XLNet: Generalized Autoregressive Pretraining for Language Understanding

Zexi Huang and Mert Kosan

(Slides partially based on Yang et al's NeurIPS'19 presentation)

Word Embedding: Related Work

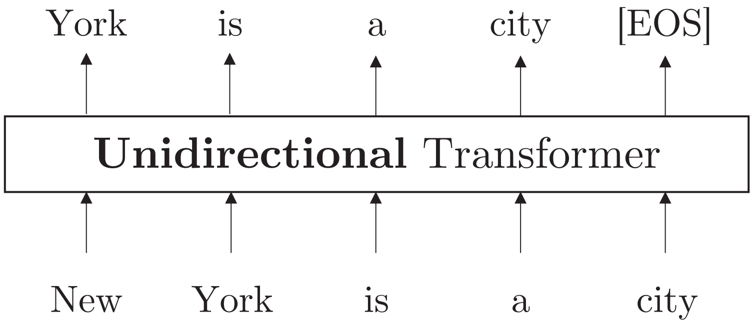
- Context-free embedding: word2vec (Mikolov et al 2013), GloVe (Pennington et al 2014), fastText (Bojanowski et al 2017)

- Autoregressive (AR) models: Semi-supervised sequence learning (Dai and Le 2015), ELMo (Peters et al 2017), GPT (Radford et al 2018)

- Autoencoding (AE) models: BERT (Devlin et al 2018), RoBERTa (Liu et al 2019), ALBERT (Lan et al 2019)

Two Notable Objectives for Language Pretraining

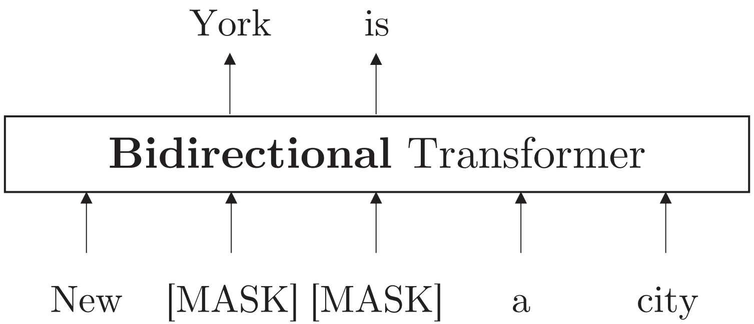
Auto-regressive Language Modeling



$$\log p(\mathbf{x}) = \sum_{t=1}^{T} \log p(x_t | \mathbf{x}_{<t})$$

- **Next-token prediction**

Denoising Auto-encoding (BERT)

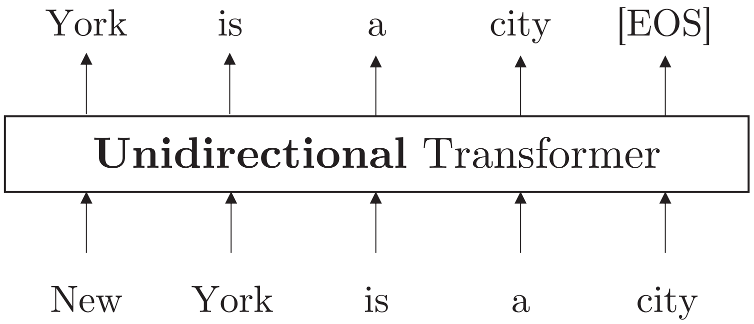


$$\log p(\bar{\mathbf{x}} | \hat{\mathbf{x}}) = \sum_{t=1}^{T} \text{mask}_t \log p(x_t | \hat{\mathbf{x}})$$

- **Reconstruct masked tokens**

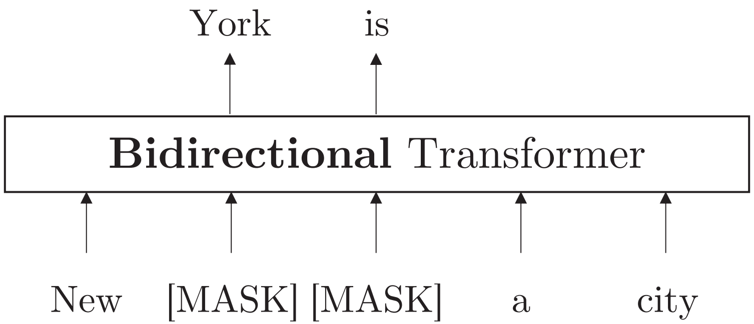
Two Notable Objectives for Language Pretraining

Auto-regressive Language Modeling



No **Bidirectional Context**

Denoising Auto-encoding (BERT)

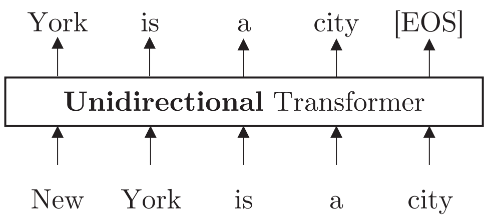


Independent Predictions

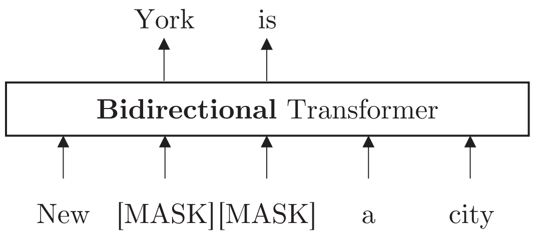
Artificial **Noise**: [MASK]

Two Notable Objectives for Language Pretraining

Auto-regressive Language Modeling



Denoising Auto-encoding (BERT)



Full Auto-regressive **Dependence** ⟺ **Independent** Predictions

Free from artificial **Noise** ⟺ Artificial **Noise**: [MASK]

