Recent Progress in Pre-training for NLP

Furu Wei
Senior Principal Research Manager (首席研究员), MSRA\NLP
Presented at CCMT 2020

Joint work with my colleagues and interns at MSRA
Attention Is All You Need

Transformer (NeurIPS 2017)
NMT

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin  Ming-Wei Chang  Kenton Lee  Kristina Toutanova
Google AI Language
{jacobdevlin, mingweichang, kentonl, kristout}@google.com

BERT (NAACL 2019)
“NMT”
The Era of NLP & Pre-training

OpenAI GPT-3
May 2020
175 billion parameters

Google AI BERT
Oct. 2018
11 language tasks

The next stage of an artificial intelligence that was dubbed 'too dangerous to release'

Fundamentally changes how we build SOTA NLP (and NL related) models
New NLP Paradigm: Pre-training + Fine-tuning

Downstream NLP Tasks: text classification, entity recognition, question answering, summarization, ...

Pre-trained Model

Task Datasets

End-to-End Neural Models

predictions

Self-supervised Learning

Massive-scale Unlabeled Corpora

Pre-training

Fine-tuning

Supervised Learning

Task Datasets

Labeled dataset per task (e.g., 1K samples)

predictions

Small task-specific models (e.g., 10K parameters)

One big pre-trained model (e.g., 340M/10B parameters) across downstream tasks
## Transfer Learning for NLP Tasks

### Language skills
- Classify text into topics or sentiments
- Extract entities from text
- Find answers to questions from text
- Summarize long text (e.g., a document)
- ... 

Understand the **meaning** of text (e.g., words and their relations) in **context**

### NLP tasks
- Text classification
- Entity recognition
- Question answering
- Text summarization
- ...

**Contextualized representations (vectors)**

**Self-supervised learning (Masked LM)**
Massive-scale self-supervised learning via cloze tests

An apple is a sweet, edible ____ produced by an apple tree.

fruit ?
company ?
...

Language Modeling (LM) / Cloze tests as pre-training tasks/objectives
An apple is a sweet, edible fruit produced by an apple tree.
**Key trends**

- Big model, large corpora
- Unified model for NLU, NLG and multilingual tasks (incl. MT)
- Novel frameworks, pre-training tasks and network structures
The Anatomy of Language Model Pre-training
The Anatomy of Language Model Pre-training

Backbone Networks
(e.g. Multilayer Transformer)
Architecture: enc/unified, dec, enc-dec

NLP Tasks

Pre-training Tasks
Self-supervised Tasks
(e.g. Cloze / Masked LM)
NLP Tasks & Neural NLP Architectures

NLP Tasks

NLU  \{ \text{NL} \Rightarrow X (\neq \text{NL}) \}  
Text \text{(Language E)} \Rightarrow \{ \text{Class, Sequence of class, Score, ...} \}

NLG  \{ X (\neq \text{audio/image/video}) \Rightarrow \text{NL} \}

- NLG (LM)  \{ \} \Rightarrow \text{Text (Language E)}
- NLG (S2S)  \text{Text (Language E)} \Rightarrow \text{Text (Language E)}
- NLG (XL)/MT  \text{Text (Language E)} \Rightarrow \text{Text (Language F)}
Pre-training Tasks: Language Modeling

Causal Language Modeling (CLM)
Predict the next word/token: predict \( w_i \) at the index of \( i-1 \) (using the hidden state of \( w_{i-1} \))

ELMo/GPT(-2/3)

Masked Language Modeling (MLM)
Predict the current word/token: predict \( w_i \) at the index of \( i \) (using the hidden state of the placeholder (or masked token) of \( w_i \))

BERT/RoBERTa/XLNet/UniLM ...

