TR	14.91	0.26	0.26	0.15	0.11	0.13	3.28	0.13	0.30	0.29	0.27	0.28
$\mu = 0.5$	CE	TR	14.88	0.27	0.26	0.14	0.12	0.13	3.24	0.13	0.30	0.28	0.28	0.28
$\mu = 0$	CE	CN	14.56	0.26	0.27	0.14	0.13	0.13	3.23	0.13	0.30	0.29	0.28	0.28
$\mu = 0$	CE	TR	**13.83**	**0.26**	**0.28**	**0.14**	**0.13**	**0.13**	**2.70**	**0.14**	**0.31**	**0.31**	**0.28**	**0.29**
$\mu = 0$	CE	MTR	14.38	0.25	0.27	0.14	0.13	0.13	3.46	0.13	0.30	0.29	0.28	0.28
$\mu = 0$	w/o	TR	14.22	0.22	0.26	0.13	0.11	0.12	3.83	0.13	0.30	0.27	0.28	0.28
w/o	TR	TR	14.84	0.19	0.24	0.11	0.11	0.11	4.32	0.12	0.29	0.25	0.27	0.26

Experiment

CE: cross-entropy
CN: constructive
TR: triplet
MTR: minimum triplet

Data components			Amazon						arXiv					
			PPL	B-4	Meteor	Rouge-L			PPL	B-4	Meteor	Rouge-L		
MVA	MAM	ALM	↓	↑	↑	P ↑	R ↑	F1 ↑	↓	↑	↑	P ↑	R ↑	F1 ↑
$\mu = 0$	CE	CN	14.56	0.26	0.27	0.14	0.13	0.13	3.23	0.13	0.30	0.29	0.28	0.28
$\mu = 0$	CE	TR	**13.83**	**0.26**	**0.28**	**0.14**	**0.13**	**0.13**	**2.70**	**0.14**	**0.31**	**0.31**	**0.28**	**0.29**
$\mu = 0$	CE	MTR	14.38	0.25	0.27	0.14	0.13	0.13	3.46	0.13	0.30	0.29	0.28	0.28

Experiment

- Compared to other models, the required training time for it has significantly decreased

	GPT2+p	NRP	FCTG
Amazon	6.8m	7.3m	$\boxed{3.5m}$
arXiv	8.7m	11.5m	$\boxed{4.8m}$

Conclusion

- FCTG’s framework allows attributes to effectively influence the generated text and significantly reduces computation costs