BERT and its family

Hung-yi Lee 李宏毅
A model that can read text

Pre-train

Text without annotation

Pre-train

Task-specific data with annotation

Fine-tune

Task Specific

Model

Task Specific

Model

Task Specific

Model
Big Bird
Big Binary Recursive Decoder?

ELMo
(Embeddings from Language Models)

ERNIE (Enhanced Representation through Knowledge Integration)

Grover (Generating Articles by Only Viewing Metadata Records)

BERT (Bidirectional Encoder Representations from Transformers)

BERT & PALs (Projected Attention Layers)

STAYREAL
Outline

What is pre-train model

How to fine-tune

How to pre-train
Pre-train Model
Pre-train Model

Represent each token by a embedding vector

The token with the same type has the same embedding.

Simply a table look-up

Word2vec [Mikolov, et al., NIPS’13]
Glove [Pennington, et al., EMNLP’14]
Pre-train Model
Represent each token by a embedding vector

The token with the same type has the same embedding.

English word as token ...

FastText
[Bojanowski, et al., TACL’17]
Pre-train Model

Represent each token by a embedding vector

The token with the same type has the same embedding.

Chinese character as token ...
Pre-train Model
Represent each token by a embedding vector

養
隻
狗

同一
單
身
狗
Pre-train Model
Represent each token by an embedding vector
different
Pre-train Model
Contextualized Word Embedding
Pre-train Model
Contextualized Word Embedding

Many Layers

- LSTM
- Self-attention layers
- Tree-based model (?)
  - Ref: https://youtu.be/z0uOq2wEGcc
Bigger Model


Megatron

[Shoeybi, et al., arXiv’19]
Smaller Model

- Distill BERT [Sanh, et al., NeurIPS workshop'19]
- Tiny BERT [Jian, et al., arXiv’19]
- Mobile BERT [Sun, et al., ACL’20]
- Q8BERT [Zafrir, et al., NeurIPS workshop 2019]
- ALBERT [Lan, et al., ICLR’20]
Smaller Model

- Network Compression
  - Network Pruning
  - Knowledge Distillation
  - Parameter Quantization
  - Architecture Design

Ref: https://youtu.be/dPp8rCAnU_A

Excellent reference:

All of them have been tried.
Network Architecture

• Transformer-XL: Segment-Level Recurrence with State Reuse  [Dai, et al., ACL’19]

• Reformer  [Kitaev, et al., ICLR’20]

• Longformer  [Beltagy, et al., arXiv’20]

Reduce the complexity of self-attention
How to fine-tune

Pre-trained Model

Task-specific Layer

For a specific NLP task
NLP tasks

Input
- one sentence
- multiple sentences

Output
- one class
- class for each token
- copy from input
- general sequence
Input

one sentence
multiple sentences

Sentence 1
Query
Premise

[SEP]

Sentence 2
Document
Hypothesis
Output

Task Specific

Task Specific

class

class

one class

class for each token

copy from input

general sequence
One class for each token
Copy from input
General sequence

Output

Model

[CLS]

\[ w_1 \quad w_2 \quad w_3 \]
Output

• Extraction-based QA

**Document:** \[ D = \{d_1, d_2, \ldots, d_N\} \]

**Query:** \[ Q = \{q_1, q_2, \ldots, q_M\} \]

\[ D \rightarrow \text{QA Model} \rightarrow s \]
\[ Q \rightarrow \text{QA Model} \rightarrow e \]

output: two integers \((s, e)\)

**Answer:** \[ A = \{d_s, \ldots, d_e\} \]

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 can also form 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**

Where do water droplets collide with ice crystals to form precipitation?
**within a cloud** \( s = 77, e = 79 \)
Copy from Input (BERT)

Task Specific

dot product

0.3

0.5

0.2

Model

[CLS] q₁ q₂ [SEP] d₁ d₂ d₃

question document

s = 2
Copy from Input (BERT)

\[ s = 2 \quad e = 3 \]

The answer is “d\textsubscript{2} d\textsubscript{3}”. 

