NLP using Transformer Architectures

TF World 2019

Aurélien Géron
ML Consultant
@aureliengeron
❖ NLP Tasks and Datasets
❖ The Transformer Architecture
❖ Recent Language Models
Natural Language Processing
An artistic triumph
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!
An artistic triumph
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!

POSITIVE ?
Sentiment Analysis

LABEL = POSITIVE

An artistic triumph
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!

POSITIVE ?
Sentiment Analysis

An artistic triumph
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!

- Recommender Systems
- Market Sentiment
Sentiment Analysis

🌟 9/10

**An artistic triumph**
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!
Sentiment Analysis

⭐ 9/10

**An artistic triumph**
21 September 2015

For some reason, I couldn't quite catch this movie in theaters and I managed to watch it on an international flight. And boy, am I glad I did!

```python
!pip install tensorflow-datasets  # or tfds-nightly

import tensorflow_datasets as tfds
datasets = tfds.load("imdb_reviews")
```
TensorFlow Datasets

datasets = tfds.load("imdb_reviews")
train_set = datasets["train"]  # 25,000 reviews
test_set = datasets["test"]   # 25,000 reviews
TensorFlow Datasets

```python
train_set, test_set = tfds.load(
    "imdb_reviews",
    split=["train", "test"]
)
```
TensorFlow Datasets

train_set, test_set = tfds.load("imdb_reviews:1.0.0", split=["train", "test"])
TensorFlow Datasets

train_set, test_set = tfds.load("imdb_reviews:1.0.0",
split=['train', 'test[:60%]'])
TensorFlow Datasets

```python
train_set, test_set, valid_set = tfds.load(
    "imdb_reviews:1.0.0",
    split=['train', 'test[:60%]', 'test[60%:]'])
```
TensorFlow Datasets

```python
train_set, test_set, valid_set = tfds.load(
  "imdb_reviews:1.0.0",
  split=['train', 'test[:60%]', 'test[60%:]'],
  as_supervised=True)
```
TensorFlow Datasets

```python
for review, label in train_set.take(2):
    print(review.numpy().decode("utf-8"))
    print(label.numpy())
```
This was an absolutely terrible movie. Don't be lured in by Christopher Walken or Michael Ironside. Both are great actors, but this must simply be their worst role in history. [...] 0

This is the kind of film for a snowy Sunday afternoon when the rest of the world can go ahead with its own [...] A family film in every sense and one that deserves the praise it received. 1
TensorFlow Datasets

tensorflow.org/datasets/catalog
Translation

"Das Leben ist wunderbar."

"Life is wonderful."
Translation

"Das Leben ist wunderbar."

"Life is wonderful."

Workshop on Machine Translation (WMT)

datasets = tfds.load("wmt19_translate/de-en") # 10 GB!
"Popular YouTubers raise 10 million dollars in 5 days to plant 10 million trees, in an effort to fight global warming."
"Popular YouTubers raise 10 million dollars in 5 days to plant 10 million trees, in an effort to fight global warming."

datasets = tfds.load("multi_news")
What is Southern California often abbreviated as?

SoCal
Question Answering

Context
Southern California, often abbreviated SoCal, is a geographic and cultural region that generally comprises California's southernmost 10 counties. The region is traditionally described as "eight counties", based on demographics and [...]

Question
What is Southern California often abbreviated as?

Answer
SoCal

Stanford Question Answering Dataset (SQuAD)

datasets = tfds.load("squad")
Semantic Equivalence

"Alice lost her keys." ↔ "Alice could not find her keys."
Semantic Equivalence

"Alice lost her keys."  "Alice could not find her keys."

Microsoft Research Paraphrase Corpus

datasets = tfds.load("glue/mrpc")
Entailment

"Alice lost her keys, but Betty found them and returned them to her."

→ "Alice got her keys back."
Entailment

"Alice lost her keys, but Betty found them and returned them to her."

Entails

"Alice got her keys back."
Entailment

"Alice lost her keys, but Betty found them and returned them to her."  Contradicts  "Alice lost her keys forever."
Entailment

"Alice lost her keys, but Betty found them and returned them to her.”

