ROBOT INTELLIGENCE LAB

Self-Supervised Learning

Language-domain SSL

Sungjoon Choi, Korea University

kakaobrain

Content

- GPT
- GPT-2
- BERT
- RoBERTa
- ALBERT
- GPT-3

ROBOT INTELLIGENCE LAB

GPT

"Improving Language Understanding by Generative Pre-Training," 2018

Generative Pre-Training (GPT)

The goal is to learn a **universal representation** that transfers with little adaptation to a wide range of tasks.

GPT

ROBOT INTELLIGENCE LAB

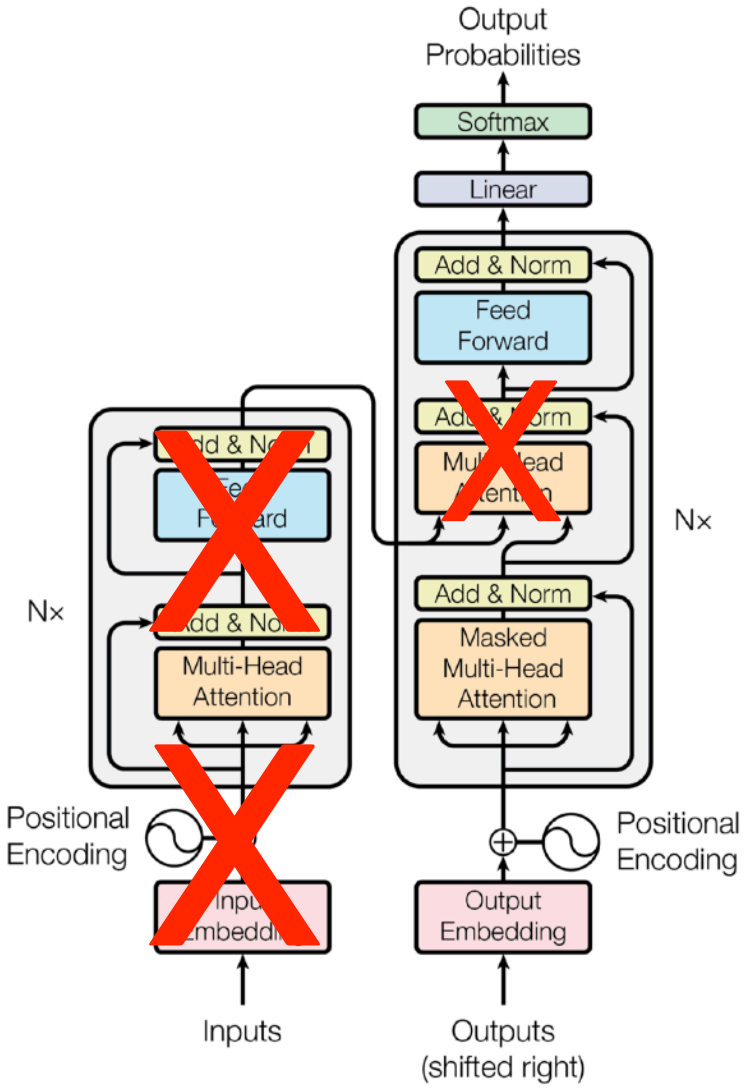
- GPT consists of two stages: the first stage is learning a high-capacity language model on a large corpus of text and followed by a fine-tuning stage.
- Unsupervised pre-training
  - Given an unsupervised corpus of tokens $\mathcal{U} = \{u_1, \ldots, u_n\}$, a standard language modeling objective is used to maximize the following likelihood:

$$L_1(\mathcal{U}) = \sum_i \log P(u_i \mid u_{i-k}, \ldots, u_{i=1}; \Theta)$$

    where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ
- A multi-layer **Transformer decoder** is used.

Why Transformer Decoder?