Unidirectional LM

Bidirectional LM

The scope of the prediction over context tokens and among context tokens
Pre-training Tasks: Masked Language Modeling

Dependency among masked tokens during prediction

Autoencoding (AE)

BERT/RoBERTa

Autoregressive (AR)

XLNet

Partially autoregressive (PAR)

(UniLMv2)

Sequence-to-Sequence LM
Pre-training Tasks: Masked Language Modeling

Denoising Autoencoder (DAE)

T5

\[ x_1 \quad [M] \quad x_3 \quad [M] \quad x_6 \]

\[ x_2 \quad [M] \quad x_4 \quad x_5 \quad [M] \]

BART

\[ x_1 \quad [M] \quad x_3 \quad [M] \quad [M] \quad x_6 \]

\[ x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \]

Sequence-to-Sequence LM
NLP Tasks & Pre-training Tasks

**NLU**  
\{ NL \rightarrow X (\neq NL) \}  
Text (Language E) \rightarrow \{ Class, Sequence of class, Scores, … \}

**Bidirectional**

**NLU**  
\{ NL \rightarrow X (\neq NL) \}  
Text (Language E) \rightarrow \{ Class, Sequence of class, Scores, … \}

**Bidirectional LM**

- AE
- AR
- PAR

**Unidirectional**

**NLU**  
\{ NL \rightarrow X (\neq audio/image/video) \rightarrow NL \}

**Unidirectional LM**

- CLM

**NLG**  
\{ X (\neq audio/image/video) \rightarrow NL \}

**Bidirectional + Unidirectional**

**NLG (LM)**  
Text (Language E) \rightarrow Text (Language E)

**Bidirectional + Unidirectional**

**NLG (S2S)**  
\{ \} \rightarrow Text (Language E)

**Sequence-to-Sequence LM**

- DAE
- AR/PAR

**NLG (XL)/MT**  
Text (Language E) \rightarrow Text (Language F)

**Context Modeling**

MLM: Masked Language Modeling  
CLM: Causal Language Modeling  
DAE: Denoising Autoencoder
Backbone Networks: Multilayer Transformers

**Multi-Head Attention**

\[
\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O
\]

where \( \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \)

**Scaled Dot-Product Attention**

\[
\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V
\]
The Anatomy of Language Model Pre-training

Architectures
- enc/unified, dec, enc-dec
Backbone Networks
- Transformer
Pre-training tasks
- MLM

1. dec 2. enc 3. enc/dec 4. enc-dec
UniLM: Unified Language Model Pre-training
**UniLM**: Unified Language Model Pre-training

* UniLMv1: (NeurIPS’19)
* UniLMv2: (ICML’20)
* InfoXLM: (Multilingual/Cross-lingual UniLM)
* UniLMv3: (?)
* UniLMv4: (?)

Unified Language Model Pre-training

**NLP Tasks**

**NLU**  \{ NL => X (≠ NL) \}  
Text (Language E) => \{ Class, Sequence of class, Score, ... \}

**NLG**  \{ X (≠ audio/image/video) => NL \}

- **NLG (LM)**  \{ \} => Text (Language E)
- **NLG (S2S)**  Text (Language E) => Text (Language E)
- **NLG (XL)/MT**  Text (Language E) => Text (Language F)

**Pre-training Tasks**

**Bidirectional LM**

- **Mlm**
- **Clm**

**Unidirectional LM**

- **1/2/3**

**Sequence-to-Sequence LM**

- **S T5, BART**
- **PAR**
Backbone Networks: Multilayer Transformer

Self-attention Masks
0: allow to attend
−∞: prevent from attending

Bidirectional Encoder
NLU / Bidirectional LM

Unidirectional Decoder
NLG (LM) / Unidirectional LM

Bidirectional Encoder + Unidirectional Decoder
NLG (S2S/XL) / Sequence-to-Sequence LM

NLP Tasks / Pre-training Tasks
Overview of the Unified Pre-training Framework in UniLM. We use different self-attention masks to control the access to context for each word token in different variants of LMs in pre-training and different NLU/NLG tasks in fine-tuning.