Neutral

"Alice and Betty are sisters."
Entailment

"Alice lost her keys, but Betty found them and returned them to her."  →  "Alice and Betty are sisters."

Neutral

Stanford Natural Language Inference Dataset

datasets = tfds.load("snli/plain_text")
Coreference Resolution

"Alice lost her keys, but Betty found them and returned them to her."
Coreference Resolution

"Alice lost her keys, but Betty found them and returned them to her."
"Alice lost her keys, but Betty found them and returned them to her."

Coreference Resolution
Coreference Resolution

"Alice lost her keys, but Betty found them and returned them to her."

Stanford Natural Language Inference Dataset

```python
datasets = tfds.load("gap")
```
Information Extraction

- Alice was born in London.
- Bob is married to Charlie.
- Eve is friends with both Bob and Charlie.
Speech to Text
Text to Speech
Optical Character Recognition
### Metrics

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>GLUE</th>
<th>CNNDM</th>
<th>SQuAD</th>
<th>SGLUE</th>
<th>EnDe</th>
<th>EnFr</th>
<th>EnRo</th>
</tr>
</thead>
<tbody>
<tr>
<td>★ C4</td>
<td>745GB</td>
<td>83.28</td>
<td>19.24</td>
<td>80.88</td>
<td>71.36</td>
<td>26.98</td>
<td>39.82</td>
<td>27.65</td>
</tr>
<tr>
<td>C4, unfiltered</td>
<td>6.1TB</td>
<td>81.46</td>
<td>19.14</td>
<td>78.78</td>
<td>68.04</td>
<td>26.55</td>
<td>39.34</td>
<td>27.21</td>
</tr>
<tr>
<td>RealNews-like</td>
<td>35GB</td>
<td>83.83</td>
<td>19.23</td>
<td>80.39</td>
<td>72.38</td>
<td>26.75</td>
<td>39.90</td>
<td>27.48</td>
</tr>
<tr>
<td>WebText-like</td>
<td>17GB</td>
<td>84.03</td>
<td>19.31</td>
<td>81.42</td>
<td>71.40</td>
<td>26.80</td>
<td>39.74</td>
<td>27.59</td>
</tr>
<tr>
<td>Wikipedia</td>
<td>16GB</td>
<td>81.85</td>
<td>19.31</td>
<td>81.29</td>
<td>68.01</td>
<td>26.94</td>
<td>39.69</td>
<td>27.67</td>
</tr>
<tr>
<td>Wikipedia + TBC</td>
<td>20GB</td>
<td>83.65</td>
<td>19.28</td>
<td>82.08</td>
<td>73.24</td>
<td>26.77</td>
<td>39.63</td>
<td>27.57</td>
</tr>
</tbody>
</table>

**Table 8:** Performance resulting from pre-training on different datasets. The first four variants are based on our new C4 dataset.

Transformer Architecture
Attention Is All You Need

https://arxiv.org/abs/1706.03762

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez†
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaiser@google.com

Illia Polosukhin*‡
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine-translation tasks show these models to
Attention Is All You Need

Source: Figure 1 from the paper “Attention Is All You Need” by Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser and Polosukhin
https://arxiv.org/abs/1706.03762
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

Abstract

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models (Peters et al., 2018; Radford et al., 2018), BERT is designed to pre-train deep bidirectional representations by jointly conditioning on both left and right context in all layers. As a result, the pre-trained representations can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks.

Language Models are Unsupervised Multitask Learners

Alec Radford 1  Jeffrey Wu 1  Rewon Child 1  David Luan 1  Dario Amodei 1  Ilya Sutskever 1

Abstract

Natural language processing tasks, such as question answering, machine translation, reading comprehension, and summarization, are typically approached with supervised learning on task-specific datasets. We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText. When conditioned on a document plus questions, the answers generated by the language model reach 55 F1 on the CoQA dataset - matching or exceeding the performance of 3 out of 4 baseline systems without using the 127,000+ training examples. The capacity of the language model is essential to the success of zero-shot task transfer and increasing it improves performance in a log-linear fashion across tasks. Our largest model, GPT-2, is a 1.5B parameter Transformer that achieves state of the art results on 7 out of 8 tested language modeling datasets in a zero-shot setting but still underfits WebText. Samples from the model reflect these improvements and contain coherent paragraphs of text. These findings suggest competent generalists. We would like to move towards more general systems which can perform many tasks - eventually without the need to manually create and label a training dataset for each one.