Task
Specific

dot product

Model

\[ 0.1 \quad 0.2 \quad 0.7 \]

Softmax

\[ d\textsubscript{1} \quad d\textsubscript{2} \quad d\textsubscript{3} \]

question
document
Output – General Sequence (v1)

- Seq2seq model

![Diagram showing a Seq2seq model with input sequence, encoder, attention, and task-specific decoder leading to an output sequence. The model includes tokens w₁, w₂, w₃, and w₄, w₅, <EOS>.]
Output – General Sequence (v2)

Model

input sequence

w1 w2 [SEP] w3 w4 w5 <EOS>

output sequence

Task Specific Task Specific Task Specific Task Specific
How to fine-tune

**Task-specific**

**Pre-trained**

Fine-tune

Pre-trained Model

Feature Extractor (Fix)

A gigantic model for down-stream tasks
Adaptor [Stickland, et al., ICML’19] [Houlsby, et al., ICML’19]

![Diagram of Adaptor model with tasks and models](image)

- Task Specific Model
- Task Specific Model
- Task Specific Model

Fine-tune

- Task Specific Model
- Task Specific Model
- Task Specific Model

Large Memory is needed
Adaptor

[Stickland, et al., ICML’19] [Houlsby, et al., ICML’19]

Task Specific

Model

Apt

Task Specific

Model

Apt

Task Specific

Model

Apt

Fine-tune

Task Specific

Model

Apt

Task Specific

Model

Apt

Task Specific

Model

Apt

[Houlsby, et al., ICML’19]

[Houlsby, et al., ICML’19]
Weighted Features

$w_1 x^1 + w_2 x^2$

$w_1$ and $w_2$ are learned in down-stream tasks

Layer 1

Layer 2

Whole Model

Task Specific
Why Pre-train Models?

- GLUE scores

Why Fine-tune?

Why Fine-tune?

How to generate the figures below?
https://youtu.be/XysGHdNOTbg

How to Pre-train

A model that can read text

Text without annotation

Pre-train

Model
Pre-training by Translation

- Context Vector (CoVe)

Input: A language

Model

Decoder

Output: B language

Need sentences pairs for languages A and B
Self-supervised Learning

I now call it "self-supervised learning", because "unsupervised" is both a loaded and confusing term.

In self-supervised learning, the system learns to predict part of its input from other parts of it input. In other words a portion of the input is used as a supervisory signal to a predictor fed with the remaining portion of the input.
Predict Next Token

\[ w_2, w_3, w_4, w_5 \]

\[ h_1, h_2, h_3, h_4 \]

Cross entropy

softmax

Linear Transform

from \( w_t \)

\[ h_t, w_{t+1} \]
Predict Next Token

This is exactly how we train language models (LM).

Universal Language Model Fine-tuning (ULMFiT)
[Howard, et al., ACL’18]

ELMo
[Peters, et al., NAACL’18]
Predict Next Token

<table>
<thead>
<tr>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$w_3$</th>
<th>$w_4$</th>
</tr>
</thead>
</table>

Self-attention

with constraint

GPT
[Alec, et al., 2018]

GPT-2
[Alec, et al., 2019]

Megatron
[Shoeybi, et al., arXiv’19]

Turing NLG
Predict Next Token
They can do generation.

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

The scientist named the population, after their distinctive horn, Ovid’s Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.
They can do generation.

I forced a bot to watch over 1,000 hours of Batman movies and then asked it to write a Batman movie of its own. Here is the first page.
BATMAN

INT. TRADITIONAL BATCAVE

BATMAN stands next to his batmobile and uses his batcomputer. He’s sometimes Bruce Wayne sometimes Batman. Alltimes orphan.

BATMAN

This is now a safe city. I have punched a penguin into prison.

ALFRED, Batman’s loyal batler, carries a tray of goth ham.

ALFRED

Eat a dinner, Mattress Wayne.

An explosion explodes. THE JOKER and TWO-FACE enter the cave. Joker is a clown but insane. Two-Face is a man but attorney.

BATMAN

No! It is Two-Face and One-Face. They hate me for being a bat.

Batman throws Alfred at Two-Face. Two-Face flips Alfred like a coin. Alfred lands heads up which means Two-Face goes home.

BATMAN (CONT’D)

It is just you and I, the Joker. Bat versus clown. Moral enemies.
THE JOKER
I am such a freak. Society is bad.
You drink water, I drink anarchy.

BATMAN
I drink bats just like a bat would!

