XLNet: Generalized Autoregressive Pretraining for Language Understanding

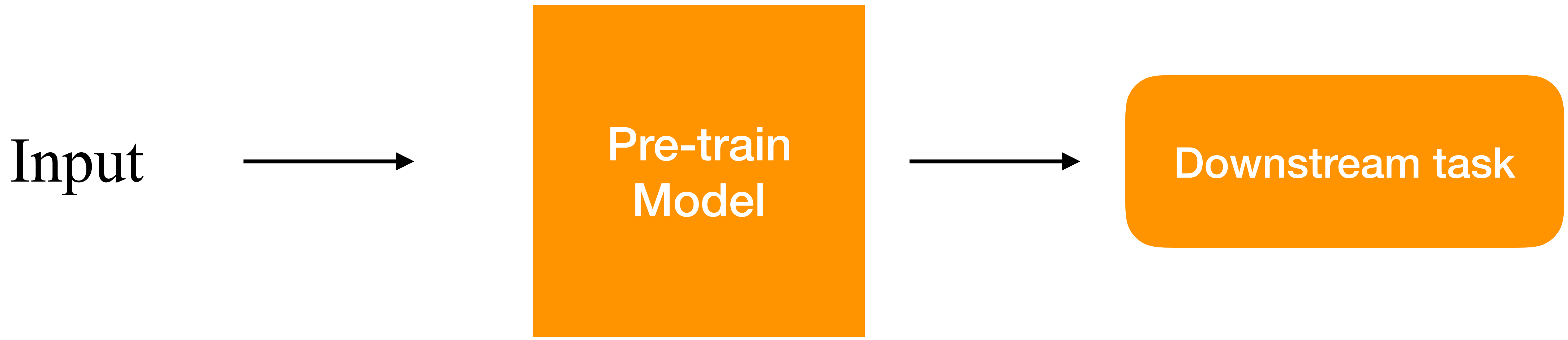
Zhilin Yang[*1], **Zihang Dai**[*12], **Yiming Yang**[1], **Jaime Carbonell**[1],
Ruslan Salakhutdinov[1], **Quoc V. Le**[2]
[1]Carnegie Mellon University, [2]Google Brain

Published Date: Jun 2019

WenWei

- Introduction
- Related Work
- Permutation Language Modeling
- Two-Stream Self-Attention
- Long Text Understanding
- Evaluation

Introduction



- Sequence (Text)
- Image
- Videos

Transfer Learning

NLP:
- ELMo
- GPT(GPT2)
- BERT
- ERNIE

CV:
- VGG
- ResNet
- Inception
- MobileNet
- NASNet
- …

NLP:
- Text classification
- Semantic classification
- Question Answering
- …

CV:
- Image classification
- Image detection
- …

Hybrid:
- Image(Video) caption
- Visual Question Answering
- …

Related Work

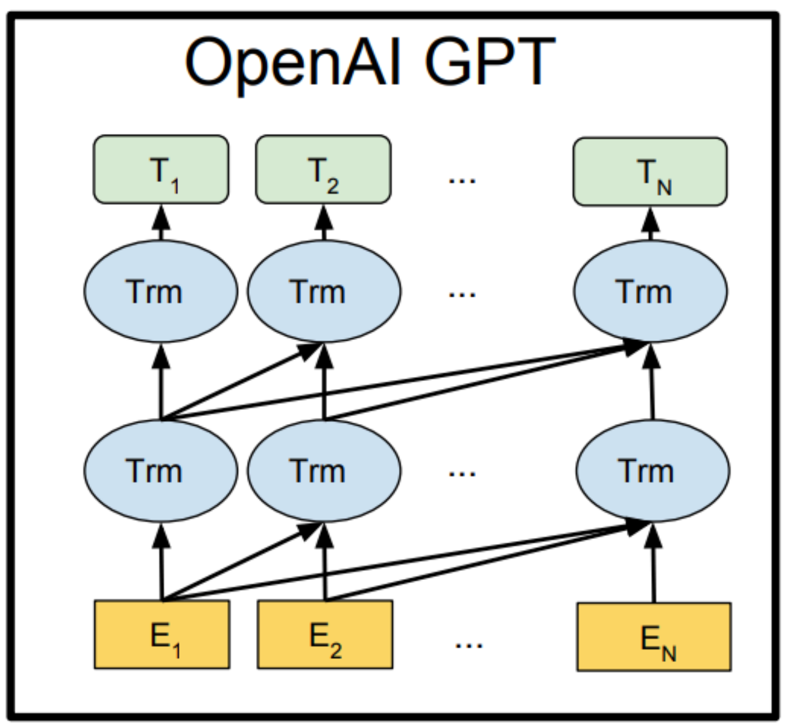
- AutoRegressive(AR): Target is only conditioned on the token up to position **t**.
  - Given a sequence $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5)$

$$\prod p(x_t \mid x_{<t}) = p(x_5 \mid x_4, x_3, x_2, x_1) \times p(x_4 \mid x_1, x_2, x_3) \times p(x_3 \mid x_1, x_2) \times p(x_2 \mid x_1) \times p(x_1)$$