The dominant approach to creating ML systems is to collect a dataset of training examples demonstrating correct behavior for a desired task, train a system to imitate these behaviors, and then test its performance on independent and identically distributed (IID) held-out examples. This has served well to make progress on narrow experts. But the often erratic behavior of captioning models (Lake et al., 2017), reading comprehension systems (Jia & Liang, 2017), and image classifiers (Alcorn et al., 2018) on the diversity and variety of possible inputs highlights some of the shortcomings of this approach.

Our suspicion is that the prevalence of single task training on single domain datasets is a major contributor to the lack of generalization observed in current systems. Progress towards robust systems with current architectures is likely to require training and measuring performance on a wide range of domains and tasks. Recently, several benchmarks have been proposed such as GLUE (Wang et al., 2018) and decaNLP (McCann et al., 2018) to begin studying this.
Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

Zihang Dai\textsuperscript{1,2}, Zhilin Yang\textsuperscript{1}\textsuperscript{*2}, Yiming Yang\textsuperscript{1}, Jaime Carbonell\textsuperscript{1}, Quoc V. Le\textsuperscript{2}, Ruslan Salakhutdinov\textsuperscript{1}  
\textsuperscript{1}Carnegie Mellon University, \textsuperscript{2}Google Brain  
\{dzihang,zhiliny,yiming,jgc,rsalakh}@cs.cmu.edu, qvl@google.com

Abstract

Transformers have a potential of learning longer-term dependency, but are limited by the fixed-length context in the setting of language modeling. We propose a novel neural architecture Transformer-XL that enables learning dependency beyond a fixed length without disrupting temporal coherence. The architecture consists of a segment-level recurrence mechanism, and a novel positional encoding scheme. The method not only enables capturing longer dependency, but also resolves the context representation problem. As a result, Transformer-XL learns dependency that is 80\% longer than RNNs and 450\% longer than vanilla Transformers, achieves better performance on a variety of downstream tasks.

---

XLNet: Generalized Autoregressive Pretraining for Language Understanding

Zhilin Yang\textsuperscript{1}, Zihang Dai\textsuperscript{1,2}, Yiming Yang\textsuperscript{1}, Jaime Carbonell\textsuperscript{1}, Ruslan Salakhutdinov\textsuperscript{1}, Quoc V. Le\textsuperscript{2}  
\textsuperscript{1}Carnegie Mellon University, \textsuperscript{2}Google Brain  
\{zhiliny,dzihang,yiming,jgc,rsalakh}@cs.cmu.edu, qvl@google.com

Abstract

With the capability of modeling bidirectional contexts, denoising autoencoding based pretraining like BERT achieves better performance than pretraining approaches based on autoregressive language modeling. However, relying on corrupting the input with masks, BERT neglects dependency between the masked positions and suffers from a pretrain-finetune discrepancy. In light of these pros and cons, we propose XLNet, a generalized autoregressive pretraining method that (1) enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and (2) overcomes the limitations of BERT thanks to its autoregressive formulation. Furthermore, XLNet integrates ideas from Transformer-XL, the state-of-the-art autoregressive model, into pretraining. Empirically, XLNet outperforms BERT on 20 tasks, often by a large margin, and achieves state-of-the-art results on 18 tasks including question answering, natural language inference, and summarization.
Abstract

Language model pretraining has led to significant performance gains but care in the comparison between different approaches is challenging. Training is computationally intensive, often done on private datasets of variable sizes, and, as we will show, hyperparameter choices have significant impact on the results. We present a replication study of pretraining (Devlin et al., 2019) that measures the impact of many key hyperparameters and training data size. We find that BERT was significantly undertrained, and can easily exceed the performance of even larger models published after it. Our best model outperforms state-of-the-art results on GLUE, RACE, and SQuAD. These results highlight the importance of previously overlooked design choices.


table

1 Introduction

Training a language model to transform natural language processing (NLP) tasks into text.
LSTM Encoder/Decoder