1 Unified Modeling

2 Unified (Multitask-Style) Pre-training

3 Unified Fine-tuning

**NLU**: text classification, entity recognition, question answering, ...

**NLG**: text generation, ...

**NLG (sequence-to-sequence)**: text summarization, question generation, ...

BIDIRECTIONAL ENCODER

UNIDIRECTIONAL DECODER
Transformer Block & Self-Attention

Transformer Block

Feed Forward

Feed Forward

Self-attention Module

h₁ h₂ h₃ h₄ h₅

Attention Output = AV
A = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})
Q = H^{l-1}W^Q, \ K = H^{l-1}W^K, \ V = H^{l-1}W^V

Weighted Sum

Dot-product

Queries (Q)

Keys (K)

Values (V)

Linear

Linear

Linear

Previous layer’s output \(H^{l-1}\)
Self-Attention Masks

\[
Q = H_{L-1}W^Q_L \quad K = H_{L-1}W^K_L \quad V = H_{L-1}V_L
\]

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

\[
A = \text{softmax}(\frac{QK^T}{\sqrt{d_k}} + M)V
\]

\[
M_{ij} = \begin{cases} 
0, & \text{allow to attend} \\
-\infty, & \text{prevent from attending}
\end{cases}
\]

\[
M = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\infty & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]
Self-Attention Masks for UniLM

Bidirectional LM

Unidirectional LM

Sequence-to-Sequence LM

The same unified pre-trained model

NLU (Natural Language Understanding)

NLG (Unconditional Language Generation)

NLG (Sequence-to-Sequence Generation)
Motivation of UniLMv2

(v1) One training example for each type of LM
• Three types of LMs
• Three forward passes with different self-attention masks

How to train multiple LMs in one forward pass?

Training Batch

Training Example

Training Example

Training Example

Bidirectional LM

Unidirectional LM

Sequence-to-Sequence LM

UniLM
Pseudo-Masked Language Model

Bidirectional LM Task
1. Bidirectionally encode context tokens
2. Predict the masked spans at the same time

Sequence-to-Sequence LM Task
1. Bidirectionally encode context tokens
2. Predict the masked spans one by one
   1. Predict $x_4, x_5$
   2. Encode $x_4, x_5$ (i.e., fill in what we have predicted)
   3. Predict $x_2$
Pseudo-Masked Language Model

Observation #1: context encoding can be reused

**Bidirectional LM Task**
1. Bidirectionally encode context tokens
2. Predict the masked spans at the same time

**Sequence-to-Sequence LM Task**
1. Bidirectionally encode context tokens
2. Predict the masked spans one by one
   1. Predict $x_4, x_5$
   2. Encode $x_4, x_5$ (i.e., fill in what we have predicted)
   3. Predict $x_2$
Pseudo-Masked Language Model

Bidirectional LM Task
1. Bidirectionally encode context tokens
2. Predict the masked spans at the same time

Sequence-to-Sequence LM Task
1. Bidirectionally encode context tokens
2. Predict the masked spans one by one
   1. Predict $x_4, x_5$
   2. Encode $x_4, x_5$ (i.e., fill in what we have predicted)
   3. Predict $x_2$

Observation #1: context encoding can be reused
Observation #2: masked positions have three roles

(1) Context masks [M]
(2) Pseudo masks [P]
(3) Original tokens
Pseudo-Masked Language Model

• A general framework to
  • **efficiently** realizes different pre-training objectives
    • AE (autoencoding): BERT/RoBERTa
    • AR (autoregressive): XLNet
    • PAR (partially autoregressive): New in UniLMv2
    • AE + AR: BERT + XLNet
    • **AE + PAR: UniLMv2 (unified pre-training), performs the best**
  • **effectively** learn different word dependencies
    • Between context and mask predictions
    • Between mask predictions

in **one forward pass**
UniLM (v2): Unified pre-training of bi-directional LM (via autoencoding) and sequence-to-sequence LM (via partially autoregressive) with Pseudo-Masked Language Model for language understanding and generation

- Transformer/Self-attention treats tokens with the same position embeddings as the same “token” at that position
- Pseudo-masked LM can be used to efficiently realize different pre-training objectives, such as AE (autoencoding), AR (autoregressive), PAR (partially autoregressive), AE + AR, and AE + PAR, among which AE + PAR performs the best
Self-Attention Masks for UniLMv2