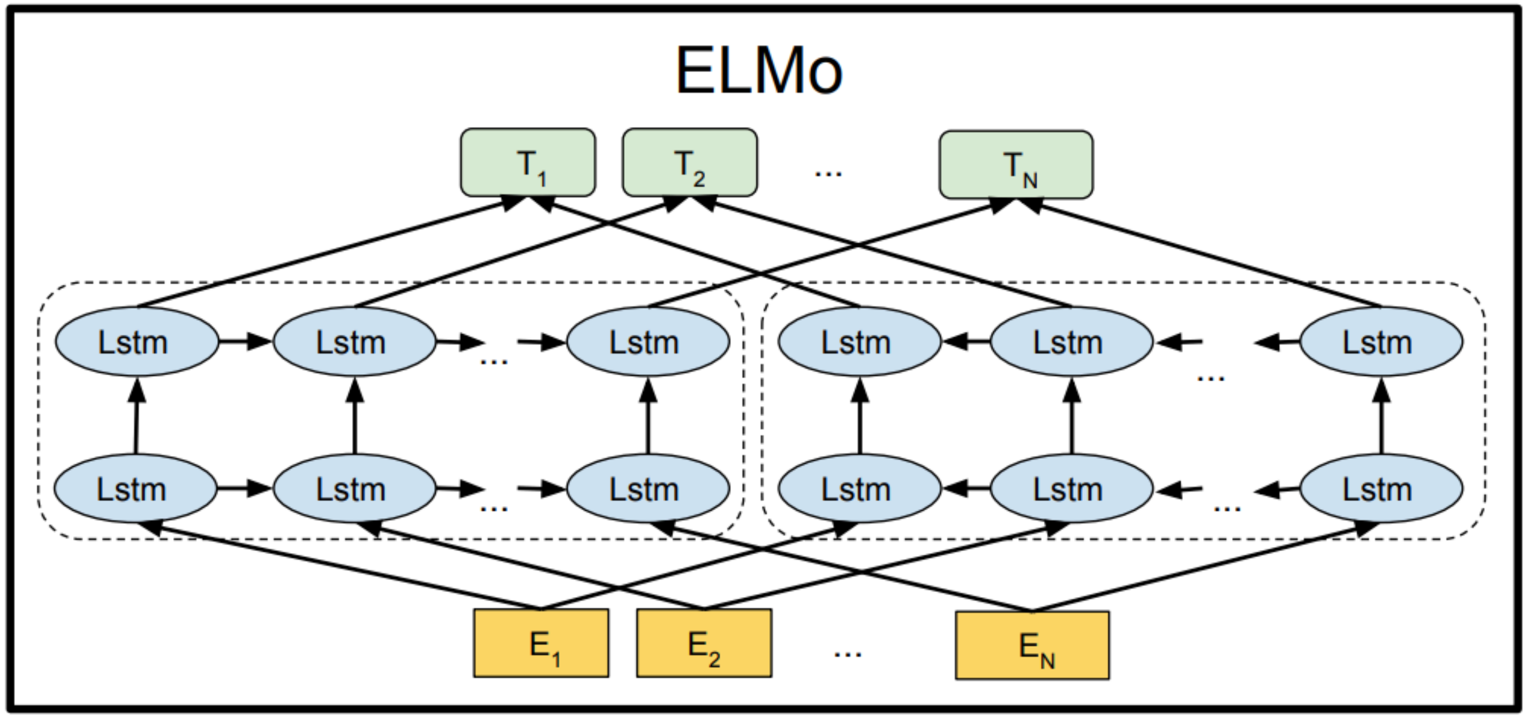
  - Maximum the likelihood

$$\max_{\theta} \quad \log p_\theta(\mathbf{x}) = \sum_{t=1}^{T} \log p_\theta(x_t \mid \mathbf{x}_{<t}) = \sum_{t=1}^{T} \log \frac{\exp(h_\theta(\mathbf{x}_{1:t-1})^T e(x_t))}{\sum_{x'} \exp(h_\theta(\mathbf{x}_{1:t-1})^T e(x'))}$$

GPT



ELMo



Source: https://arxiv.org/pdf/1810.04805.pdf

Related Work

Given a sequence $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5)$

- AutoRegressive(AR):

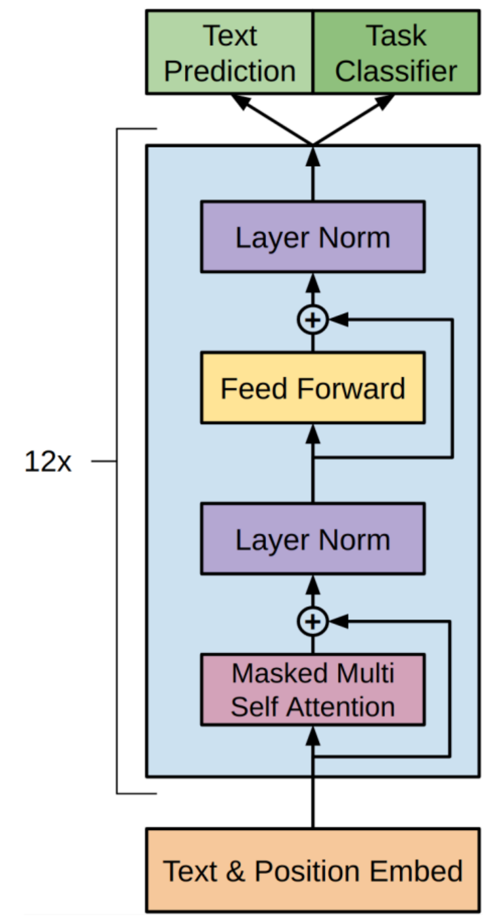
Forward: Given $x_1, x_2 \rightarrow$ Predict x_3, x_4

$$p(x_3 \mid x_1, x_2) \times p(x_4 \mid x_1, x_2, x_3)$$

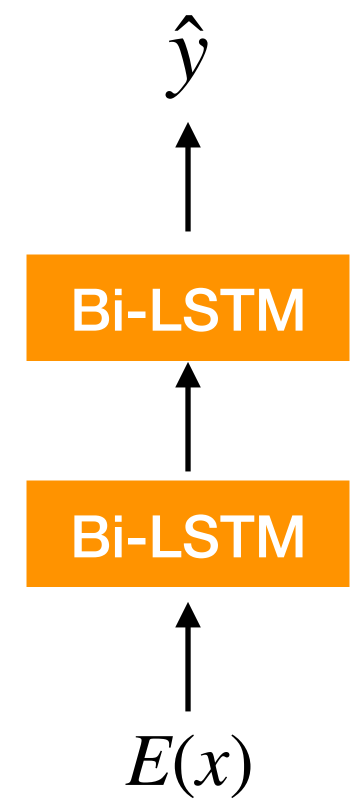
Backward: Given $x_5, x_4 \rightarrow$ Predict x_3, x_2

$$p(x_3 \mid x_5, x_4) \times p(x_2 \mid x_5, x_4, x_3)$$

GPT



ELMo(Bi-LSTM)



Issue:

- ***Context dependency***

Passage: Thom Yorke is the singer of Radiohead.

AR:

$$p(\text{is} \mid \text{Thom Yorke}) \times \ldots \times$$
$$p(\text{singer} \mid \text{Thom York is the}) \times \ldots \times$$
$$p(\text{Radiohead} \mid \text{Thom York is the singer of})$$

Question: What band is Thom Yorke in?
Answer: Radiohead.

Question: Who is the singer of Radiohead?
Answer: Thom Yorke.

XLNet:

$$p(\text{Thom Yorke} \mid \text{is the singer of Radiohead}) \times \ldots \times$$
$$p(\text{singer} \mid \text{Thom York is the, of Radiohead}) \times \ldots \times$$
$$p(\text{Radiohead} \mid \text{Thom York is the singer of})$$

Related Work

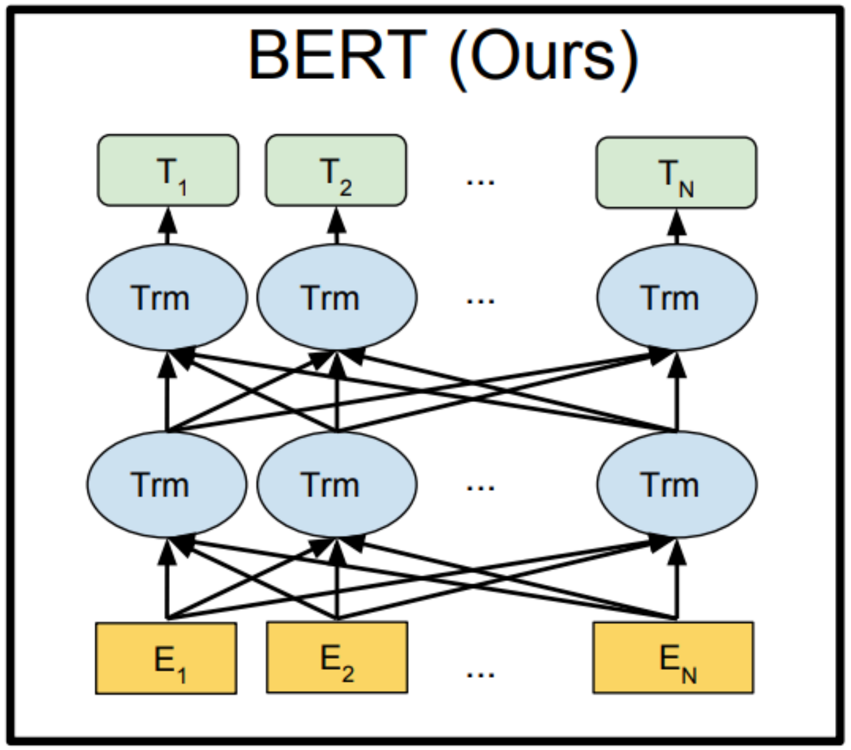
- AutoEncoding(AE): Utilize bidirectional contexts for reconstruction.
  - Given a sequence $\mathbf{x} = (x_1, x_2, x_{mask}, x_4, x_{mask})$

$$\prod p(x_{mask} \mid \hat{x}) = p(x_3 \mid x_{1,2,4}) \times p(x_5 \mid x_{1,2,4})$$