ROBOT INTELLIGENCE LAB



Traversal-Style Approach

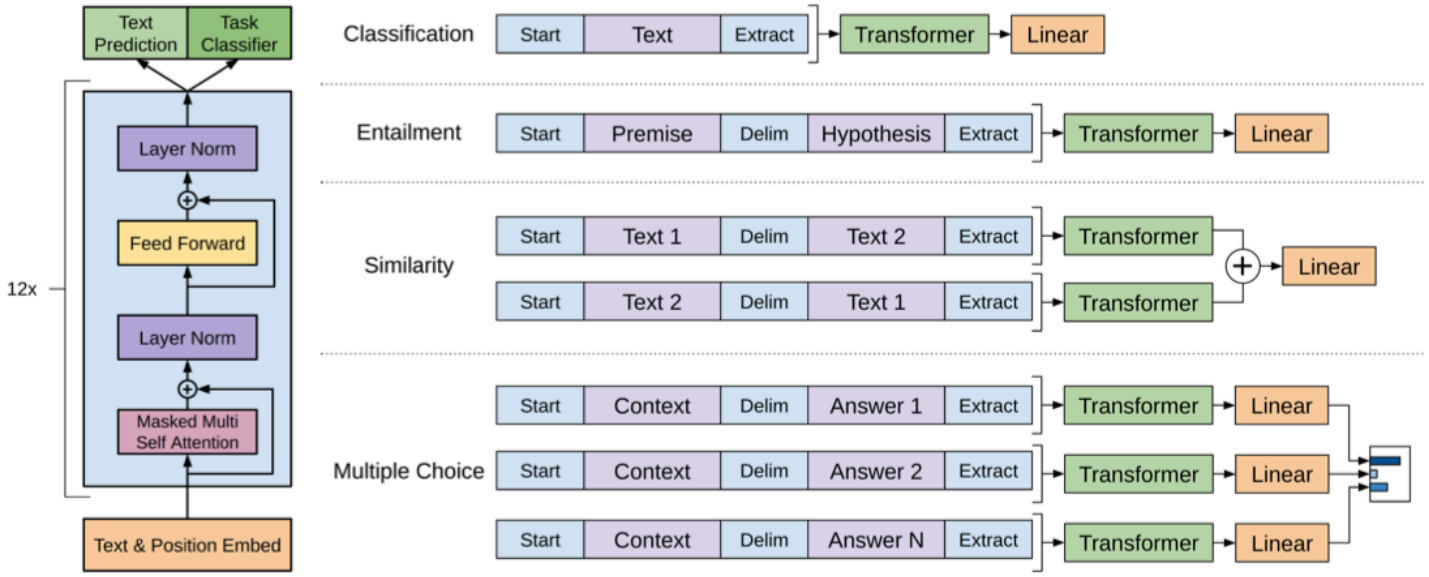


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

I dislike old cabin cruisers. (negative)

Peter is snoring. $ A man sleeps. (true)

President greets the press in Chicago.
Obama speaks in Illinois.
(similar)

Dataset

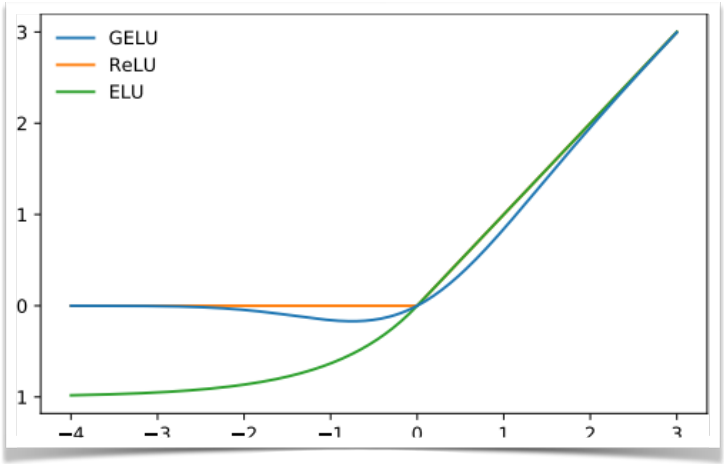
ROBOT INTELLIGENCE LAB

- **BooksCorpus** dataset is used for pre-training
  - It contains over 7,000 unique unpublished books from a variety of genres including Adventure, Fantasy, and Romance.
  - In contains long stretches of contiguous text, which allows the generative model to learn to condition on long-range information.
- **1D Word Benchmark** is an alternative option
  - It has approximately the same size as BooksCorpus but is shuffled at a sentence level, destroying long-range structure.
  - GPT achieves a very low performance on this dataset.