Unified pre-training of bi-directional LM (via autoencoding) and sequence-to-sequence LM (via partially autoregressive) with one training sample (forward pass)
# UniLMv2 Base for NLU Tasks

<table>
<thead>
<tr>
<th>Model</th>
<th>SQuAD v1.1</th>
<th>SQuAD v2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>EM</td>
</tr>
<tr>
<td>BERT</td>
<td>88.5</td>
<td>80.8</td>
</tr>
<tr>
<td>XLNet</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>RoBERTa</td>
<td>91.5</td>
<td>84.6</td>
</tr>
<tr>
<td>UniLMv2</td>
<td><strong>93.1</strong></td>
<td><strong>87.1</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>MNLI</th>
<th>SST-2</th>
<th>MRPC</th>
<th>RTE</th>
<th>QNLI</th>
<th>QQP</th>
<th>STS</th>
<th>CoLA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
<td>Acc</td>
</tr>
<tr>
<td>BERT</td>
<td>84.5</td>
<td>93.2</td>
<td>87.3</td>
<td>68.6</td>
<td>91.7</td>
<td>91.3</td>
<td>89.5</td>
<td>58.9</td>
</tr>
<tr>
<td>XLNet</td>
<td>86.8</td>
<td>94.7</td>
<td>88.2</td>
<td>74.0</td>
<td>91.7</td>
<td>91.4</td>
<td>89.5</td>
<td>60.2</td>
</tr>
<tr>
<td>RoBERTa</td>
<td>87.6</td>
<td>94.8</td>
<td>90.2</td>
<td>78.7</td>
<td>92.8</td>
<td><strong>91.9</strong></td>
<td><strong>91.2</strong></td>
<td>63.6</td>
</tr>
<tr>
<td>UniLMv2</td>
<td><strong>88.5</strong></td>
<td><strong>95.1</strong></td>
<td><strong>91.8</strong></td>
<td><strong>81.3</strong></td>
<td><strong>93.5</strong></td>
<td>91.7</td>
<td>91.0</td>
<td><strong>65.2</strong></td>
</tr>
</tbody>
</table>

Results of **BASE-size** pre-trained models on the **SQuAD v1.1/v2.0** development sets. We report F1 scores and exact match (EM) scores. Results of UniLMv2 are averaged over five runs.

Results of **BASE-size** models on the development set of the **GLUE benchmark**. We report Matthews correlation coefficient (MCC) for CoLA, Pearson correlation coefficient (PCC) for STS, and accuracy (Acc) for the rest. Metrics of UniLMv2 are averaged over five runs for the tasks.
UniLMv2 Base/Large for NLG Tasks

Outperforming T5(11B), BART, Pegasus, w/ 3% of T5 parameters, 20% of T5 (4% of Pegasus) data