  - Maximum the likelihood

$$\max_{\theta} \quad \log p_\theta(\mathbf{x}_{mask} \mid \hat{\mathbf{x}}) = \sum_{t=1}^{T} \log p_\theta(x_t \mid \hat{\mathbf{x}}) = \sum_{t=1}^{T} \log \frac{\exp(H_\theta(\hat{\mathbf{x}})_t^T e(x_t))}{\sum_{x'} \exp(H_\theta(\hat{\mathbf{x}})_t^T e(x'))}$$

BERT



Source: https://arxiv.org/pdf/1810.04805.pdf

Related Work

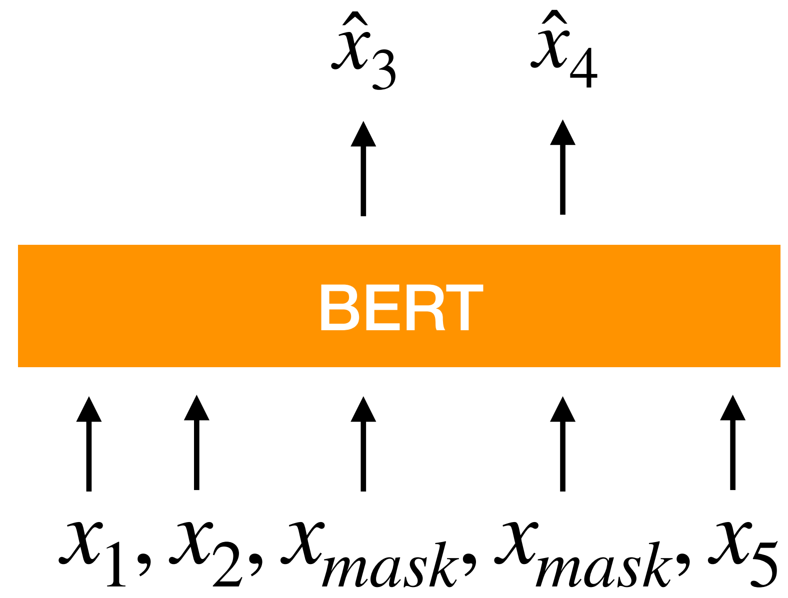
Given a sequence $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5)$

- AutoEncoding(AE):

  Objective: Given $x_1, x_2, x_5 \rightarrow$ Predict x_3, x_4

  $$p(x_3 \mid x_1, x_2, x_5) \times p(x_4 \mid x_1, x_2, x_5)$$

BERT MLM



Issue:
- ***Independence Assumption***
- ***Input Noise***

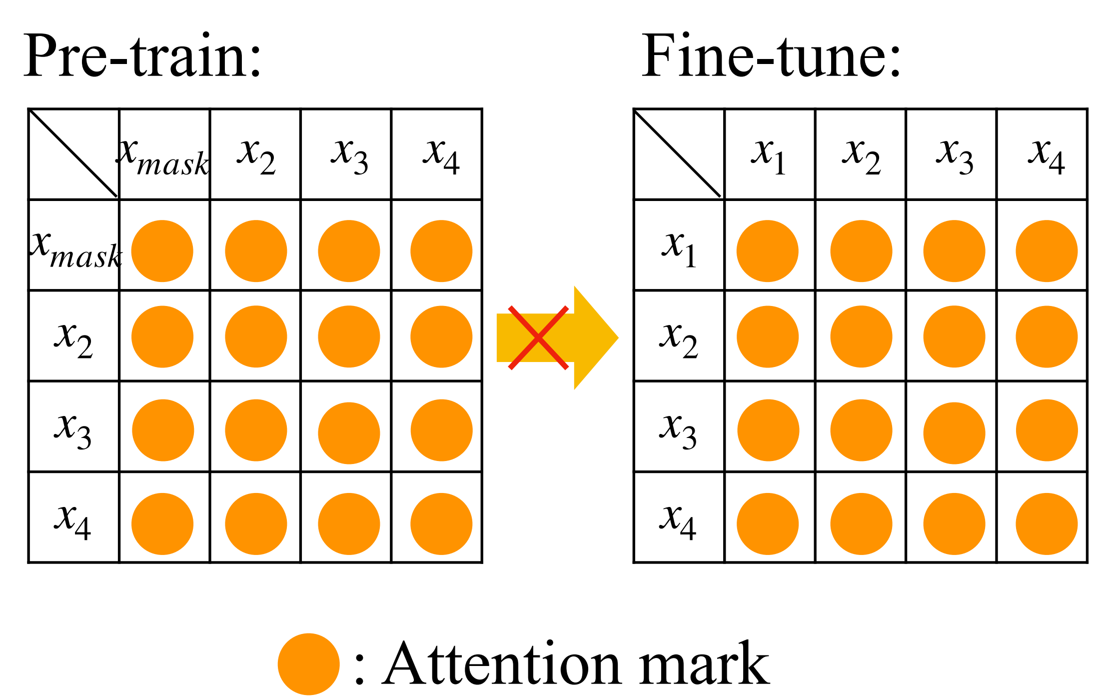
Passage: Thom Yorke is the singer of Radiohead.

Mask token $\rightarrow$ Thom Yorke is the x_{mask} of x_{mask}.

BERT: $p(\text{singer} \mid \text{Thom Yorke is the, of}) \times$
$p(\text{Radiohead} \mid \text{Thom Yorke is the, of})$

XLNet: $p(\text{singer} \mid \text{Thom Yorke is the, of}) \times$
$p(\text{Radiohead} \mid \text{Thom Yorke is the singer of})$