No **Bidirectional Context** ⟺ Natural **Bidirectional Context**

Desire: Combine the Pros and Remove the Cons

☺ Full Auto-regressive **Dependence**

☺ Free from **Noise**

☺ Natural **Bidirectional Context**

Desire: Combine the Pros and Remove the Cons

XLNet

- An **auto-regressive** model that captures **bidirectional context**

Context Depends on the **Factorization Order**

- **Standard LM**: Left-to-right factorization 1 → 2 → 3 → 4

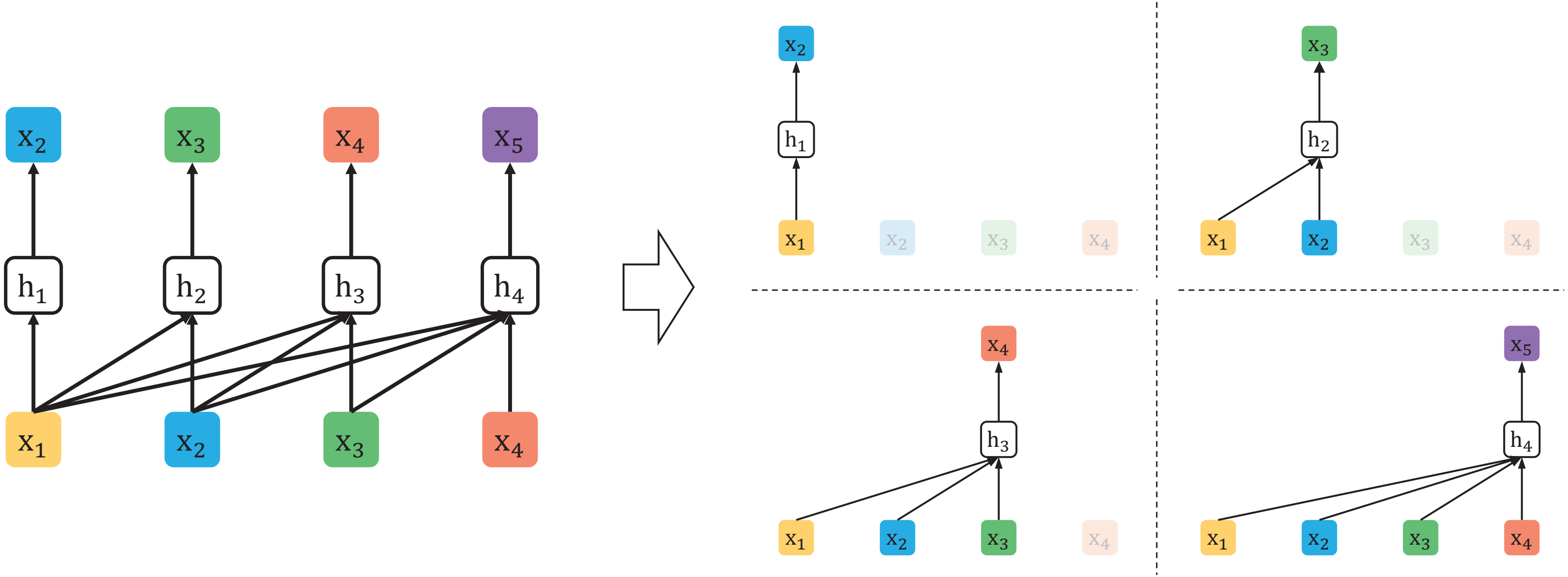
$$P(\mathbf{x}) = P(x_1)P(x_2 \mid \mathbf{x}_1)P(x_3 \mid \mathbf{x}_{1,2})P(x_4 \mid \mathbf{x}_{1,2,3}) \cdots$$

x_1 x_2 x_3 x_4

Context Depends on the **Factorization Order**

- **Standard LM**: Left-to-right factorization 1 → 2 → 3 → 4

$$P(\mathbf{x}) = P(x_1)P(x_2 \mid \mathbf{x}_1)P(x_3 \mid \mathbf{x}_{1,2})P(x_4 \mid \mathbf{x}_{1,2,3})\cdots$$



Context Depends on the **Factorization Order**

- Change the Factorization order to: 4 → 1 → 3 → 2

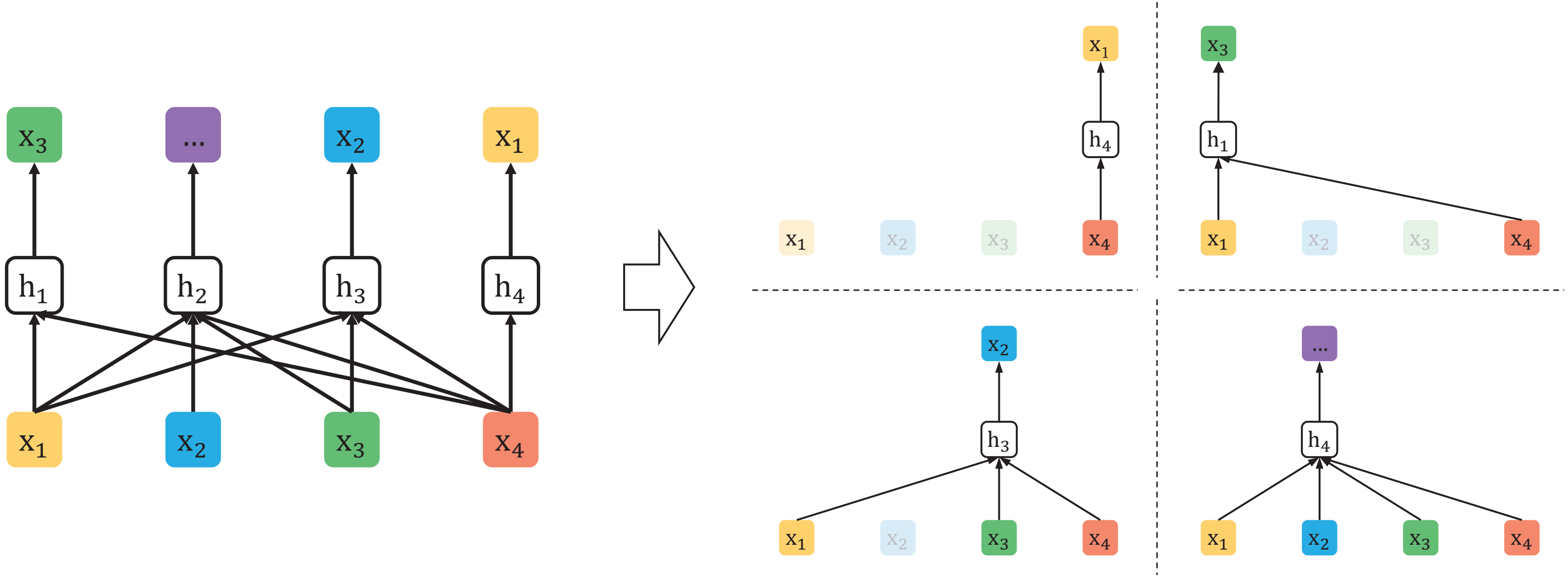
$$P(\mathbf{x}) = P(x_4)P(x_1 \mid \mathbf{x}_4)P(x_3 \mid \mathbf{x}_{1,4})P(x_2 \mid \mathbf{x}_{1,3,4})\cdots$$