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

Embedding lookup

“<go> Je bois du lait”

“I drink milk”
BiLSTM Encoder/Decoder

Target: Je
Prediction: Je bois du lait <eos>

Encoder – Decoder

Softmax

“<go> Je bois du lait ”

“I drink milk”
# Neural Machine Translation by Jointly Learning to Align and Translate

**Dzmitry Bahdanau**  
Jacobs University Bremen, Germany

**KyungHyun Cho**  **Yoshua Bengio**  
Université de Montréal

## Abstract

Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder-decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder-decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.
Encoder/Decoder with Attention

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

"I drink milk"
Encoder/Decoder with Attention

Encoder – Decoder

Target: Je  bois  du  lait  <eos>
Prediction: Je  bois  le  lait  <eos>

Embedding lookup

288  3335  72

“I drink milk”

“<go> Je bois du lait”
Encoder/Decoder with Attention

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

"I drink milk"

"<go> Je bois du lait "

Embedding lookup

Encoder – Decoder

Softmax
Encoder/Decoder with Attention

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

"I drink milk"

"<go> Je bois du lait "
Attention Mechanism

I

drink

milk
Attention Mechanism

Verb

Food-related

Non-verb

Not food-related

drink

milk
Encoder/Decoder with Attention

Target: Je  bois  du  lait  <eos>
Prediction: Je  bois  le  lait  <eos>

Embedding lookup
288  3335  72

“I drink milk”

“<go> Je bois du lait”
Attention Mechanism

```
“\text{I drink}"
```

Verb

Not food-related

Food-related

Non-verb

milk
Encoder/Decoder with Attention

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

<table>
<thead>
<tr>
<th>Prediction</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Je</td>
<td>Je</td>
</tr>
<tr>
<td>bois</td>
<td>bois</td>
</tr>
<tr>
<td>du</td>
<td>le</td>
</tr>
<tr>
<td>lait</td>
<td>lait</td>
</tr>
<tr>
<td>&lt;eos&gt;</td>
<td>&lt;eos&gt;</td>
</tr>
</tbody>
</table>

Encoder – Decoder

Embedding lookup

“I drink milk”

“<go> Je bois du lait”
Attention Mechanism

Verb

Not food-related

Food-related

Non-verb

drink

“I

“Je bois le <”

Query

milk
Attention Mechanism

The diagram illustrates the attention mechanism for the word "drink" in the context of a query "I drink milk." The axes represent "Non-verb" and "Food-related" dimensions. The vector from the query point to the "drink" point indicates the attention weight, with a value of 0.9. The other food-related term, "milk," is also shown with a lower attention weight of 0.05 for non-verb and 0.05 for food-related dimensions.
Attention Mechanism

I

Verb

Not food-related

Food-related

Non-verb

Input to next step

drink

milk
Encoder/Decoder with Attention

Target: Je bois du lait <eos>
Prediction: Je bois le lait <eos>

encoder/decoder with attention

Embedding lookup

“<go> Je bois du lait”

“I drink milk”
Attention Is All You Need

Target: *Je*  
Prediction: *Je* *bois* *du* *lait* *<eos>*

Encoder – Decoder

Softmax

Embedding lookup

“I drink milk”

“<go> Je bois du lait”
Attention Is All You Need

bat
Attention Is All You Need

bat

?
Attention Is All You Need

The **bat** was sleeping
Attention Is All You Need

The bat was sleeping

Attention layer

The bat was sleeping
Attention Is All You Need

The bat was sleeping

Attention layer

The bat was sleeping
Figure 1 from the paper
Figure 1 from the paper (simplified)
Figure 1 from the paper (simplified)
Figure 1 from the paper (simplified)
Figure 1 from the paper
(simplified)
Attention Is All You Need

The bat was sleeping
Attention Is All You Need

The bat was sleeping
The bat was sleeping

The bat was sleeping
Attention Is All You Need

The bat was sleeping

The bat was sleeping

The bat was sleeping
Attention Is All You Need

The bat was sleeping

The bat was sleeping

The bat was sleeping

The bat was sleeping
The bat was sleeping

The bat was sleeping

The bat was sleeping

The bat was sleeping

<sos> La chauve souris dort
Attention Is All You Need

The bat was sleeping

La chauve souris dort

<sos> La chauve souris dort
Attention Is All You Need

The bat was sleeping

<sos> La chauve souris dort
Attention Is All You Need

The bat was sleeping
The bat was sleeping
The bat was sleeping
The bat was sleeping