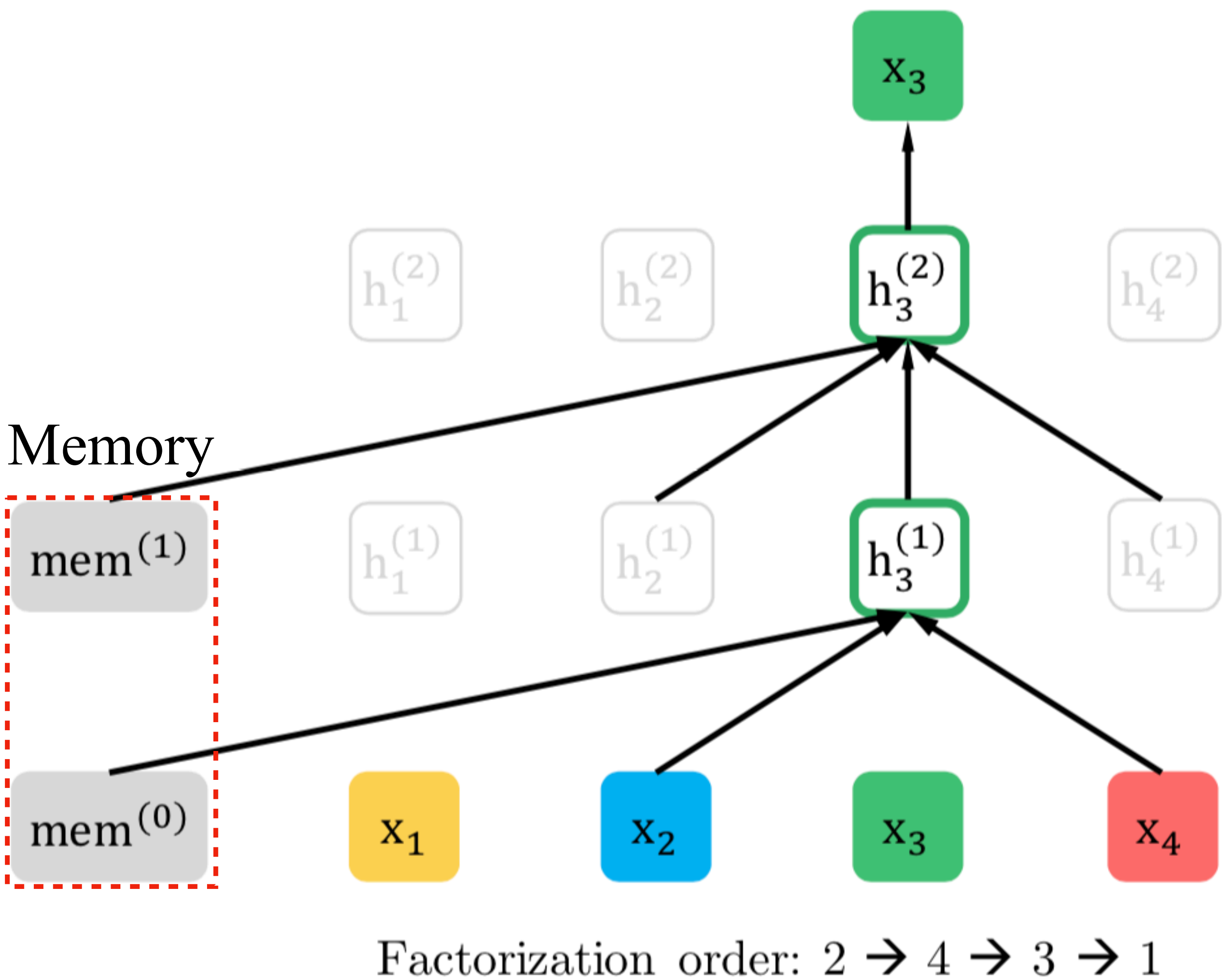
BERT



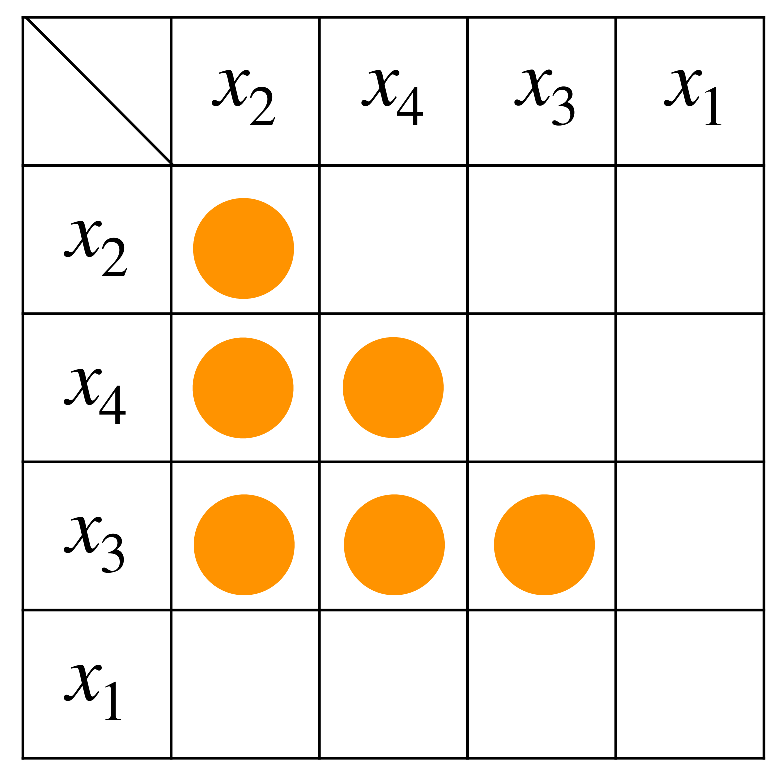
Permutation Language Modeling

PLM learns to utilize **contextual information** from **all positions** (capturing bidirectional context) without using ***<mask>*** token.

1. Given a sequence: x_1, x_2, x_3, x_4, target: x_3
2. Permutation: x_2, x_4, x_3, x_1
3. $p(x_3 \,|\, mem, x_2, x_4, x_3)$

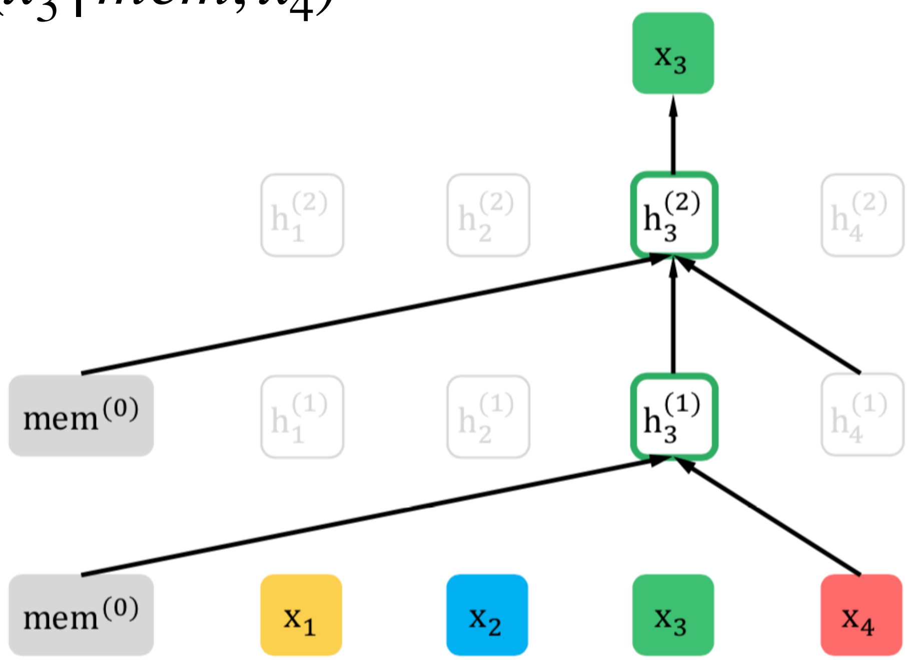


Self-Attention



Permutation Language Modeling

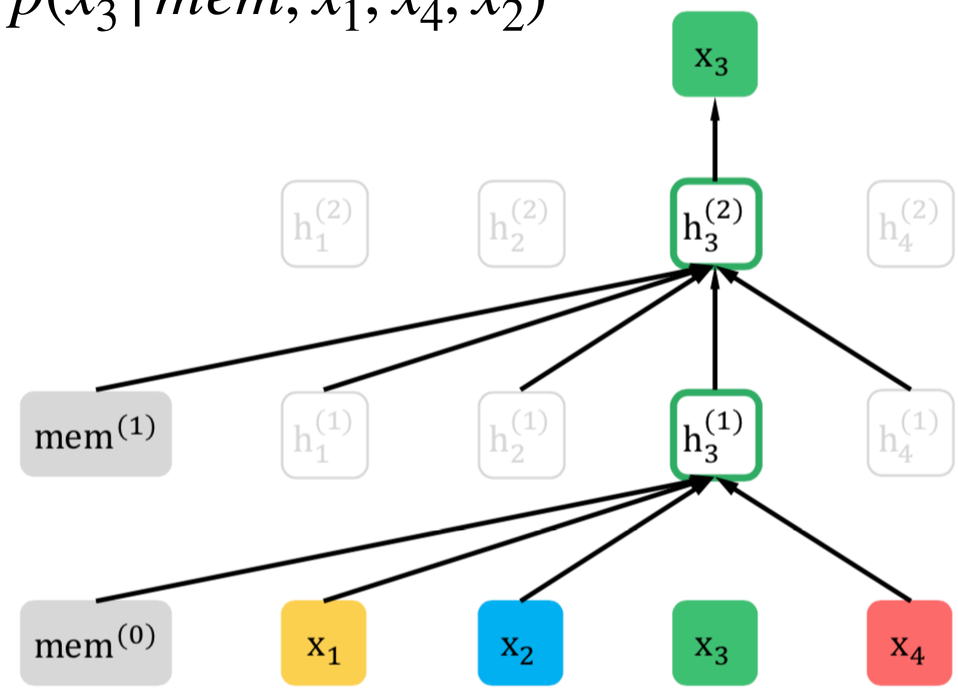
1. Given a sequence: x_1, x_2, x_3, x_4, target: x_3
2. Permutation: x_4, x_3, x_1, x_2
3. $p(x_3 \mid mem, x_4)$