x_1 x_2 x_3 x_4

Context Depends on the **Factorization Order**

- Change the Factorization order to: 4 → 1 → 3 → 2

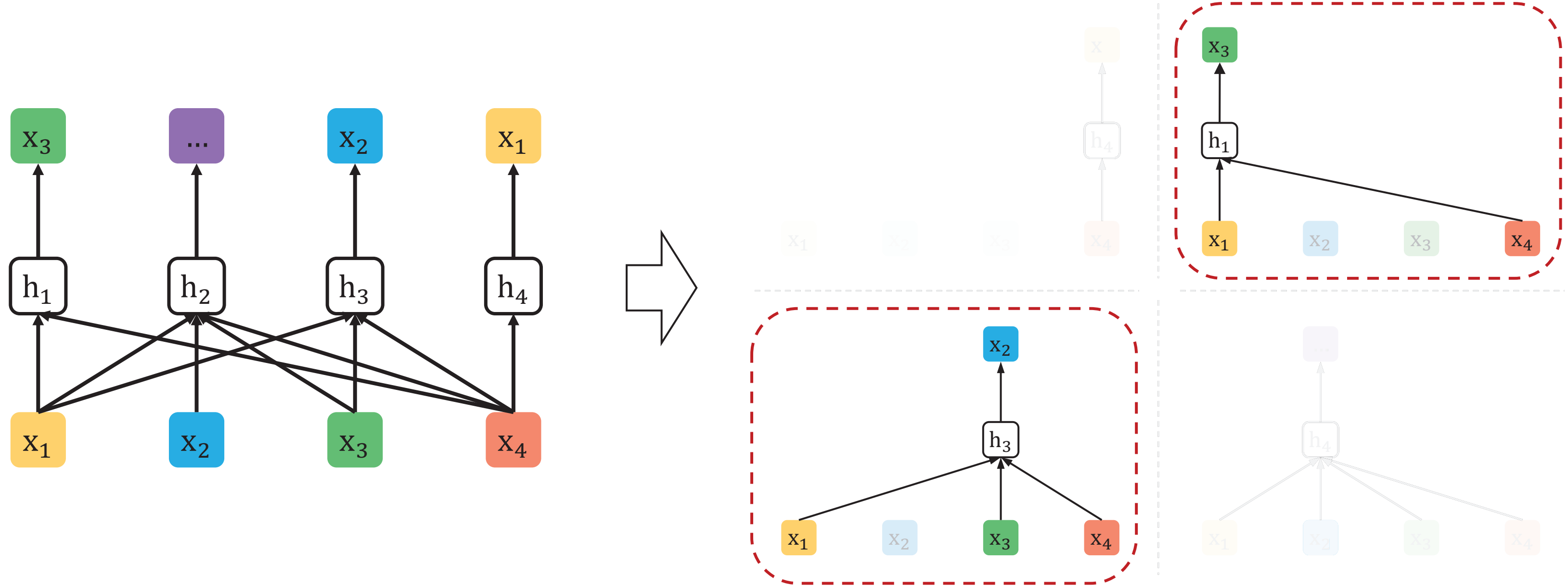
$$P(\mathbf{x}) = P(x_4)P(x_1 \mid \mathbf{x}_4)P(x_3 \mid \mathbf{x}_{1,4})P(x_2 \mid \mathbf{x}_{1,3,4}) \cdots$$



Context Depends on the **Factorization Order**

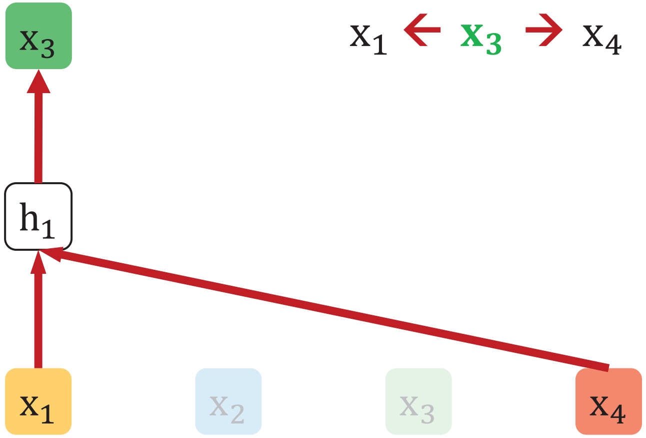
- Change the Factorization order to: 4 → 1 → 3 → 2

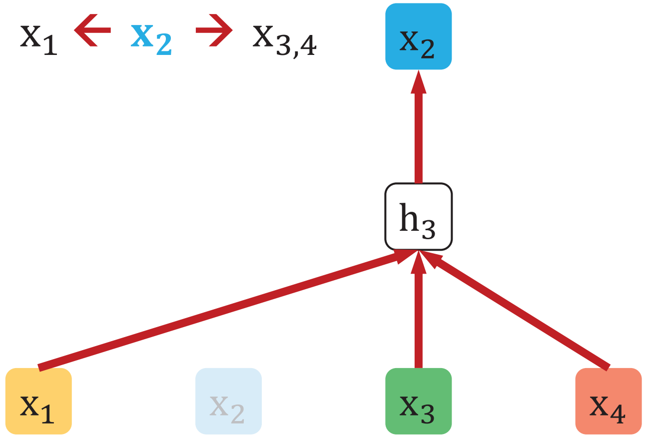
$$P(\mathbf{x}) = P(x_4)P(x_1 \mid \mathbf{x}_4)P(x_3 \mid \mathbf{x}_{1,4})P(x_2 \mid \mathbf{x}_{1,3,4}) \cdots$$



Bidirectional Context via Factorization Order

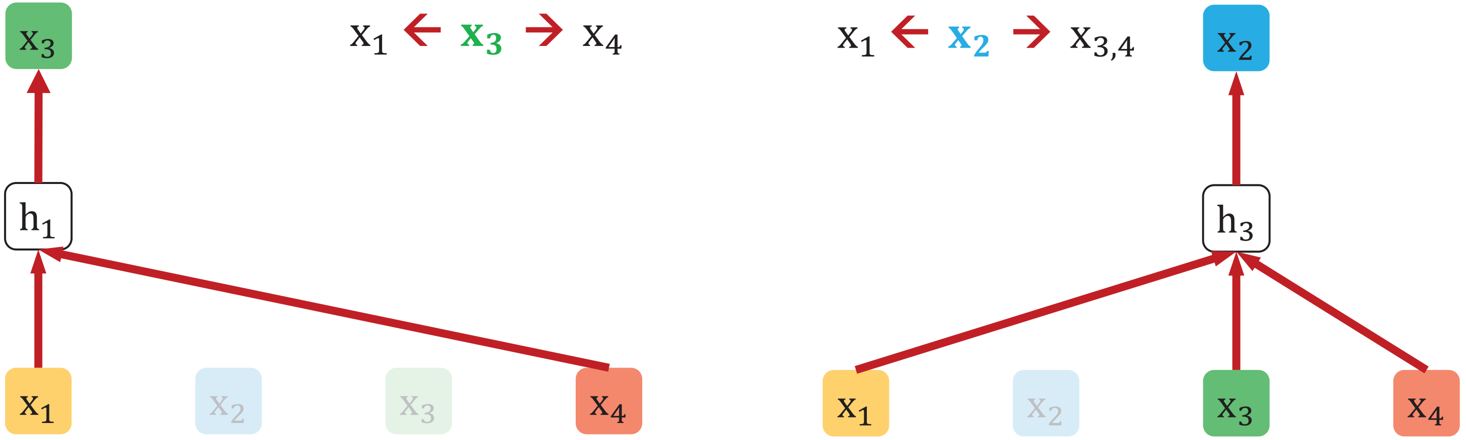
- Factorization order: 4 → 1 → 3 → 2





Bidirectional Context via Factorization Order

- Factorization order: 4 → 1 → 3 → 2



Bidirectional Context

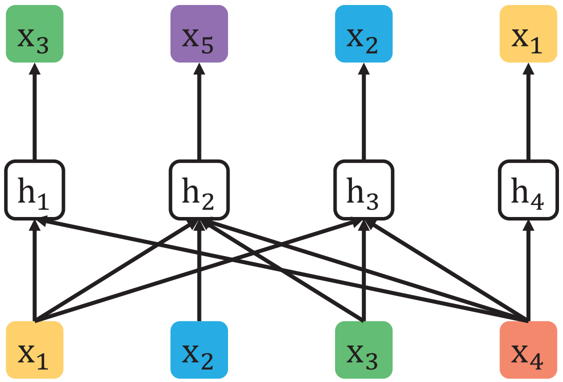
Permutation Language Modeling

- Given a sequence $\mathbf{x}$ of length T
- Uniformly sample a factorization order $\mathbf{z}$ from all possible permutations
- Maximize the permutated log-likelihood