Implementation Details

ROBOT INTELLIGENCE LAB

- 12-layer-decoder-only transformer with masked self-attention heads
  - 768 dimensional states and 12 attention heads
  - For the position-wise-feed-forward networks, 3072 dimensional inner states are used.
  - Adam optimizer with a maximum learning rate of $2.5 * 10^{-4}$ with a cosine schedule
  - Gaussian Error Linear Unit (GELU) activations
  - Learned position embeddings
- Fine-tuning details
  - Dropout to the classifier with a rate of 0.1
  - Learning rate of $6.25 * 10^{-5}$ with a linear decay and a batchsize of 32
  - 3 epochs of training was sufficient for most cases



Experiments

ROBOT INTELLIGENCE LAB

Table 2: Experimental results on natural language inference tasks, comparing our model with current state-of-the-art methods. 5x indicates an ensemble of 5 models. All datasets use accuracy as the evaluation metric.

Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	89.3	-	-	-
CAFE [58] (5x)	80.2	79.0	89.3	-	-	-
Stochastic Answer Network [35] (3x)	80.6	80.1	-	-	-	-
CAFE [58]	78.7	77.9	88.5	83.3		
GenSen [64]	71.4	71.3	-	-	82.3	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	**61.7**
Finetuned Transformer LM (ours)	**82.1**	**81.4**	**89.9**	**88.3**	**88.1**	56.0

Natural language inference (NLI): reading a pair of sentences and judging the relationship between from one of entailment, contradiction, or neutral

Experiments

ROBOT INTELLIGENCE LAB

Table 3: Results on question answering and commonsense reasoning, comparing our model with current state-of-the-art methods.. 9x means an ensemble of 9 models.

Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	77.6	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	60.2	50.3	53.3
Finetuned Transformer LM (ours)	**86.5**	**62.9**	**57.4**	**59.0**

Question answering and **commonsense reasoning**: English passages with associated questions from middle and high school exams / selecting the correct ending to multi-sentence stories from two options.

Experiments

ROBOT INTELLIGENCE LAB

Table 4: Semantic similarity and classification results, comparing our model with current state-of-the-art methods. All task evaluations in this table were done using the GLUE benchmark. (*mc*= Mathews correlation, *acc*=Accuracy, *pc*=Pearson correlation)

Method	**Classification**		**Semantic Similarity**			**GLUE**
	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	
Sparse byte mLSTM [16]	-	**93.2**	-	-	-	-
TF-KLD [23]	-	-	**86.0**	-	-	-
ECNU (mixed ensemble) [60]	-	-	-	81.0	-	-
Single-task BiLSTM + ELMo + Attn [64]	35.0	90.2	80.2	55.5	66.1	64.8
Multi-task BiLSTM + ELMo + Attn [64]	18.9	91.6	83.5	72.8	63.3	68.9
Finetuned Transformer LM (ours)	**45.4**	91.3	82.3	**82.0**	**70.3**	**72.8**

Semantic Similarity: Predicting whether two sentences are semantically equivalent or not.

GPT-2

"Language Models are Unsupervised Multitask Learners," 2018

GPT-2

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**.

Language-based Task Specification

- Language Prompt
  - Language provides a flexible way to specify **tasks**, **inputs**, and **outputs** all as a sequence of symbols.
  - Translation tasks
    - (translate to french, english text, french text)
  - Reading comprehension tasks
    - (answer the question, document, question, answer)

WebText

ROBOT INTELLIGENCE LAB

- OpenAI created a new web scrape which emphasizes document quality.
  - Web pages which have been curated/filtered by humans are only used.
  - First, all outbound links from Reddit which received at least 3 karma are scraped.
  - The resulting **WebText** contains the text subset of these 45 million links.
    - It contains slightly over 8 million documents for a total of 40 GB of text.
    - All Wikipedia documents are removed.