Factorization order: 4 → 3 → 1 → 2

	x_4	x_3	x_1	x_2
x_4	●			
x_3				
x_1				
x_2				

1. Given a sequence: x_1, x_2, x_3, x_4, target: x_3
2. Permutation: x_1, x_4, x_2, x_3
3. $p(x_3 \mid mem, x_1, x_4, x_2)$



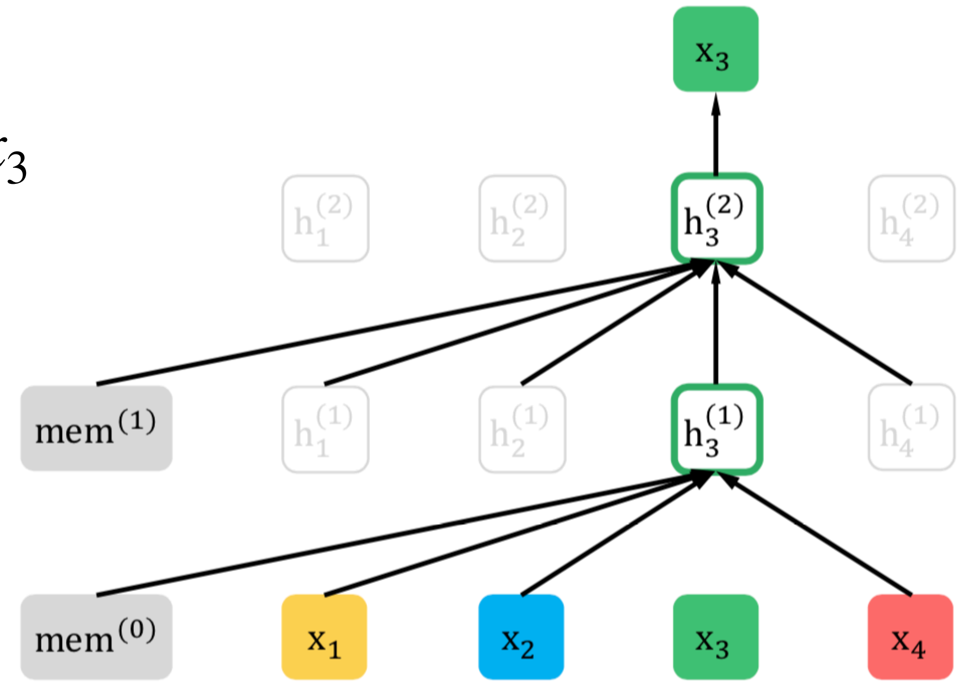
Factorization order: 1 → 4 → 2 → 3

	x_1	x_4	x_2	x_3
x_1	●			
x_4	●	●		
x_2	●	●	●	
x_3				

Permutation Language Modeling

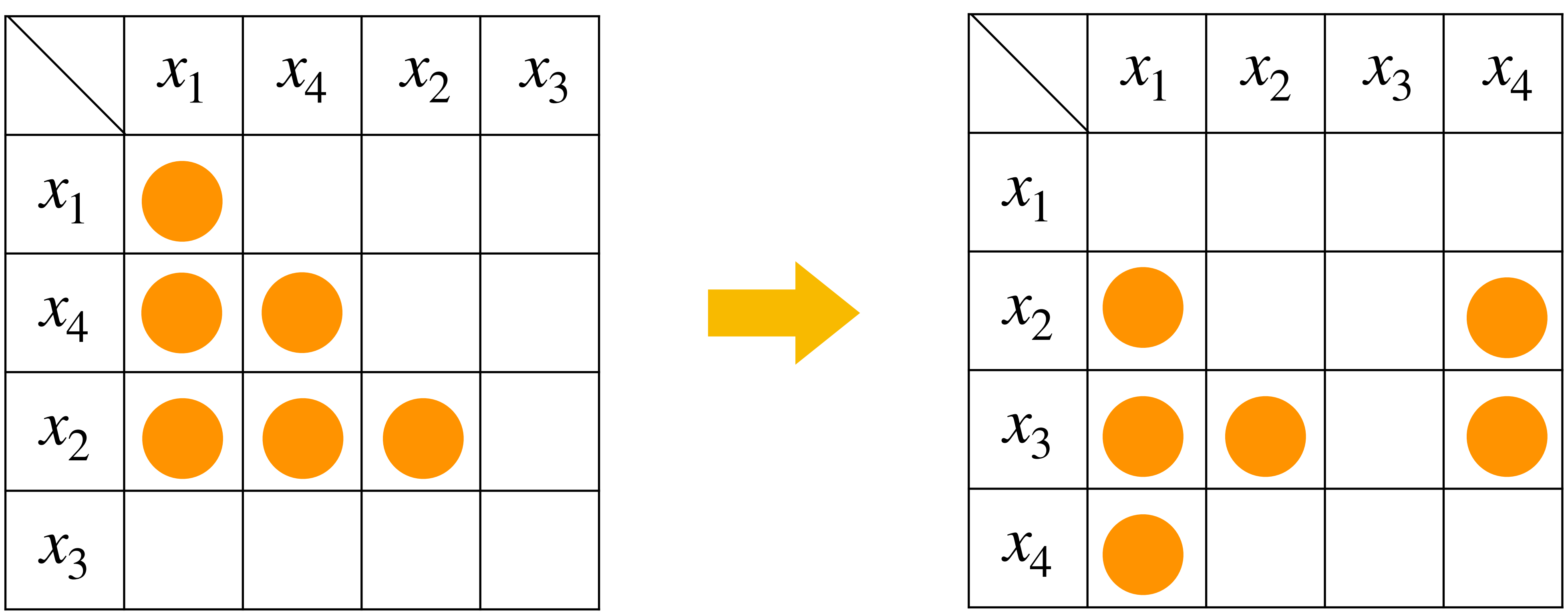
Remark on Permutation

1. Given a sequence: x_1, x_2, x_3, x_4, target: x_3
2. Permutation: x_1, x_4, x_2, x_3
3. $p(x_3 \,|\, mem, x_1, x_4, x_2)$



Factorization order: 1 → 4 → 2 → 3

XLNet



Two-Stream Self-Attention

- Use **Content stream** to learn **bidirectional contexts**.
- Use **Query stream** instead of **<Mask> token** and **predict target**.