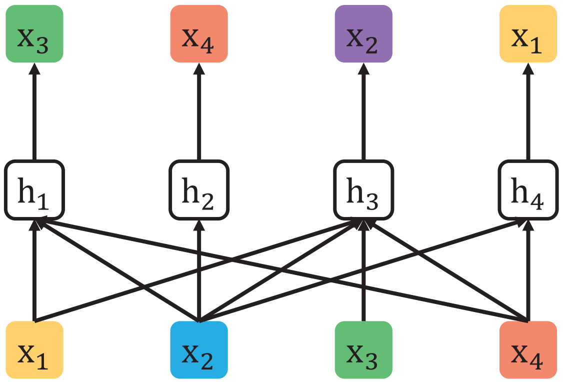
$$\mathbb{E}_{\mathbf{z}\sim\mathcal{Z}_T}\left[\log P(\mathbf{x} \mid \mathbf{z})\right] = \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}_T}\left[\sum_{t=1}^{T} P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)\right]$$

More examples

Factorization order: 4 → 1 → 3 → 2



Factorization order: 2 → 4 → 1 → 3



Target-position-aware Distribution

$$\mathbb{E}_{z_t \sim \mathcal{Z}_T}\left[\sum_{t=1}^{T} \log P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)\right]$$

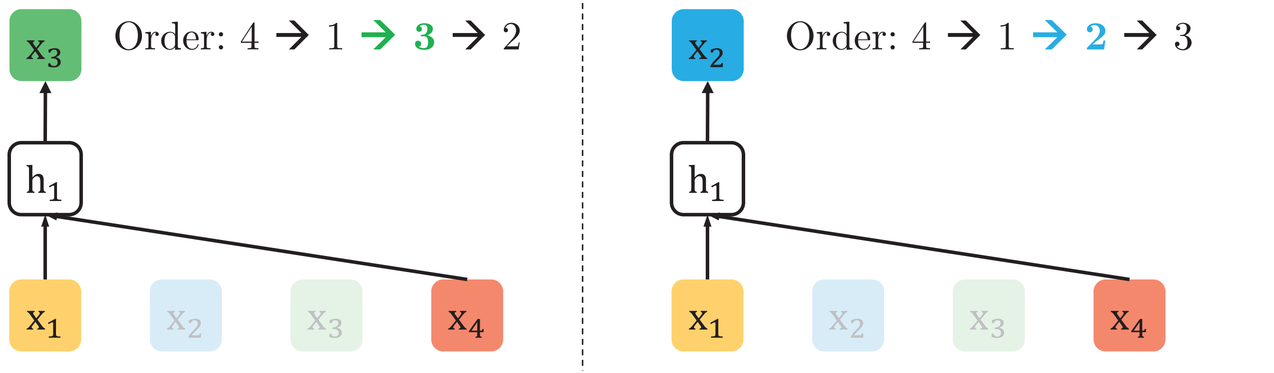
⇒

The distribution $P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)$ must condition on the **target position** z_t

Target-position-aware Distribution

$$\mathbb{E}_{z_t \sim \mathcal{Z}_T}\left[\sum_{t=1}^{T} \log P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)\right]$$

The distribution $P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)$ must condition on the **target position** z_t

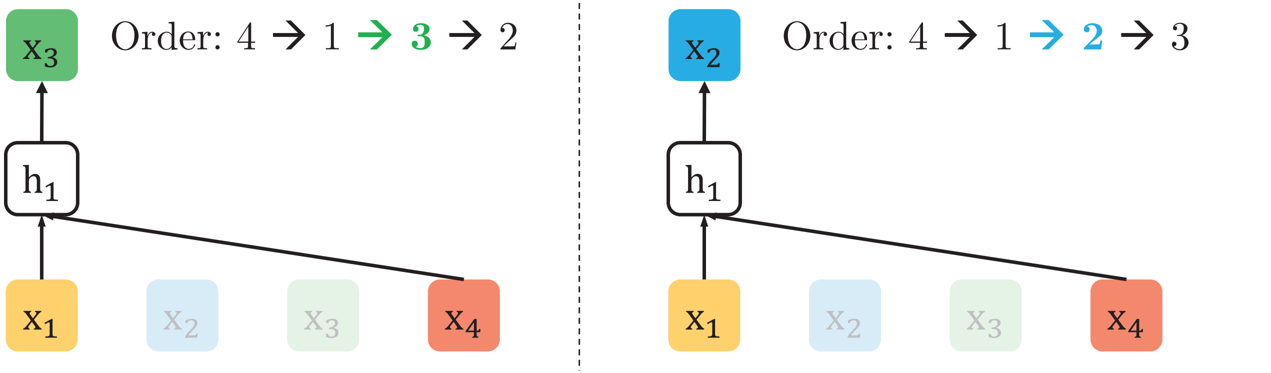


Target-position-aware Distribution

$$\mathbb{E}_{z_t \sim \mathcal{Z}_T}\left[\sum_{t=1}^{T} \log P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)\right]$$

$\Rightarrow$

The distribution $P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t)$ must condition on the **target position** z_t



- Predicting **position 3** and **position 2** requires different prediction distributions
- The prediction distribution should **change according to the target position**

Reparameterization

- Standard Softmax does **NOT** work

$$P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t) = \frac{\exp\left(e(x_{z_t})^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}{\sum_{x'} \exp\left(e(x')^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}$$

No info. of z_t

Reparameterization

- Standard Softmax does **NOT** work

$$P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t) = \frac{\exp\left(e(x_{z_t})^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}{\sum_{x'} \exp\left(e(x')^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}$$

No info. of z_t

- **Proposed** solution: incorporate z_t into **hidden states**

$$P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t) = \frac{\exp\left(e(x_{z_t})^\top g(z_t, \mathbf{x}_{\mathbf{z}_{<t}})\right)}{\sum_{x'} \exp\left(e(x')^\top g(z_t, \mathbf{x}_{\mathbf{z}_{<t}})\right)}$$

Deep Net

Reparameterization

- Standard Softmax does **NOT** work

$$P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t) = \frac{\exp\left(e(x_{z_t})^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}{\sum_{x'} \exp\left(e(x')^\top h(\mathbf{x}_{\mathbf{z}_{<t}})\right)}$$

No info. of z_t

- **Proposed** solution: incorporate z_t into **hidden states**

$$P(x_{z_t} \mid \mathbf{x}_{\mathbf{z}_{<t}}, z_t) = \frac{\exp\left(e(x_{z_t})^\top g(z_t, \mathbf{x}_{\mathbf{z}_{<t}})\right)}{\sum_{x'} \exp\left(e(x')^\top g(z_t, \mathbf{x}_{\mathbf{z}_{<t}})\right)}$$

Deep Net

Question: how to implement $g(z_t, \mathbf{x}_{z_{<t}})$?