Batman looks around for his parents, but they are still dead. This makes him have anger. He fires a batrocket. The Joker deflects it with his sick sense of humor. A clownly power.

THE JOKER
I have never followed a rule. That is my rule. Do you follow? I don’t.

BATMAN
Alfred, give birth to Robin.

Alfred begins the process since it is his job. The Joker now has a present in his hand. He juggles it over to Batman.

THE JOKER
Happy batday, Birthman.

Batman opens the present since he’s a good guy. It contains a coupon for new parents, but is expired. This is a Joker joke.
I forced a bot to watch over 1,000 hours of Olive Garden commercials and then asked it to deliver the commercials. Here is the first page:

I forced a bot to watch over 1,000 episodes of Jerry Springer and then asked it to see the page. Here is the first page:

I forced a bot to watch over 1,000 hours of the Saw movies and then asked it to write a Saw movie of its own. Here is the first page.
You shall know a word by the company it keeps

John Rupert Firth

encoding $w_4$ and its left context

Model

$w_1$, $w_2$, $w_3$, $w_4$, $w_5$, $h_4$

How about the right context!?
Predict Next Token
- Bidirectional

ELMO
Masking Input

BERT

[Devlin, et al., NAACL’19]

Transformer
(no limitation on self-attention)

\[ \text{MASK} \]

(special token)

Random Token

Model

\[ w_1 \]
\[ w_2 \]
\[ w_3 \]
\[ w_4 \]
Masking Input

Using context to predict the missing token
Masking Input

Is random masking good enough?

• Whole Word Masking (WWM)  [Cui, et al., arXiv’19]

[Original Sentence]
使用语言模型来预测下一个词的probability。  

[Original Sentence with CWS]
使用语言模型来预测下一个词的probability。  

Source of image:  

[Original BERT Input]
使用语言[MASK]型来[MASK]测下一个词的pro[MASK]#ility。  

[Whold Word Masking Input]
使用语言[MASK][MASK]来[MASK][MASK]下一个词的[MASK][MASK][MASK][MASK]。

• Phrase-level & Entity-level  
[Sun, et al., ACL’19]

Enhanced Representation through Knowledge Integration (ERNIE)
SpanBert

[Joshi, et al., TACL’20]

<table>
<thead>
<tr>
<th></th>
<th>SQuAD 2.0</th>
<th>NewsQA</th>
<th>TriviaQA</th>
<th>Coreference</th>
<th>MNLI-m</th>
<th>QNLI</th>
<th>GLUE (Avg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subword Tokens</td>
<td>83.8</td>
<td>72.0</td>
<td>76.3</td>
<td>77.7</td>
<td>86.7</td>
<td>92.5</td>
<td>83.2</td>
</tr>
<tr>
<td>Whole Words</td>
<td>84.3</td>
<td>72.8</td>
<td>77.1</td>
<td>76.6</td>
<td>86.3</td>
<td>92.8</td>
<td>82.9</td>
</tr>
<tr>
<td>Named Entities</td>
<td>84.8</td>
<td>72.7</td>
<td>78.7</td>
<td>75.6</td>
<td>86.0</td>
<td>93.1</td>
<td>83.2</td>
</tr>
<tr>
<td>Noun Phrases</td>
<td>85.0</td>
<td>73.0</td>
<td>77.7</td>
<td>76.7</td>
<td>86.5</td>
<td>93.2</td>
<td><strong>83.5</strong></td>
</tr>
<tr>
<td>Geometric Spans</td>
<td><strong>85.4</strong></td>
<td>73.0</td>
<td><strong>78.8</strong></td>
<td>76.4</td>
<td><strong>87.0</strong></td>
<td>93.3</td>
<td>83.4</td>
</tr>
</tbody>
</table>

SpanBert – Span Boundary Objective (SBO)
SpanBert – Span Boundary Objective (SBO)

Useful in coreference?

Span BERT
XLNet [Yang, et al., NeurIPS’19]

Transformer-XL

Model

深度1

深度2

学习3

学习4

Model

学习4

深度2

学习3

深度1
XLNet [Yang, et al., NeurIPS’19]

Transformer-XL

Model

深1 度2 學3 習4

Model

深1 學3 度2 習4
XLNet [Yang, et al., NeurIPS’19]

Transformer-XL
BERT cannot talk?