WebText

ROBOT INTELLIGENCE LAB

”I’m not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I’m not a fool].**
In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: ”**Mentez mentez, il en restera toujours quelque chose**,” which translates as, ”**Lie lie and something will always remain.**”
“I hate the word ‘**perfume**,”’ Burr says. ‘It’s somewhat better in French: ‘**parfum**.’
If listened carefully at 29:55, a conversation can be heard between two guys in French: “**-Comment on fait pour aller de l’autre coté? -Quel autre coté?**”, which means “**- How do you get to the other side? - What side?**”.
If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?
“**Brevet Sans Garantie Du Gouvernement**”, translated to English: “**Patented without government warranty**”.

Table 1. Examples of naturally occurring demonstrations of English to French and French to English translation found throughout the WebText training set.

Byte Pair Encoding

ROBOT INTELLIGENCE LAB

- Word-level encoding
  - Construct a fixed-size dictionary based on word frequencies
  - Out-of-vocabulary (OOV) problem occurs
- Character-level encoding
  - Construct a dictionary with characters (e.g., lower-case Alphabet: 26, Korean: 11,172)
- Byte Pair Encoding (BPE)
  - BPE is a practical middle ground between character and word level language modeling
  - It is based on subword segmentation

```
#dictionary (word: frequency)
low : 5
lower : 2
newest : 6
widest : 3
# vocabulary
l, o, w, e, r, n, w, s, t, i, d
```

→

```
#dictionary (word: frequency)
l o w : 5
l o w e r : 2
n e w es t : 6
w i d es t : 3
# vocabulary
l, o, w, e, r, n, w, s, t, i, d, es
```

→

```
#dictionary (word: frequency)
l o w : 5
l o w e r : 2
n e w est : 6
w i d est : 3
# vocabulary
l, o, w, e, r, n, w, s, t, i, d, es, est
```

https://wikidocs.net/22592

Implementation Details

ROBOT INTELLIGENCE LAB

- Transformer-based architecture following the OpenAI GPT model
  - Layer normalization is moved to the input of each sub-block and an additional layer normalization was added after the final self-attention block
  - The vocabulary is expanded to 50,257. The context size is increased from 512 to 1,024 tokens and a batchsize is 512.

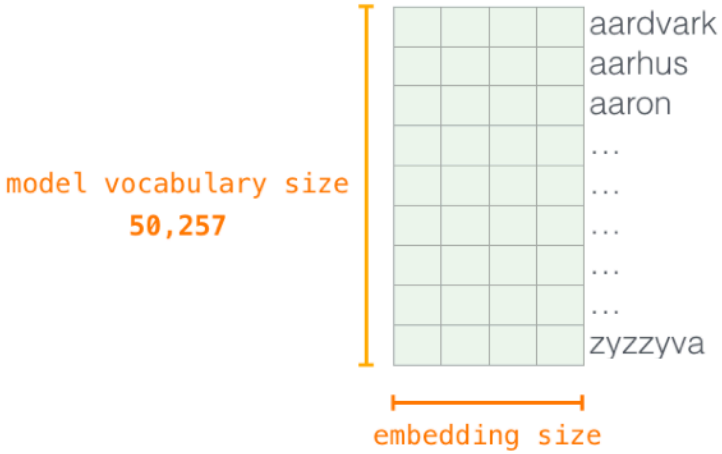
Parameters	Layers	d_{model}	
117M	12	768	← similar to the **original GPT**
345M	24	1024	← similar to the **BERT**
762M	36	1280	
1542M	48	1600	← this one is called **GPT-2**

Table 2. Architecture hyperparameters for the 4 model sizes.