Target Position Aware Representation: $g(z_t, \mathbf{x}_{z_{<t}})$

Reuse the Idea of Attention $\Rightarrow$

- Stand at the target position z_t
- Gather information from $\mathbf{x}_{z_{<t}}$

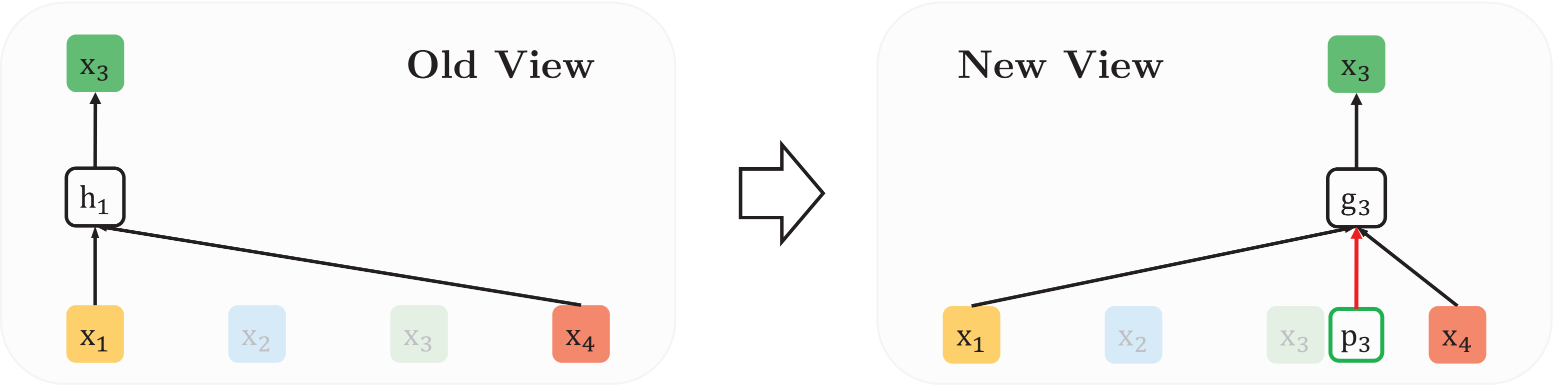
$$g(z_t, \mathbf{x}_{\mathbf{z}_{<t}}) = \mathrm{Attn}_\theta\Big(\underbrace{\mathrm{Q} = \mathrm{Enc}(z_t)}_{\text{Stand at } z_t}, \quad \underbrace{\mathrm{KV} = \mathbf{h}(\mathbf{x}_{\mathbf{z}_{<t}})}_{\text{Gather info. from } \mathbf{x}_{\mathbf{z}_{<t}}}\Big)$$

Target Position Aware Representation: $g(z_t, \mathbf{x}_{z_{<t}})$

Reuse the Idea of Attention ⇨

- Stand at the target position z_t
- Gather information from $\mathbf{x}_{z_{<t}}$

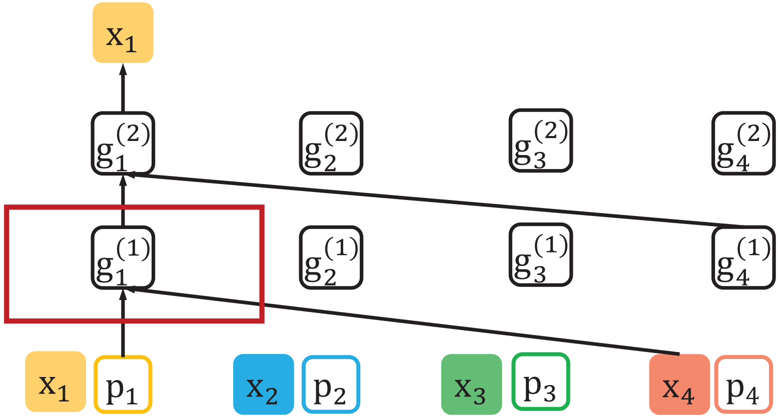
$$g(z_t, \mathbf{x}_{\mathbf{z}_{<t}}) = \mathrm{Attn}_\theta\Big(\underbrace{Q = \mathrm{Enc}(z_t)}_{\text{Stand at } z_t},\ \underbrace{KV = \mathbf{h}(\mathbf{x}_{\mathbf{z}_{<t}})}_{\text{Gather info. from } \mathbf{x}_{\mathbf{z}_{<t}}}\Big)$$



Contradiction: Predicting Self and Others

- Factorization order: 4 → 1 → 3 → 2

Use $g_1^{(1)}$ to predict $\boldsymbol{x_1}$ (self)

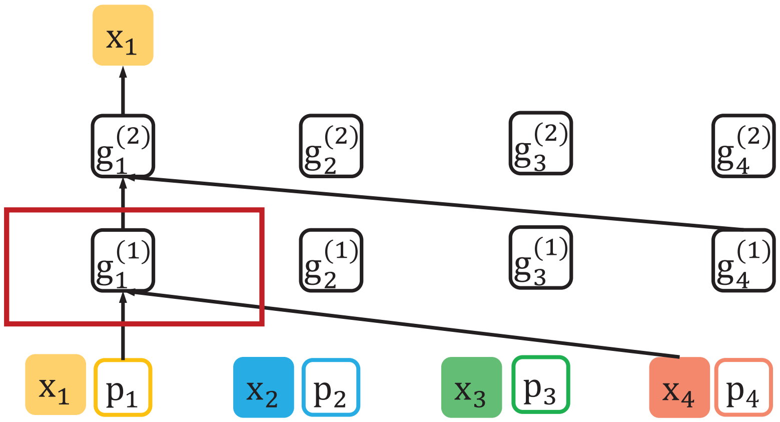


Should not encode $\boldsymbol{x_1}$

Contradiction: Predicting Self and Others

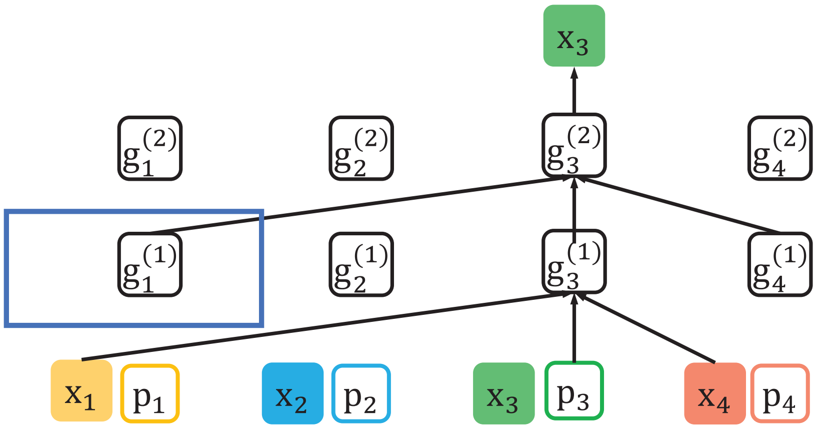
- Factorization order: 4 → 1 → 3 → 2

Use $g_1^{(1)}$ to predict $\boldsymbol{x_1}$ (self)



Should not encode $\boldsymbol{x_1}$

Use $g_1^{(1)}$ to predict $\boldsymbol{x_3}$ (other)