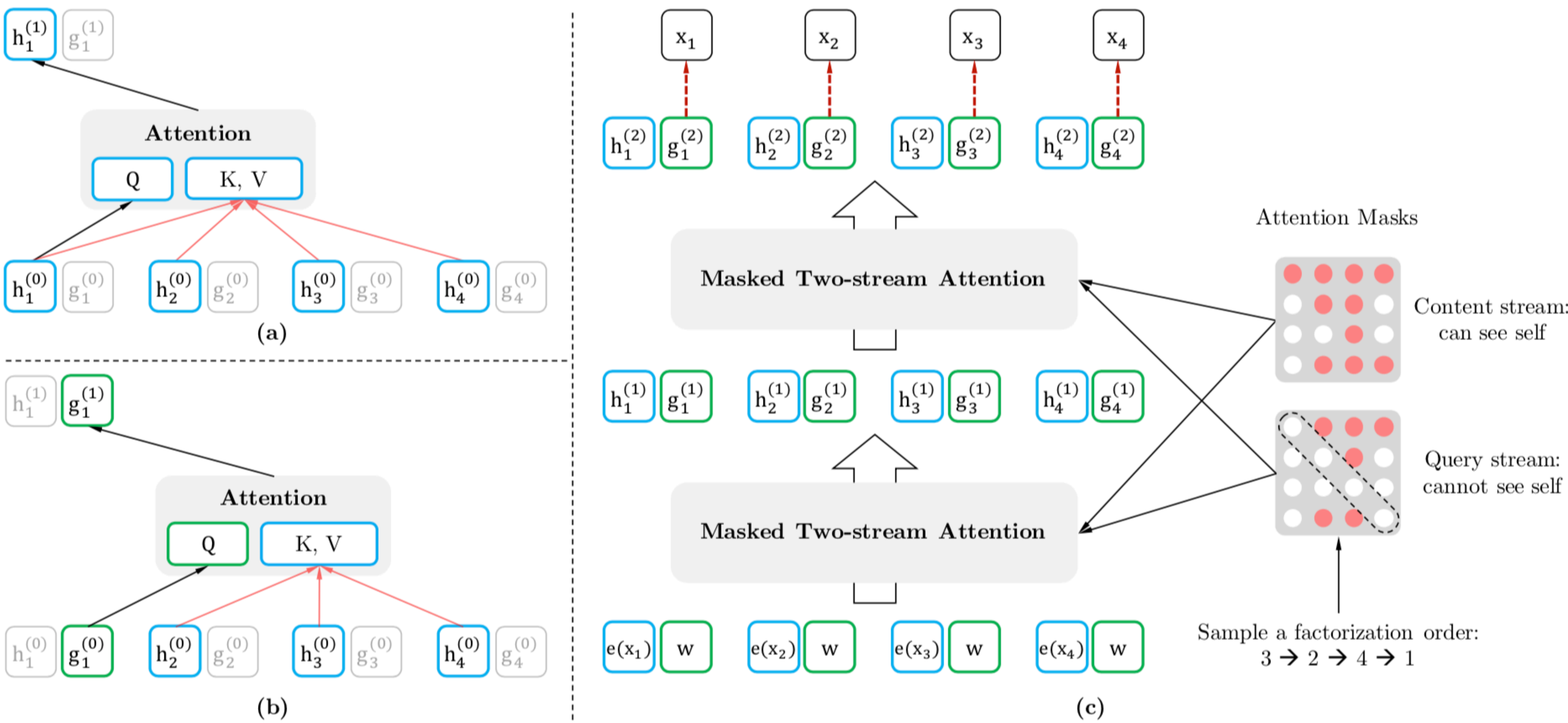
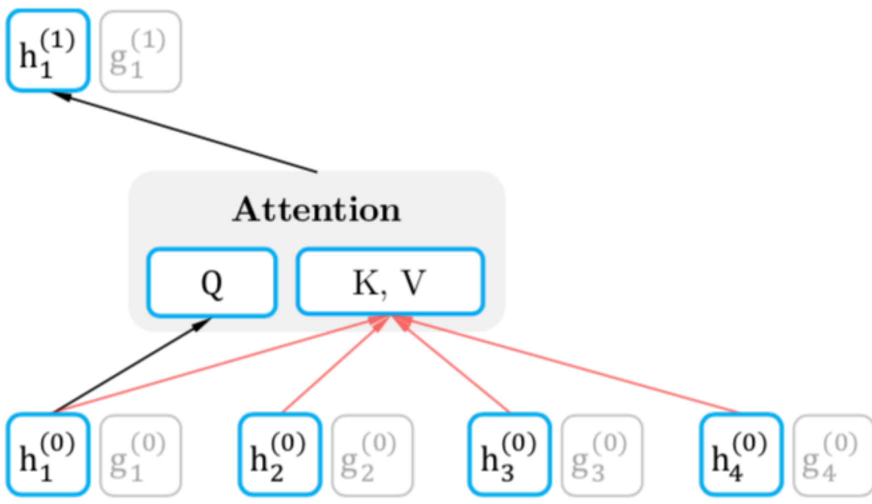


Figure 2: (a): Content stream attention, which is the same as the standard self-attention. (b): Query stream attention, which does not have access information about the content x_{z_t}. (c): Overview of the permutation language modeling training with two-stream attention.

Two-Stream Self-Attention

Content stream



(a)

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

$$h_{z_t}^{m} = \text{Attention}(Q = h_{z_t}^{m-1}, KV = \underline{h_{z_{\leq t}}^{m-1}}; \theta)$$

Sample a factorization order : 3 → 2 → 4 → 1

target : 1

	h_1^0	h_2^0	h_3^0	h_4^0
h_1^0	●	●	●	●
h_2^0		●	●	
h_3^0			●	
h_4^0		●	●	●

QK^T

h_1^0
h_2^0
h_3^0
h_4^0

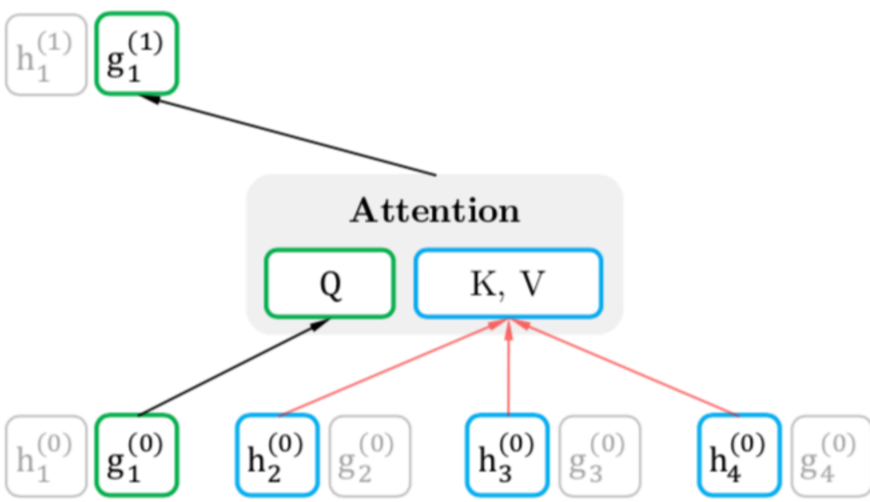
V

=

$h_1^1 = h_{1\sim4}^0$
$h_2^1 = h_{2\sim3}^0$
$h_3^1 = h_3^0$
$h_4^1 = h_{2\sim4}^0$