Implementation Details

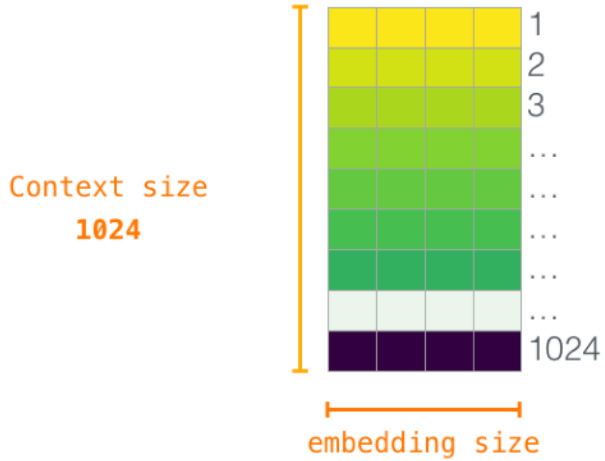
ROBOT INTELLIGENCE LAB

Token Embeddings (wte)



768 (small) / **1024** (medium) / **1280** (large) / **1600** (extra large)

Positional Encodings (wpe)



768 (small) / **1024** (medium) / **1280** (large) / **1600** (extra large)

https://yngie-c.github.io/nlp/2020/07/05/nlp_gpt/

Experiments

ROBOT INTELLIGENCE LAB

Language Models are Unsupervised Multitask Learners

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

Experiments

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%
Largest state in the us by land mass?	California	✗	59.2%
Green algae is an example of which type of reproduction?	parthenogenesis	✗	56.5%
Vikram samvat calender is official in which country?	India	✓	55.6%
Who is mostly responsible for writing the declaration of independence?	Thomas Jefferson	✓	53.3%
What us state forms the western boundary of montana?	Montana	✗	52.3%
Who plays ser davos in game of thrones?	Peter Dinklage	✗	52.1%
Who appoints the chair of the federal reserve system?	Janet Yellen	✗	51.5%
State the process that divides one nucleus into two genetically identical nuclei?	mitosis	✓	50.7%
Who won the most mvp awards in the nba?	Michael Jordan	✗	50.2%
What river is associated with the city of rome?	the Tiber	✓	48.6%
Who is the first president to be impeached?	Andrew Johnson	✓	48.3%
Who is the head of the department of homeland security 2017?	John Kelly	✓	47.0%
What is the name given to the common currency to the european union?	Euro	✓	46.8%
What was the emperor name in star wars?	Palpatine	✓	46.5%
Do you have to have a gun permit to shoot at a range?	No	✓	46.4%
Who proposed evolution in 1859 as the basis of biological development?	Charles Darwin	✓	45.7%
Nuclear power plant that blew up in russia?	Chernobyl	✓	45.7%
Who played john connor in the original terminator?	Arnold Schwarzenegger	✗	45.2%

Table 5. The 30 most confident answers generated by GPT-2 on the development set of Natural Questions sorted by their probability according to GPT-2. None of these questions appear in WebText according to the procedure described in Section 4.

GPT-2 answers 4.1% of questions correctly, which is much worse than 30-50% SOTA results.

ROBOT INTELLIGENCE LAB

BERT

"BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2019

BERT

ROBOT INTELLIGENCE LAB

Bidirectional **E**ncoder **R**epresentation from **T**ransformers (**BERT**)

BERT

ROBOT INTELLIGENCE LAB

- **OpenAI GPT** uses left-to-right architecture, where every token can only attend to **previous** tokens in previous tokens in the self-attention layers of the Transformer.
- **BERT** alleviates the previously mentioned unidirectionality constraint by using a "**masked language model**" (MLN) pre-training objective.
  - The **masked language model** randomly masks some of the tokens from the input, and the objective is to predict the original vocabulary id of the masked word based only on its context.
  - In addition to the masked language model, a "**next sentence prediction**" task is also used that jointly pre-trains text-pair representations.