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Without pre-training</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PTRNet (See et al., 2017)</td>
<td>-</td>
<td>-</td>
<td>39.53/17.28/36.38</td>
<td>28.10/8.02/21.72</td>
<td>-</td>
</tr>
<tr>
<td><strong>Fine-tuning BASE-size pre-trained models</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MASS\textsubscript{BASE} (Song et al., 2019)</td>
<td>123M</td>
<td>-</td>
<td>42.12/19.50/39.01</td>
<td>39.75/17.24/31.95</td>
<td>38.73/19.71/35.96</td>
</tr>
<tr>
<td>BERTSUM\textsubscript{ABS} (Liu &amp; Lapata, 2019)</td>
<td>156M</td>
<td>16GB</td>
<td>41.72/19.39/38.76</td>
<td>38.76/16.33/31.15</td>
<td>-</td>
</tr>
<tr>
<td>ERNIE-GEN\textsubscript{BASE} (Xiao et al., 2020)</td>
<td>110M</td>
<td>16GB</td>
<td>42.30/19.92/39.68</td>
<td>39.74/17.24/31.95</td>
<td>38.83/20.04/36.20</td>
</tr>
<tr>
<td>T5\textsubscript{BASE} (Raffel et al., 2019)</td>
<td>220M</td>
<td>750GB</td>
<td>42.05/20.34/39.40</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>UniLMv2</td>
<td>110M</td>
<td>160GB</td>
<td>43.89/21.05/41.02</td>
<td>44.67/21.78/36.81</td>
<td>-</td>
</tr>
<tr>
<td><strong>Fine-tuning LARGE-size pre-trained models</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UniLMv2\textsubscript{LARGE} (Dong et al., 2019)</td>
<td>340M</td>
<td>16GB</td>
<td>43.08/20.43/40.34</td>
<td>-</td>
<td>38.90/20.05/36.00</td>
</tr>
<tr>
<td>ERNIE-GEN\textsubscript{LARGE} (Xiao et al., 2020)</td>
<td>340M</td>
<td>16GB</td>
<td>44.02/21.17/41.26</td>
<td>-</td>
<td>39.25/20.25/36.53</td>
</tr>
<tr>
<td>BART\textsubscript{LARGE} (Lewis et al., 2019)</td>
<td>400M</td>
<td>160GB</td>
<td>44.16/21.28/40.90</td>
<td>45.14/22.27/37.25</td>
<td>-</td>
</tr>
<tr>
<td>ProphetNet (Yan et al., 2020)</td>
<td>400M</td>
<td>160GB</td>
<td>44.20/21.17/41.30</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>PEGASUS\textsubscript{C4} (Zhang et al., 2019)</td>
<td>568M</td>
<td>750GB</td>
<td>43.90/21.20/40.76</td>
<td>45.20/22.06/36.99</td>
<td>38.75/19.96/36.14</td>
</tr>
<tr>
<td>PEGASUS\textsubscript{HUGENews} (Zhang et al., 2019)</td>
<td>568M</td>
<td>3800GB</td>
<td>44.17/21.47/41.11</td>
<td>47.21/24.56/39.25</td>
<td>39.12/19.86/36.24</td>
</tr>
<tr>
<td>T5\textsubscript{11B} (Raffel et al., 2019)</td>
<td>11B</td>
<td>750GB</td>
<td>43.52/21.55/40.69</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>UniLMv2</td>
<td>340M</td>
<td>160GB</td>
<td>44.79/21.98/41.93</td>
<td>47.58/24.35/39.50</td>
<td>39.73/20.70/36.89</td>
</tr>
</tbody>
</table>

Abstractive summarization results on CNN/DailyMail and XSum. The evaluation metric is the F1 version of ROUGE (RG) scores. We also include the number of parameters (#Param) for the methods using pre-trained models.
Question generation results on SQuAD v1.1. MTR is short for METEOR, and RG for ROUGE. The official split is from (Du & Cardie, 2018), while the reversed split is the same as in (Zhao et al., 2018).