Two-Stream Self-Attention

Query stream



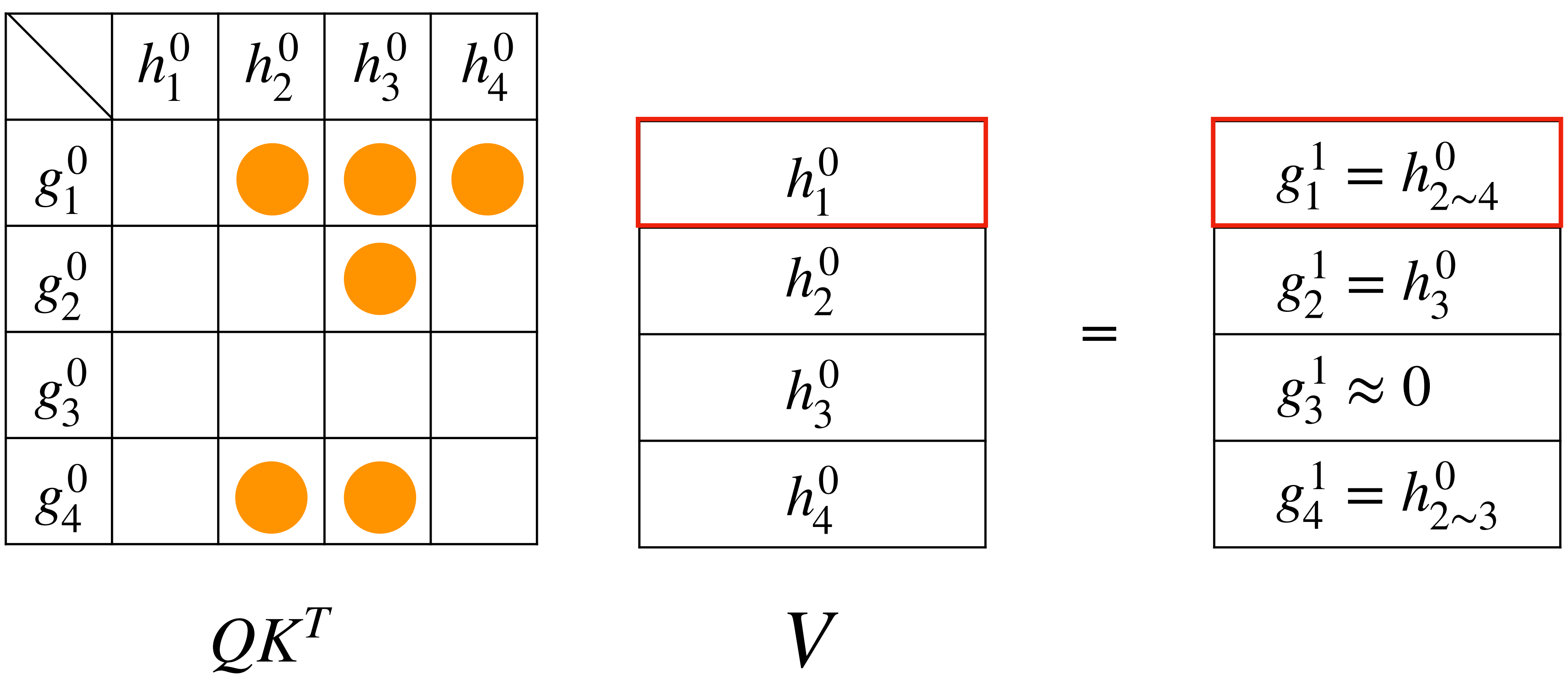
(b)

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

$$g_{z_t}^{m} = \text{Attention}(Q = g_{z_t}^{m-1}, KV = \underline{h_{z_{<t}}^{m-1}}; \theta)$$

Sample a factorization order : $3 \rightarrow 2 \rightarrow 4 \rightarrow 1$

target : 1



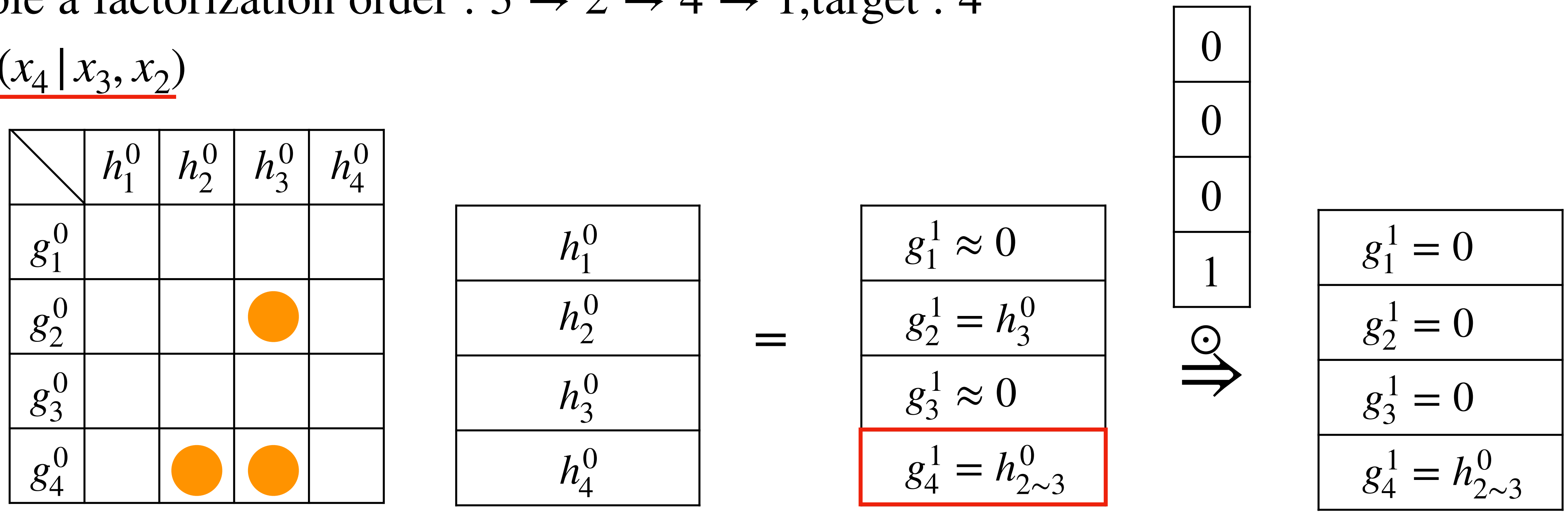
Two-Stream Self-Attention

Query stream

$$p_\theta(X_{z_t} = x \mid \mathbf{x}_{z<t}) = \frac{\exp(e(x)^T g_\theta(\mathbf{x}_{z<t}, z_t))}{\sum_{x'} \exp(e(x')^T g_\theta(\mathbf{x}_{z<t}, z_t))}$$

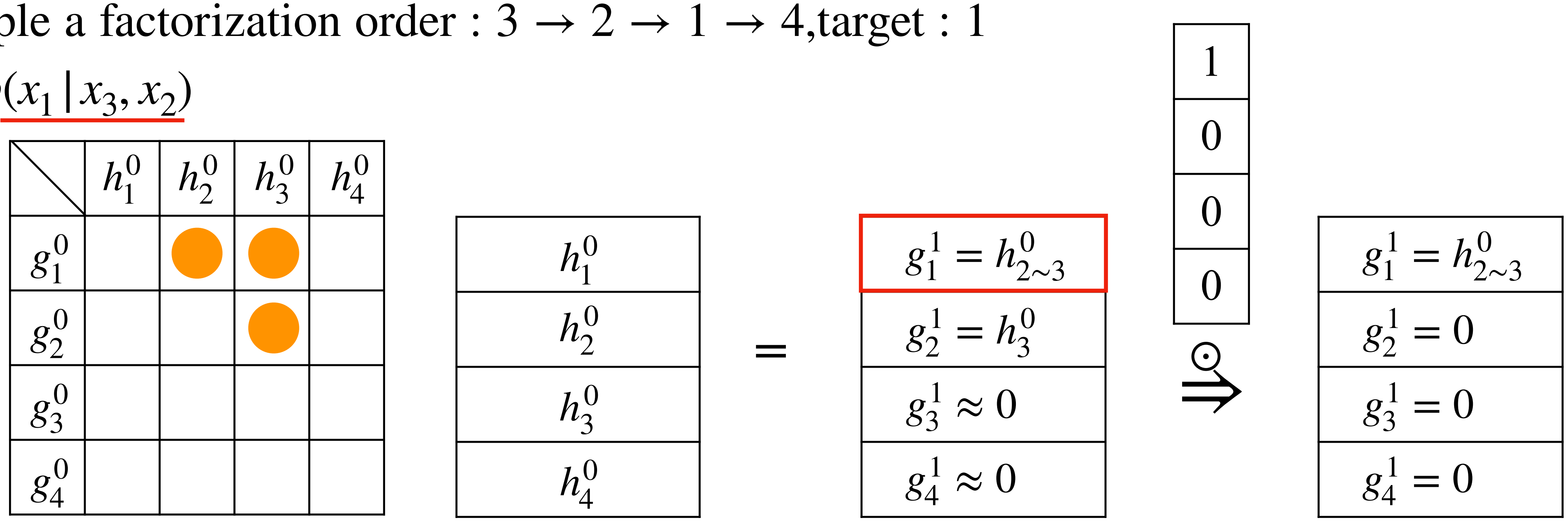
Sample a factorization order : $3 \to 2 \to 4 \to 1$,target : 4