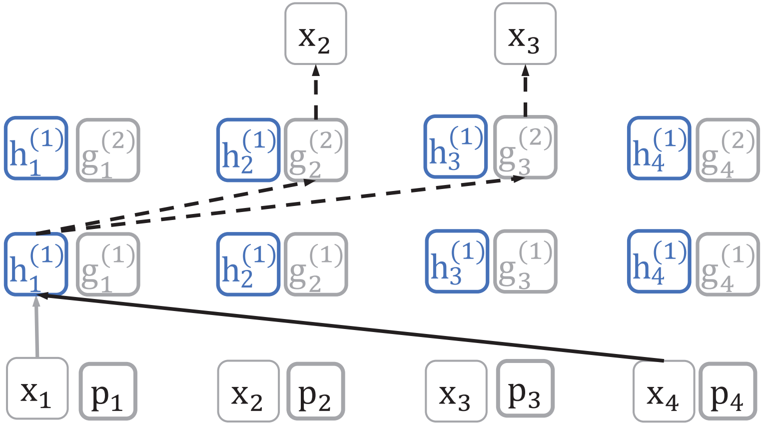


Should encode $\boldsymbol{x_1}$

Two-Stream Attention

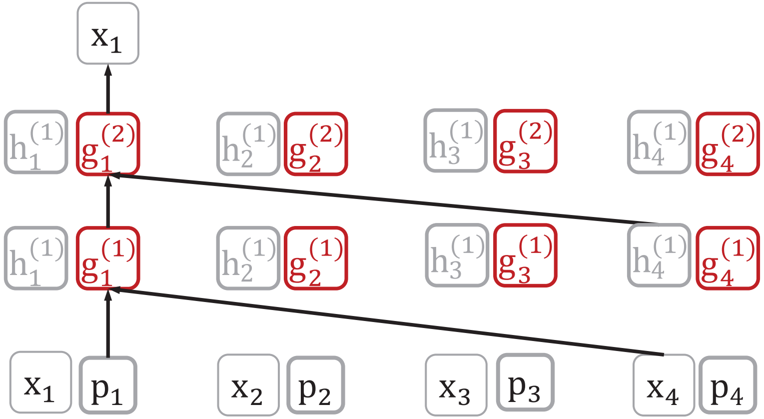
- Factorization order: 4 → 1 → 3 → 2

Encoding. Predicting x_2 and x_3 (others).



$\boldsymbol{h_1}$ encodes $\boldsymbol{x_1}$

Decoding. Predicting $\boldsymbol{x_1}$ (self).



$\boldsymbol{g_1}$ does not encode $\boldsymbol{x_1}$

Summarizing XLNet

Challenges

Solutions

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Solutions

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Solutions

Permutation language modeling

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Standard parameterization is reduced to bag-of-words

Solutions

Permutation language modeling

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Standard parameterization is reduced to bag-of-words

Solutions

Permutation language modeling

Reparameterization with positions

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Standard parameterization is reduced to bag-of-words