<table>
<thead>
<tr>
<th>Model</th>
<th>#Param</th>
<th>Corpus</th>
<th>Official Split</th>
<th>Reversed Split</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>BLEU-4 / MTR / RG-L</td>
<td>BLEU-4 / MTR / RG-L</td>
</tr>
<tr>
<td><strong>Without pre-training</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Du &amp; Cardie, 2018)</td>
<td>-</td>
<td>-</td>
<td>15.16 / 19.12 / -</td>
<td>-</td>
</tr>
<tr>
<td>(Zhao et al., 2018)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16.38 / 20.25 / 44.48</td>
</tr>
<tr>
<td>(Zhang &amp; Bansal, 2019)</td>
<td>-</td>
<td>-</td>
<td>18.37 / 22.65 / 46.68</td>
<td>20.76 / 24.20 / 48.91</td>
</tr>
<tr>
<td><strong>Fine-tuning BASE-size pre-trained models</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERNIE-GEN_BASE (Xiao et al., 2020)</td>
<td>110M</td>
<td>16GB</td>
<td>22.28 / 25.13 / 50.58</td>
<td>23.52 / 25.61 / 51.45</td>
</tr>
<tr>
<td>UniLMv2</td>
<td>110M</td>
<td>160GB</td>
<td><strong>24.70 / 26.33 / 52.13</strong></td>
<td><strong>26.30 / 27.09 / 53.19</strong></td>
</tr>
<tr>
<td><strong>Fine-tuning LARGE-size pre-trained models</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UniLMv1_LARGE (Dong et al., 2019)</td>
<td>340M</td>
<td>16GB</td>
<td>22.12 / 25.06 / 51.07</td>
<td>23.75 / 25.61 / 52.04</td>
</tr>
<tr>
<td>ERNIE-GEN_LARGE (Xiao et al., 2020)</td>
<td>340M</td>
<td>16GB</td>
<td>24.03 / 26.31 / 52.36</td>
<td>25.57 / 26.89 / 53.31</td>
</tr>
<tr>
<td>ProphetNet (Yan et al., 2020)</td>
<td>400M</td>
<td>16GB</td>
<td>25.01 / 26.83 / 52.57</td>
<td>26.72 / 27.64 / 53.79</td>
</tr>
<tr>
<td>UniLMv2</td>
<td>340M</td>
<td>160GB</td>
<td><strong>25.97 / 27.33 / 53.43</strong></td>
<td><strong>27.12 / 27.95 / 54.25</strong></td>
</tr>
</tbody>
</table>
UniLM: Unified Language Model Pre-training

UniLMv1  
(NeurIPS’19)

UniLMv2  
(ICML’20)

InfoXLM  
(Multilingual/Cross-lingual UniLM)

UniLMv3  
(?)

UniLMv4  
(?)

InfoXLM – Cross-lingual Language Model Pre-training

- Multilingual Masked Language Modeling (MMLM)
- Translation Language Modeling (TLM)
- Cross-lingual Contrast (XLCo)

Multilingual text

Bilingual text (parallel corpora)

Unified Information Theoretic Framework
Maximize the InfoNCE (Oord, 2018) lower bound of mutual information $I(x_{en}; x_{fr})$

Cross-lingual Contrast (XLCo)

Task: whether they are parallel sentences

Queue of negative examples

InfoXLM Encoder

Momentum Copy

InfoXLM Encoder

$x_{en}$

Parallel Text

(xCross-lingual views of the same meaning)

$x_{fr}$
Because parallel sentences are always in different languages, the model easily knows $y_{en}$ is a negative example of $x_{en}$.

- Problem: sentences in the same language are “dissimilar” even if they have similar meanings.

- Solution: mixup contrast
  - Randomly mix up translation pairs and shuffle their orders.
  - Encourage multilingual representations blend in together.

Mixup Contrast
Contrast on Universal Layer

• Multilingual masked language modeling encourages bottleneck structure
  • Language-specific (bottom layers) -> language-agnostic (middle layers) -> language-specific (upper layers)
• Contrastive learning is put on the middle layer to avoid violating the universal space
InfoXLM/T-ULRv2 Achieves #1 on XTREME Benchmark

- Cover four paradigms: classification, structured prediction, question answering, retrieval
- Average over 9 datasets, and 40 languages

XTREME Benchmark (CMU & Google): https://sites.research.google/xtreme/
Pre-training for NLP: Revisit
BUILDING SOTA PRE-TRAINED MODELS

Modeling
Network structures, pre-training tasks & objectives

Data
Massive-scale (unlimited) unlabeled corpora

System & Infra
Large-scale training & optimization
#0 - big model, large corpora

- **T-NLG**: 17B, ~174G
- **GPT-3**: 175B, ~950G
- **T5**: 11B, 750G
- **BERT**: 340M, 16G
- **RoBERTa**: 340M, 160G
#1 - Pre-training tasks: Masked Language Modeling

DAE (enc-dec), TLM (cross-lingual), ...