La chauve souris dort <eos>
<sos> La chauve souris dort
<sos> La chauve souris dort
<sos> La chauve souris dort
The bat was sleeping
Attention Is All You Need

The bat was sleeping

The bat was sleeping

The bat was sleeping

The bat was sleeping

La

<sos>

<sos>

<sos>

<sos> La
Attention Is All You Need

The bat was sleeping

The bat was sleeping

The bat was sleeping

The bat was sleeping

La chauve

<sos> La

<sos> La

<sos> La

<sos> La
Attention Is All You Need

The bat was sleeping
The bat was sleeping
The bat was sleeping
The bat was sleeping

La chauve -
<sos> La
<sos> La
<sos> La
<sos> La chauve
The bat was sleeping
The bat was sleeping
The bat was sleeping
The bat was sleeping

La chauve souris dort <eos>
<sos> La chauve souris dort
<sos> La chauve souris dort
<sos> La chauve souris dort
Multi-Head Attention

Verb

Food-related

Non-verb

Not food-related

Query

drink

milk
Multi-Head Attention

Verb

Food-related

Not food-related

Non-verb

Query

milk

drink

I
Multi-Head Attention

Verb

drink

I

Non-verb

milk

Query

Food-related

Not food-related
Multi-Head Attention

Verb

Not food-related  

Food-related

I

Non-verb

milk

Query

drink
Multi-Head Attention

Verb

Non-verb

Not food-related

Food-related

Query

I

drink

milk
Multi-Head Attention

Verb

drink

Non-verb

milk

Not food-related

Food-related

Query
Language Models
Pretraining + Fine-tuning

Universal Language Model Fine-tuning for Text Classification

Jeremy Howard∗
fast.ai
University of San Francisco
j@fast.ai

Abstract
Inductive transfer learning has greatly impacted computer vision, but existing approaches in NLP still require task-specific modifications and training from scratch. We propose Universal Language Model Fine-tuning (ULMFiT), an effective transfer learning method that can be applied to any task in NLP, and introduce techniques that are key for fine-tuning a language model. Our method significantly outperforms the state-of-the-art on six text classification tasks.

ULMFiT
Jan 2018

Deep contextualized word representations

Matthew E. Peters‡, Mark Neumann†, Mohit Iyyer†, Matt Gardner†,
{matthewp,markn,mohiti,mattg}@allenai.org

Christopher Clark∗, Kenton Lee∗, Luke Zettlemoyer†∗
{csquared,kentonl,lsz}@cs.washington.edu

†Allen Institute for Artificial Intelligence
*Paul G. Allen School of Computer Science & Engineering, University of Washington

Abstract
We introduce a new type of deep contextualized word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). Our word vectors are learned functions of the internal states of a deep bidirectional language model (biLM), which is pre-trained on a large text corpus. We show that these representations can be easily added to existing models and significantly improve the language model (LM) objective on a large text corpus. For this reason, we call them ELMo (Embeddings from Language Models) representations. Unlike previous approaches for learning contextualized word vectors (Peters et al., 2017; McCann et al., 2017), ELMo representations are deep, in the sense that they are a function of all of the internal layers of the biLM. More specifically, we learn a linear combination of the vectors stacked above each input word for each end task, which markedly improves performance over just using the top LSTM layer.

ELMo
Feb 2018
BERT

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

Abstract

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models (Peters et al., 2018; Radford et al., 2018), BERT is designed to pre-train deep bidirectional representations by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT representations can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing models are required to produce fine-grained output at the token-level.

There are two existing strategies for applying pre-trained language representations to downstream tasks: feature-based and fine-tuning. The feature-based approach, such as ELMo (Peters et al., 2018), uses tasks-specific architectures that include the pre-trained representations as additional features. The fine-tuning approach, such as the Generative Pre-trained Transformer (OpenAI GPT) (Radford et al., 2018), introduces minimal task-specific parameters, and is trained on the downstream tasks by simply fine-tuning the pre-trained parameters. In previous work, both approaches share the same objective function during pre-training, where they use unidirectional language models to learn general language representations.