BERT

ROBOT INTELLIGENCE LAB

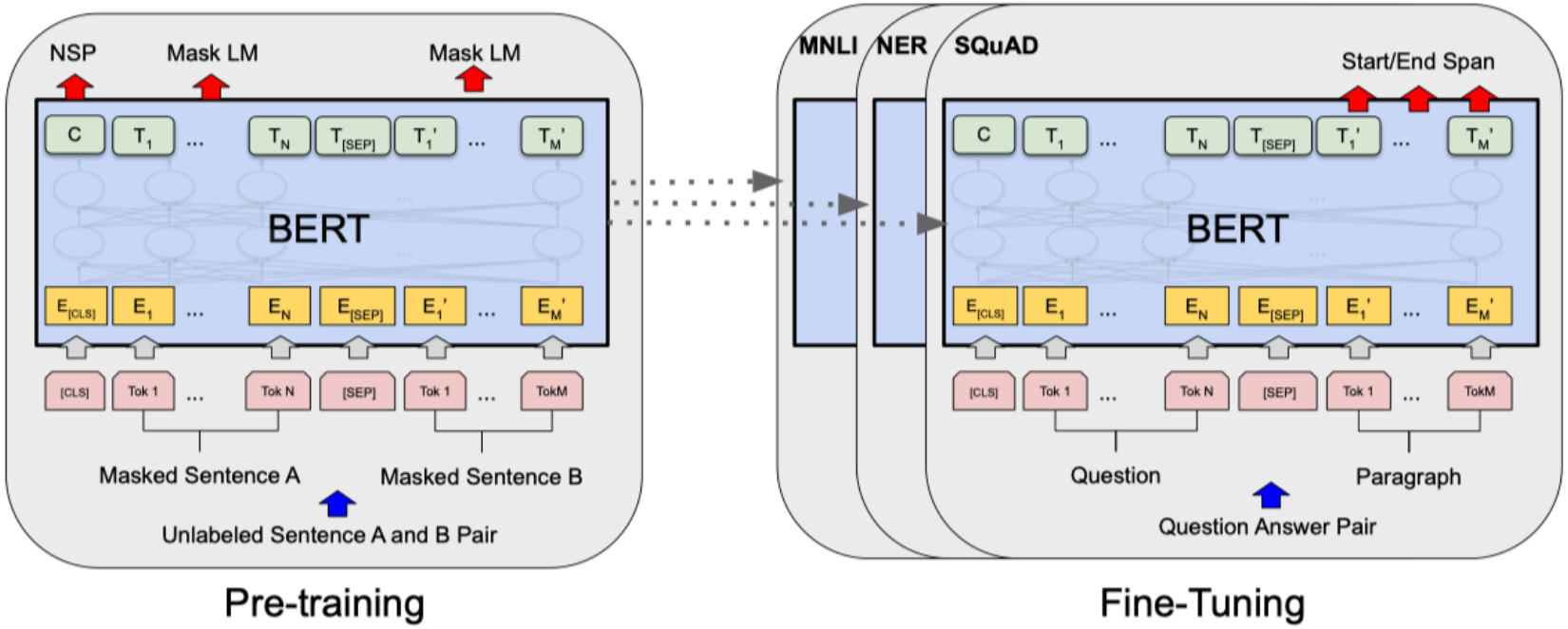


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. `[CLS]` is a special symbol added in front of every input example, and `[SEP]` is a special separator token (e.g. separating questions/answers).

Implementation Details

- BERT's model architecture is a multi-layer bidirectional **Transformer encoder**.
- Suppose that L is the number of layers, H is the hidden size, and A is the number of self-attention heads:
  - **BERT$_{BASE}$**: $(L = 12, H = 786, A = 12)$ #param: 110M
  - **BERT$_{LARGE}$**: $(L = 24, H = 1024, A = 16)$ #param: 340M
  - **GPT-1**: 125M, **GPT-2**: 1.5B, **GPT-3**: 175B
- Input/Output Representation
  - Both a single sentence and a pair of sentences (e.g., question and answer pairs) can be used as an input to BERT.
    - Sentence pairs are packed and they are separated with a special token, [SEP]. A learned embedding to every token is added to indicate whether it belongs to sentence A or sentence B.
  - WordPiece embeddings with a 30,000 token vocabulary are used.
  - The first token is always a special classification token, [CLS]. The final hidden state corresponding to this token is used as the aggregate sequence representation for classification tasks.