Contradiction for predicting both self and others

Solutions

Permutation language modeling

Reparameterization with positions

Summarizing XLNet

Challenges

Independence assumption and distribution discrepancy in BERT

Standard parameterization is reduced to bag-of-words

Contradiction for predicting both self and others

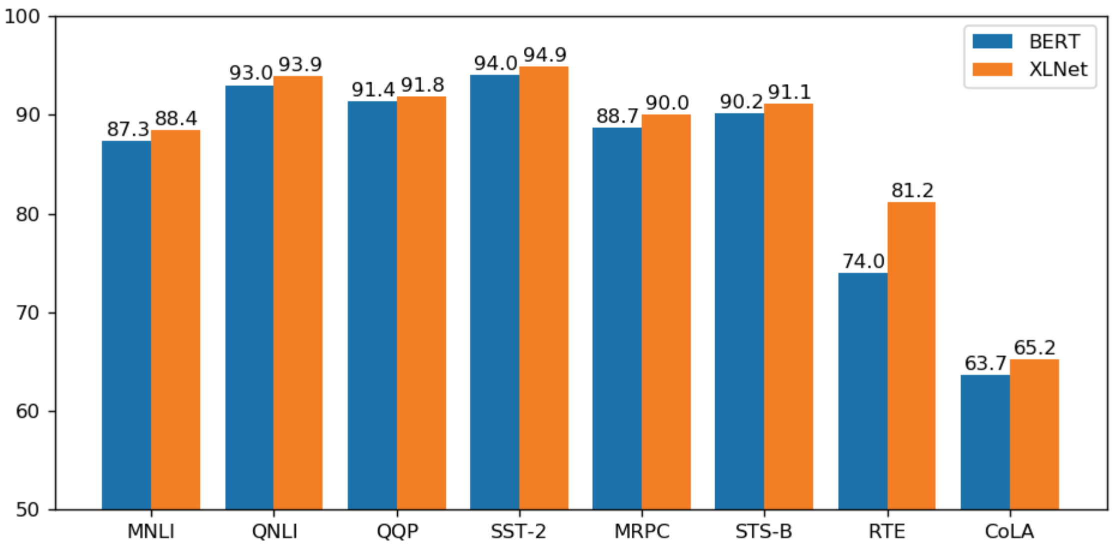
Solutions

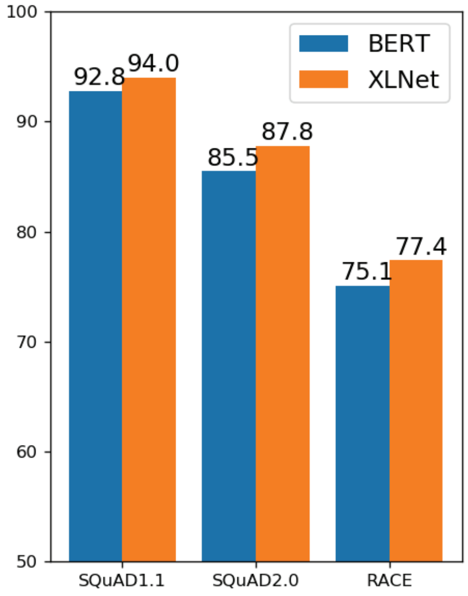
Permutation language modeling

Reparameterization with positions

Two-stream attention

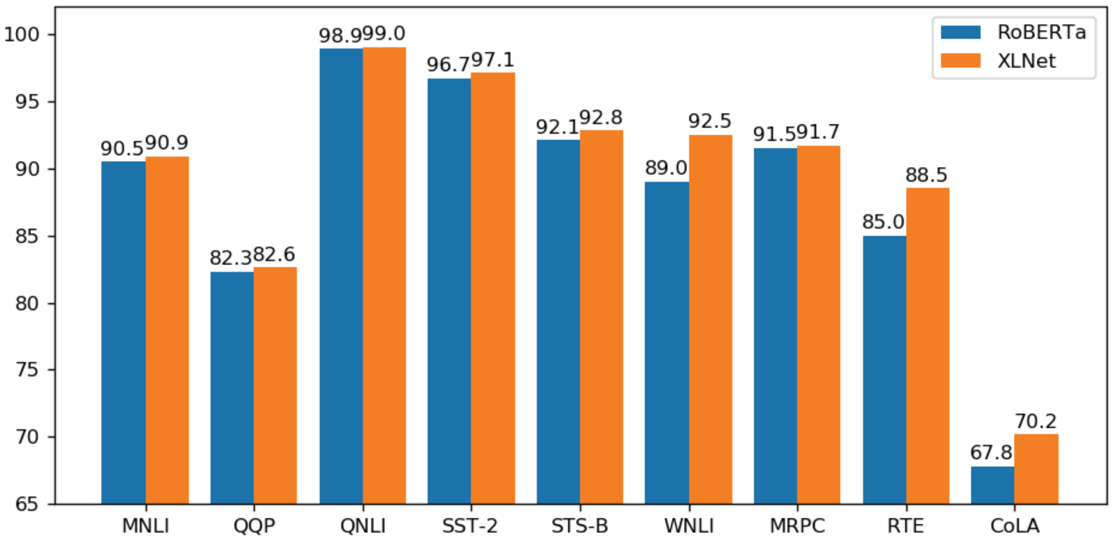
Experiment: Comparison with BERT

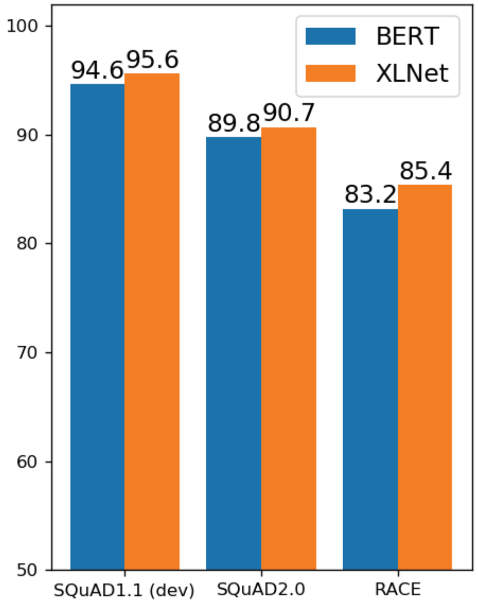




We report the best of 3 BERT variants.
Almost identical training recipes.

Experiment: Comparison with RoBERTa



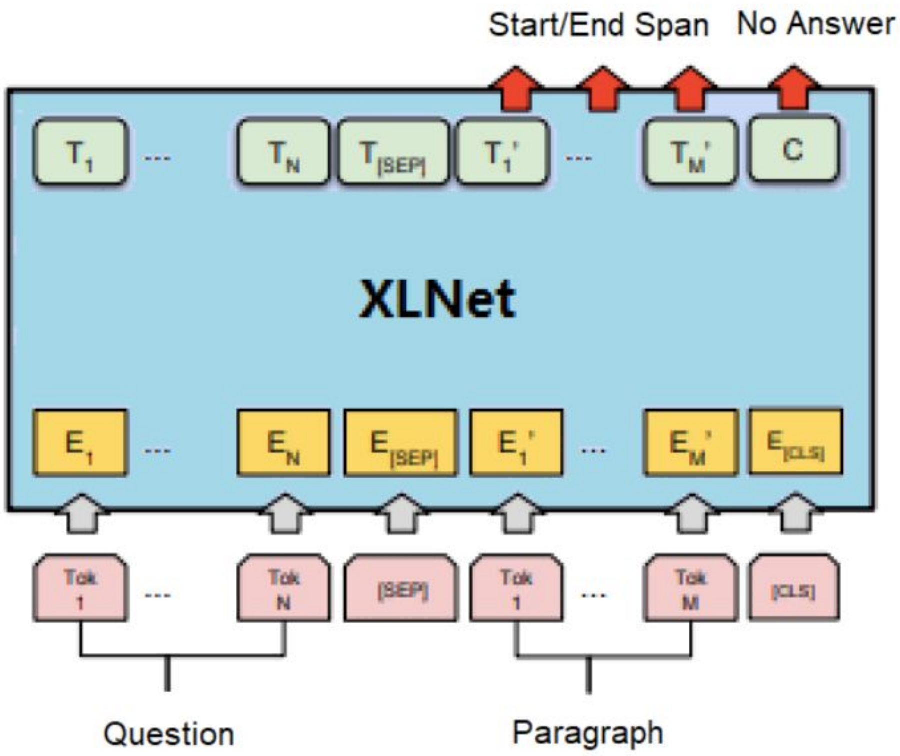


Almost identical training recipes.

Challenges: Scalability of XLNet

Memory	Time
128GB GPU/TPU (Google TPU v3-8)	Days

Question Answering Fine-tuning



Default Parameters

- Batch size: 8
- Max Sequence length: 512
- Fine-tuned layers: 24 out of 24
- Output Layers
  - Start: 1024 -> 1
  - End: 1024 -> 1
  - Class: 1024 -> 1

SQuAD 1.1 & 2.0

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

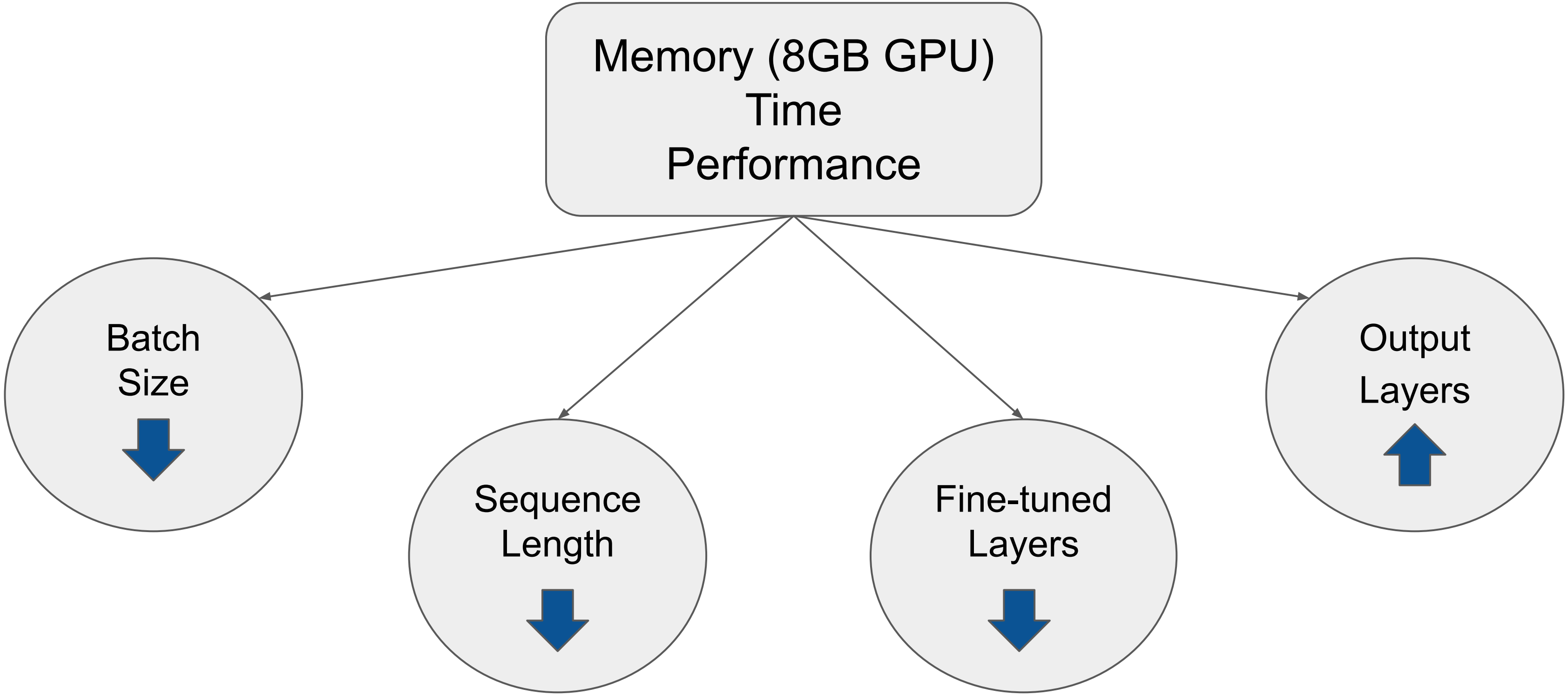
Paragraph: " *... Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940. These later laws had a low cost to society—the species were relatively rare—and little opposition was raised.*"