- Large number of parameters
- Pretraining on a huge corpus
- Subword tokenization
- Single architecture for multiple tasks
Subword Tokenization

Going on computerless vacation

Going on a computer less vacation
BERT — Figure 4 From Paper

(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA
(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER
Language Models are Unsupervised Multitask Learners

Alec Radford 1, Jeffrey Wu 1, Rewon Child 1, David Luan 1, Dario Amodei 1, 1 Ilya Sutskever 1

Abstract

Natural language processing tasks, such as question answering, machine translation, reading comprehension, and summarization, are typically approached with supervised learning on task-specific datasets. We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText. When conditioned on a document plus questions, the answers generated by the language model reach 55 F1 on the CoQA dataset - matching or exceeding the performance of 3 out of 4 baseline systems without using the 127,000+ training examples. The capacity of the language model is essential to the success of zero-shot task transfer and increasing it improves performance in a log-linear fashion across tasks. Our largest model, GPT-2, is a 1.5B parameter Transformer that achieves state of the art results on 7 out of 8 tested language modeling datasets in a zero-shot setting but still underfits WebText. Samples from the model reflect these improvements and contain coherent paragraphs of text. These findings suggest competent generalists. We would like to move towards more general systems which can perform many tasks - eventually without the need to manually create and label a training dataset for each one.

The dominant approach to creating ML systems is to collect a dataset of training examples demonstrating correct behavior for a desired task, train a system to imitate these behaviors, and then test its performance on independent and identically distributed (IID) held-out examples. This has served well to make progress on narrow experts. But the often erratic behavior of captioning models (Lake et al., 2017), reading comprehension systems (Jia & Liang, 2017), and image classifiers (Alcorn et al., 2018) on the diversity and variety of possible inputs highlights some of the shortcomings of this approach.

Our suspicion is that the prevalence of single task training on single domain datasets is a major contributor to the lack of generalization observed in current systems. Progress towards robust systems with current architectures is likely to require training and measuring performance on a wide range of domains and tasks. Recently, several benchmarks have been proposed such as GLUE (Wang et al., 2018) and decaNL (McCann et al., 2018) to begin studying this.
Transformer-XL: Attentive Language Models
Beyond a Fixed-Length Context

Zihang Dai¹², Zhilin Yang¹², Yiming Yang¹, Jaime Carbonell¹,
Quoc V. Le², Ruslan Salakhutdinov¹
¹Carnegie Mellon University, ²Google Brain
{dzihang,zhiliny,yiming,jgc,rsalakhu}@cs.cmu.edu, qvl@google.com

Abstract
Transformers have a potential of learning longer-term dependency, but are limited by a fixed-length context in the setting of language modeling. We propose a novel neural architecture Transformer-XL that enables learning dependency beyond a fixed length without disrupting temporal coherence. It consists of a segment-level recurrence mechanism and a novel positional encoding scheme. Our method not only enables capturing longer-term dependency, but also resolves the context fragmentation problem. As a result, Transformer-XL learns dependency that is 80% longer than RNNs and 450% longer than vanilla Transformers, achieves better performance on both Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997), have been a standard solution to language modeling and obtained strong results on multiple benchmarks. Despite the wide adaption, RNNs are difficult to optimize due to gradient vanishing and explosion (Hochreiter et al., 2001), and the introduction of gating in LSTMs and the gradient clipping technique (Graves, 2013) might not be sufficient to fully address this issue. Empirically, previous work has found that LSTM language models use 200 context words on average (Khandelwal et al., 2018), indicating room for further improvement.