$\Rightarrow p(x_4 \mid x_3, x_2)$



Sample a factorization order : $3 \to 2 \to 1 \to 4$,target : 1

$\Rightarrow p(x_1 \mid x_3, x_2)$



Long Text Understanding

- BERT: Input is restricted to **fixed-length** context. $\{x_1, x_2, \ldots, x_{512}\}$
- Transformer-XL: Use **segment recurrence mechanism** and **relative positional encoding** to pre-train long context (>512).
- XLNet:

$$h^m_{z_t} = \text{Attention}(Q = h^{m-1}_{z_t}, KV = [\tilde{h}^{m-1}, h^{m-1}_{z_{\le t}}]; \theta),\ [\,.\,,\,.\,] : \text{Concatenate}$$

$\tilde{h}^m$: Previous segment hidden representation.

$$X = s_{1:2T}$$

$$\text{Segment} \Rightarrow \tilde{x} = s_{1:T},\ x = s_{T+1:2T}$$

$$\text{Permutation} \Rightarrow \tilde{z} = [1,\ldots,T],\ z = [T+1,\ldots,2T]$$

$$\tilde{h}^m_{z_t} = \text{Attention}(Q = \tilde{h}^{m-1}_{z_t}, KV = \tilde{h}^{m-1}_{z_{\le t}})$$

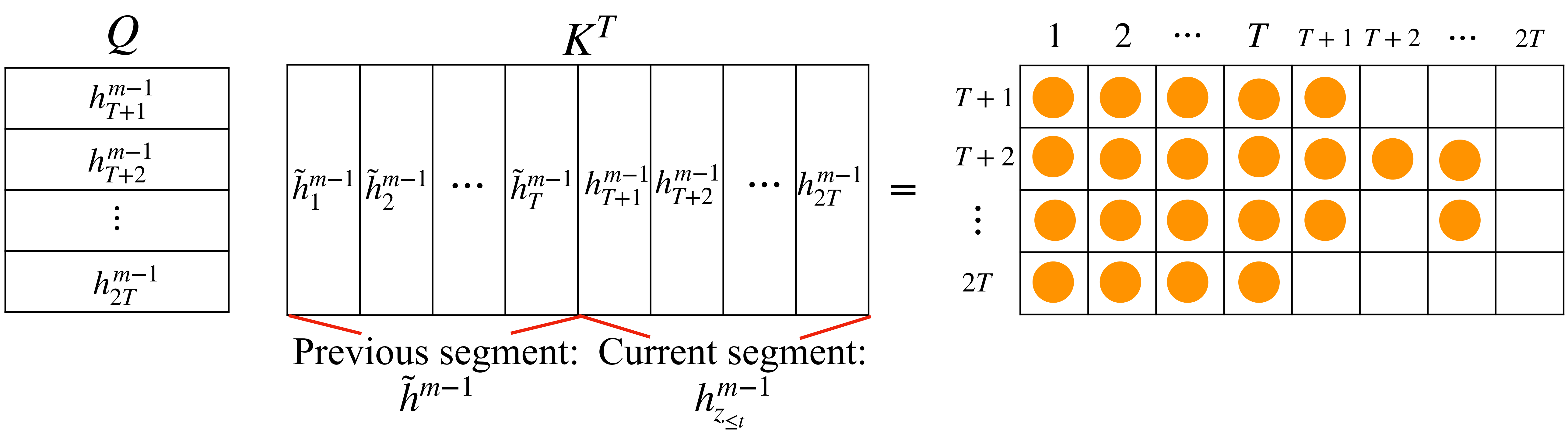
$s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8$

$\downarrow$

z_5, z_7, z_6, z_8

$\downarrow$

Target: z_6



Evaluation

Partial Prediction: Predict last $\frac{1}{K}$ tokens (K=6).

- Sequence length: 512
- Memory length: 384
- Train XLNet-Large on 512 TPUs for 500k steps about 2.5 days.

Pre-train Dataset	**Size(GB)**
English wiki	2.78
BookCorpus	1.09
Giga5	4.75
ClueWeb	4.3
Common Crawl	19.97

Fine-tune Dataset	**Task**
Race	QA
SQuAD	QA
IMDB	Text classification
Yelp	Text classification
DBpedia	Text classification
AG	Text classification
Amazon	Text classification
GLEU	…

Evaluation

RACE	Accuracy	Middle	High
GPT [25]	59.0	62.9	57.4
BERT [22]	72.0	76.6	70.1
BERT+OCN* [28]	73.5	78.4	71.5
BERT+DCMN* [39]	74.1	79.5	71.8
XLNet	**81.75**	**85.45**	**80.21**

Table 1: Comparison with state-of-the-art results on the test set of RACE, a reading comprehension task. $*$ indicates using ensembles. "Middle" and "High" in RACE are two subsets representing middle and high school difficulty levels. All BERT and XLNet results are obtained with a 24-layer architecture with similar model sizes (aka BERT-Large). Our single model outperforms the best ensemble by 7.6 points in accuracy.

SQuAD1.1	EM	F1	SQuAD2.0	EM	F1
Dev set results without data augmentation					
BERT [10]	84.1	90.9	BERT† [10]	78.98	81.77
XLNet	**88.95**	**94.52**	XLNet	**86.12**	**88.79**
Test set results on leaderboard, with data augmentation (as of June 19, 2019)					
Human [27]	82.30	91.22	BERT+N-Gram+Self-Training [10]	85.15	87.72
ATB	86.94	92.64	SG-Net	85.23	87.93
BERT* [10]	87.43	93.16	BERT+DAE+AoA	85.88	88.62
XLNet	**89.90**	**95.08**	XLNet	**86.35**	**89.13**

Table 2: A single model XLNet outperforms human and the best ensemble by 7.6 EM and 2.5 EM on SQuAD1.1. $*$ means ensembles, † marks our runs with the official code.

Evaluation

†: Jointly train XLNet on the four largest datasets-MNLI, SST-2, QNLI, QQP, and fine-tune on the other datasets.

Model	MNLI	QNLI	QQP	RTE	SST-2	MRPC	CoLA	STS-B	WNLI
Single-task single models on dev									
BERT [2]	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-
XLNet	**89.8/-**	**93.9**	**91.8**	**83.8**	**95.6**	**89.2**	**63.6**	**91.8**	-
Single-task single models on test									
BERT [10]	86.7/85.9	91.1	89.3	70.1	94.9	89.3	60.5	87.6	65.1
Multi-task ensembles on test (from leaderboard as of June 19, 2019)									
Snorkel* [29]	87.6/87.2	93.9	89.9	80.9	96.2	91.5	63.8	90.1	65.1
ALICE*	88.2/87.9	95.7	**90.7**	83.5	95.2	92.6	**68.6**	91.1	80.8
MT-DNN* [18]	87.9/87.4	96.0	89.9	**86.3**	96.5	92.7	68.4	91.1	89.0
XLNet*	**90.2/89.7**†	**98.6**†	90.3†	**86.3**	**96.8**†	**93.0**	67.8	**91.6**	**90.4**

Table 4: Results on GLUE. ∗ indicates using ensembles, and † denotes single-task results in a multi-task row. All results are based on a 24-layer architecture with similar model sizes (aka BERT-Large). See the upper-most rows for direct comparison with BERT and the lower-most rows for comparison with state-of-the-art results on the public leaderboard.