Question 1: "*Which laws faced significant opposition?*"
Plausible Answer: *later laws*

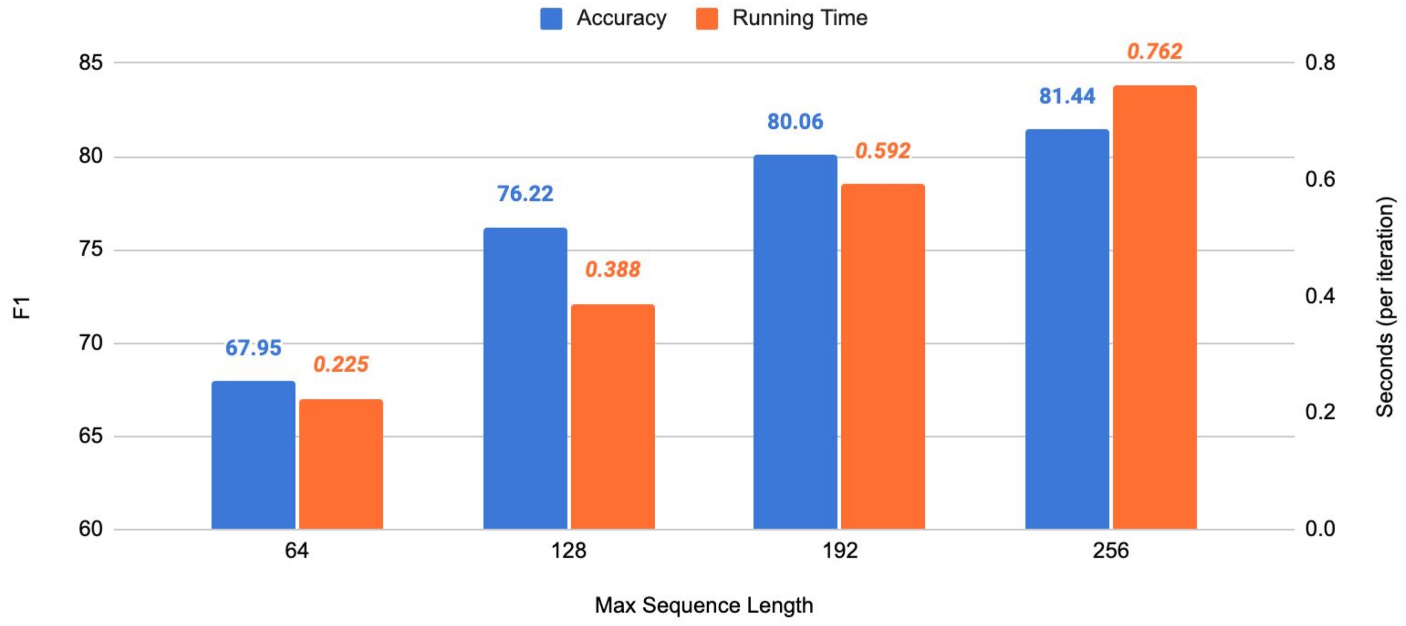
Question 2: "*What was the name of the 1937 treaty?*"
Plausible Answer: *Bald Eagle Protection Act*

✻ **100,000 answerable + 50,000 unanswerable questions**

Alternatives

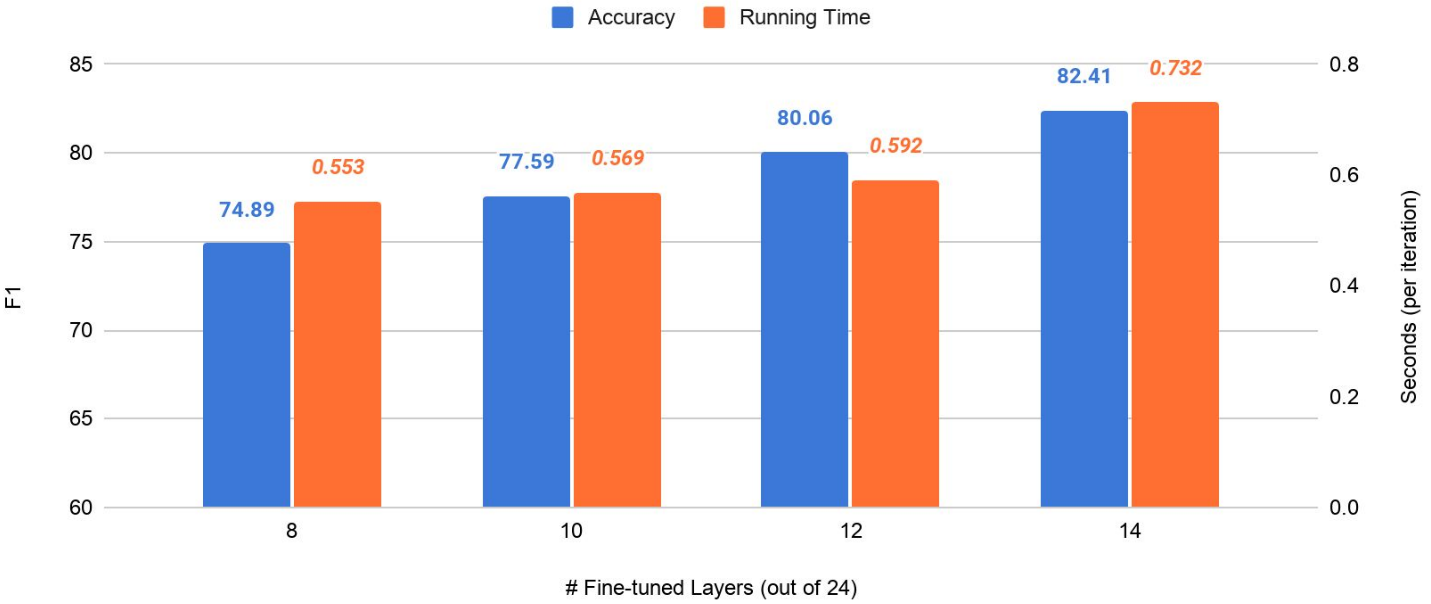


Experiment: Sequence Length



The figure shows the change in F1 score and running time when max sequence length varies. Each experiment only runs once with 12000 train-steps.

Experiment: Partial Fine-tuning



The figure shows the change in F1-score and running time when the number of trained layers varies. Each experiment only runs once with 12000 train-steps and max sequence length is set to 192.

Experiment: Output Layers

The figure shows the change in F1-score and running time when output layers vary. Each experiment only runs once with 12000 train-steps, max sequence length is set to 192 and the number of trained layers is 14.

Conclusion

- XLNet is an ***autoregressive*** model that naturally captures ***bidirectional*** **context**, and has shown ***superior performance*** to autoencoding models, such as BERT and RoBERTa.

- We explore scalable alternatives to the question-answering fine-tuning to ***tackle memory and time challenges*** of XLNet. We have provided insights into ***tradeoffs*** between performance and running time with ***different sequence lengths***, ***partial fine-tuning***, and ***different output layers***.