On the other hand, the direct connections between long-distance word pairs baked in atten-
XLNet: Generalized Autoregressive Pretraining for Language Understanding

Zhilin Yang\textsuperscript{1}, Zihang Dai\textsuperscript{12}, Yiming Yang\textsuperscript{1}, Jaime Carbonell\textsuperscript{1}, Ruslan Salakhutdinov\textsuperscript{1}, Quoc V. Le\textsuperscript{2}
\textsuperscript{1}Carnegie Mellon University, \textsuperscript{2}Google Brain
\{zhiliny, dzihang, yiming, jgc, rsalakhu\}@cs.cmu.edu, qvl@google.com

Abstract

With the capability of modeling bidirectional contexts, denoising autoencoding based pretraining like BERT achieves better performance than pretraining approaches based on autoregressive language modeling. However, relying on corrupting the input with masks, BERT neglects dependency between the masked positions and suffers from a pretrain-finetune discrepancy. In light of these pros and cons, we propose XLNet, a generalized autoregressive pretraining method that (1) enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and (2) overcomes the limitations of BERT thanks to its autoregressive formulation. Furthermore, XLNet integrates ideas from Transformer-XL, the state-of-the-art autoregressive model, into pretraining. Empirically, XLNet outperforms BERT on 20 tasks, often by a large margin, and achieves state-of-the-art results on 18 tasks including question answering, natural
RoBERTa: A Robustly Optimized BERT Pretraining Approach

Yinhan Liu† Myle Ott† Naman Goyal† Jingfei Du† Mandar Joshi†
Danqi Chen§ Omer Levy§ Mike Lewis§ Luke Zettlemoyer§ Veselin Stoyanov§

† Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA
{mandar90,lsz}@cs.washington.edu
§ Facebook AI
{yinhanliu,myleott,naman,jingfeidu,
danqi,omerlevy,mikelewis,lsz,ves}@fb.com

Abstract

Language model pretraining has led to significant performance gains but careful comparison between different approaches is challenging. Training is computationally expensive, often done on private datasets of different sizes, and, as we will show, hyperparameter choices have significant impact on the final results. We present a replication study of BERT pretraining (Devlin et al., 2019) that carefully measures the impact of many key hyperparameters and training data size. We find that BERT was significantly undertrained, and can match or exceed the performance of every model published after it. Our best model achieves state-of-the-art results on GLUE, RACE and SQuAD. These results highlight the importance of previously overlooked design choices.

We present a replication study of BERT pretraining (Devlin et al., 2019), which includes a careful evaluation of the effects of hyperparameter tuning and training set size. We find that BERT was significantly undertrained and propose an improved recipe for training BERT models, which we call RoBERTa, that can match or exceed the performance of all of the post-BERT methods. Our modifications are simple, they include: (1) training the model longer, with bigger batches, over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data. We also collect a large new dataset (CC-NEWS) of comparable size to other privately used datasets, to better control for training set size effects.
Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer

Colin Raffel*  Noam Shazeer*  Adam Roberts*  Katherine Lee*
Sharan Narang  Michael Matena  Yanqi Zhou  Wei Li  Peter J. Liu

Google

Abstract
Transfer learning, where a model is first pre-trained on a data-rich task before being fine-tuned on a downstream task, has emerged as a powerful technique in natural language processing (NLP). The effectiveness of transfer learning has given rise to a diversity of approaches, methodology, and practice. In this paper, we explore the landscape of transfer learning techniques for NLP by introducing a unified framework that converts every language problem into a text-to-text format. Our systematic study compares pre-training objectives, architectures, unlabeled datasets, transfer approaches, and other factors on dozens of language understanding tasks. By combining the insights from our exploration with scale and our new “Colossal Clean Crawled Corpus”, we achieve state-of-the-art results on many benchmarks covering summarization, question answering, text classification, and more. To facilitate future work on transfer learning for NLP, we release our dataset, pre-trained models, and code.

1 Introduction

Training a hierarchical language model on a large corpus of text (e.g., Wikipedia) can improve the performance on a variety of downstream tasks.
"translate English to German: That is good."

"cola sentence: The course is jumping well."

"sts1: The rhino grazed on the grass. sts2: A rhino is grazing in a field."

"summarize: state authorities dispatched emergency crews Tuesday to survey the damage after an onslaught of severe weather in Mississippi."

"Das ist gut."

"not acceptable"

"3.8"

"six people hospitalized after a storm in Attala County."
Get Coding!

https://github.com/huggingface/transformers
https://medium.com/@lysandrejik
Thank you!