Given partial sequence, predict the next token

**LM-Style**

What LM born for

**BERT-style**

Never seen partial sequence

Limited to autoregressive model (non-autoregressive next time)
MASS / BART

• The pre-train model is a typical \textit{seq2seq} model.

MAsked Sequence to Sequence pre-training (MASS) [Song, et al., ICML’19]

Bidirectional and Auto-Regressive Transformers (BART) [Lewis, et al., arXiv’19]
Input Corruption

- Permutation / Rotation do not perform well.
- Text Infilling is consistently good.

Text Infilling
**BART/MASS**

![Diagram of BART/MASS model](image)

**UniLM**

![Diagram of UniLM model](image)

[Dong, et al., NeurIPS’19]
Unilm

Unified LM with Shared Parameters

Self-attention Masks

Replace or Not?
Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA)

Predicting yes/not is easier than reconstruction.
Every output position is used.
Note: This is not GAN.
Sentence Level

Representation for each token

Model

Representation for whole sequence
You shall know a **sentence** by the company it keeps?

**Skip Thought**

**Quick Thought**
In the original BERT, ..... NSP: Next sentence prediction

Robustly optimized BERT approach (RoBERTa)
[Liu, et al., arXiv’19]

SOP: Sentence order prediction

Used in ALBERT

structBERT (Alice) [Want, et al., ICLR’20]
# T5 – Comparison [Raffel, et al., arXiv’19]

- Transfer Text-to-Text Transformer (T5)
- Colossal Clean Crawled Corpus (C4)

<table>
<thead>
<tr>
<th>Objective</th>
<th>Inputs</th>
<th>Targets</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefix language modeling</td>
<td>Thank you for inviting me to your party last week.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BERT-style</td>
<td>Thank you &lt;M&gt; &lt;M&gt; me to your party apple week.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deshuffling</td>
<td>party me for your to . last</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I.i.d. noise, mask tokens</td>
<td>Thank you &lt;M&gt; &lt;M&gt; me to your party</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I.i.d. noise, replace spans</td>
<td>Thank you &lt;X&gt; me to your party</td>
<td></td>
<td></td>
</tr>
<tr>
<td>I.i.d. noise, drop tokens</td>
<td>Thank you me to your party your party</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random spans</td>
<td>Thank you &lt;X&gt; to &lt;Y&gt; we</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## High-level approaches
- Language modeling
- BERT-style
- Deshuffling

## Corruption strategies
- Mask
- Replace spans
- Drop

#### Corruption rate
- 10%
- 15%
- 25%
- 50%

#### Corrupted span length
- 2
- 3
- 5
- 10
Knowledge

• Enhanced Language Representation with Informative Entities (ERNIE)

This is another story ......
Audio BERT
This is another story ......
Reference


• [Song, et al., ICML’19] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, Tie-Yan Liu, MASS: Masked Sequence to Sequence Pre-training for Language Generation, ICML, 2019

Reference


• [Hao, et al., EMNLP’19] Yaru Hao, Li Dong, Furu Wei, Ke Xu, Visualizing and Understanding the Effectiveness of BERT, EMNLP, 2019


Reference


• [Dai, et al., ACL’19] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, Ruslan Salakhutdinov, Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context, ACL, 2019

Reference


Reference


• [Liu, et al., ACL’19] Xiaodong Liu, Pengcheng He, Weizhu Chen, Jianfeng Gao, Multi-Task Deep Neural Networks for Natural Language Understanding, ACL, 2019

• [Stickland, et al., ICML’19] Asa Cooper Stickland, Iain Murray, BERT and PALs: Projected Attention Layers for Efficient Adaptation in Multi-Task Learning, ICML, 2019
Reference

• [Howard, et al., ACL’18] Jeremy Howard, Sebastian Ruder, Universal Language Model Fine-tuning for Text Classification, ACL, 2018
Reference

• [Cui, et al., arXiv’19] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, Guoping Hu, Pre-Training with Whole Word Masking for Chinese BERT, arXiv, 2019

• [Sun, et al., ACL’19] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, Hua Wu, ERNIE: Enhanced Representation through Knowledge Integration, ACL, 2019

• [Dong, et al., NeurIPS’19] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, Hsiao-Wuen Hon, Unified Language Model Pre-training for Natural Language Understanding and Generation, NeurIPS, 2019