**Masked LM and the Masking Procedure**  Assuming the unlabelled sentence is *my dog is hairy*, and during the random masking procedure we chose the 4-th token (which corresponding to hairy), our masking procedure can be further illustrated by 15% of tokens

- 80% of the time: Replace the word with the [MASK] token, e.g., *my dog is hairy* → *my dog is [MASK]*
- 10% of the time: Replace the word with a random word, e.g., *my dog is hairy* → *my dog is apple*
- 10% of the time: Keep the word unchanged, e.g., *my dog is hairy* → *my dog is hairy*. The purpose of this is to bias the representation towards the actual observed word.

An [MASK] is a sweet, edible [MASK] produced by an apple [MASK]

A cloze test (also cloze deletion test) is an exercise, test, or assessment consisting of a portion of language with certain items, words, or signs removed (cloze text), where the participant is asked to replace the missing language item. Cloze tests require the ability to understand context and vocabulary in order to identify the correct language or part of speech that belongs in the deleted passages. This exercise is commonly administered for the assessment of native and second language learning and instruction.

The word cloze is derived from closure in Gestalt theory. The exercise was first described by W.L. Taylor in 1953.[1]

A language teacher may give the following passage to students:

Today, I went to the ______ and bought some milk and eggs. I knew it was going to rain, but I forgot to take my ______, and ended up getting wet on the way.

Students would then be required to fill in the blanks with words that would best complete the passage. Context in language and content terms is essential in most, if not all, cloze tests. The first blank is preceded by "the"; therefore, a noun, an adjective or an adverb must follow. However, a conjunction follows the blank; the sentence would not be grammatically correct if anything other than a noun were in the blank. The words "milk and eggs" are important for deciding which noun to put in the blank; "supermarket" is a possible answer; depending on the student, however, the first blank could be store, supermarket, shop, shops, market, or grocer while umbrella, brolly or raincoat could fit the second.


“Cloze Procedure”: A New Tool for Measuring Readability ...
Cloze Test

Taylor (1953) first suggested the cloze procedure for determining the difficulty, or “readability,” of a text — reasoning that if several people could reproduce the missing words of a “mutilated” passage than the text must be easy to read, but if they could not supply the missing words the text must be difficult.

It was not long before Taylor himself saw the potential to use cloze as a measure of reading comprehension. In a 1956 paper, he reasoned that “if the statement that a passage is ‘readable’ means that it is ‘understandable,’ then the scores that measure readability should measure comprehension too”.

Further research has demonstrated that cloze tests tend to correlate well with almost every kind of language test. Traditional cloze tests were found to correlate not only with reading comprehension tests but also with dictation, listening comprehension, structure/grammar, and vocabulary tests.

Results such as these are what led Oller and Conrad (1971) and others to mark cloze procedures as integrative tests and as good measures of general language proficiency rather than the more specific reading comprehension alone.

Gestalt psychologists emphasized that organisms perceive entire patterns or configurations, not merely individual components. The view is sometimes summarized using the adage, "the whole is more than the sum of its parts."

Gestalt psychologists believed that humans tend to perceive objects as complete rather than focusing on the gaps that the object might contain. For example, a circle has good Gestalt in terms of completeness. However, we will also perceive an incomplete circle as a complete circle. That tendency to complete shapes and figures is called closure. The law of closure states that individuals perceive objects such as shapes, letters, pictures, etc., as being whole when they are not complete. Specifically, when parts of a whole picture are missing, our perception fills in the visual gap.

* https://en.wikipedia.org/wiki/Gestalt_psychology
#2 - Backbone Networks: Multilayer Transformers

Relative position, self-attention, ...

Multi-Head Attention

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_h)W^O$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$

Scaled Dot-Product Attention

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$
Future Development of (LM) Pre-training

• **Pre-training tasks**
  • Parameterized self-supervised tasks

• **Network structures**
  • Improved (e.g. better/fast) Transformer & beyond

• Learning efficiency
  • Data/sample selection and planning (curriculum learning)

• Big & small pre-trained models
  • Large-scale training & optimization
  • The “ceiling” of small models

NMT & Pre-training
Pre-training for MT

Can we beat the BT baseline?
Low-resource?

|  

MT for Pre-training

Can we have better pre-training tasks besides TLM/XLCo?
Can BT be used for multilingual LM pre-training?
Thanks