Input Representation

ROBOT INTELLIGENCE LAB

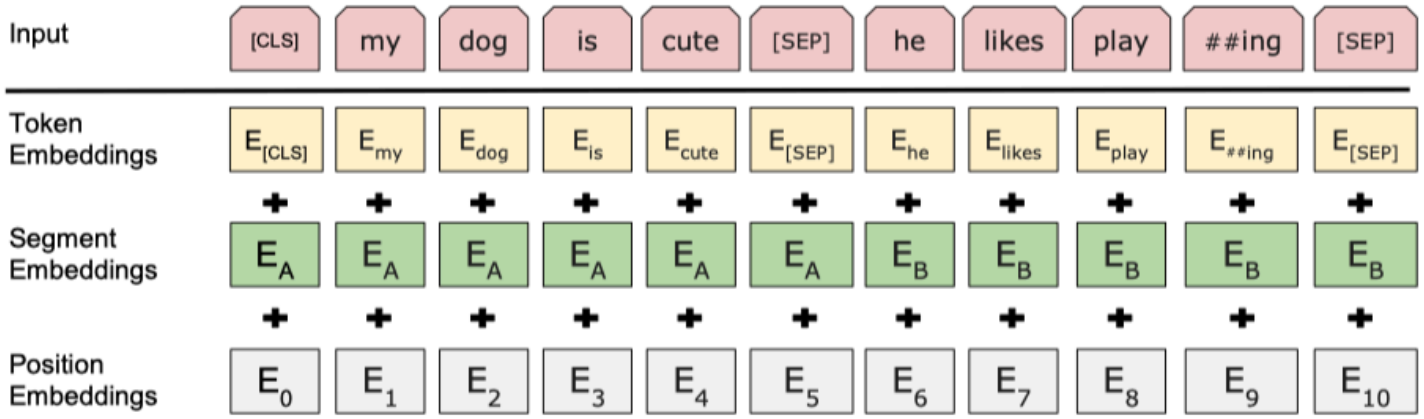


Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Pre-training BERT

- BERT is pre-trained with two unsupervised tasks: **masked language model** and **next sentence prediction**.
- Task 1: Masked LM
  - The masked words (15%) are only predicted rather than the entire input.
  - One downside is that it creates a mismatch between pre-training and fine-tuning phases, since the [MASK] token does not appear during fine-tuning.
  - To mitigate this issue, if the i-th token is chosen, we replace the i-th token with (1) the [MASK] token 80% of the time (2) a random token 10% of the time (3) the unchanged i-th token 10% of the time.
- Task 2: Next Sentence Prediction (NSP)
  - A binarized next sentence prediction tasks is used.
  - When choosing two sentences A and B, 50% of the time B is the actual next sentence that follows A and 50% of the time it is a random sentence from the corpus.

Experiments

ROBOT INTELLIGENCE LAB

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Table 1: GLUE Test results, scored by the evaluation server (https://gluebenchmark.com/leaderboard). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

GLUE

- General Language Understanding Evaluation (**GLUE**)
  - **CoLA**: The Corpus of Linguistic Acceptability: Binary classification: single sentences that are either grammatical or ungrammatical
  - **SST-2**: Stanford Sentiment Treebank:phrases culled from movie reviews scored on their positivity/ negativity. Phrases can be positive, negative, or completely neutral
  - **STS-B**: The Semantic Textual Similarity Benchmark: task of determining the similarity on a continuous scale from 1 to 5 of a pair of sentences drawn from various sources
  - **QQP**: The Quora Question Pairs dataset: collection of question pairs from the community question-answering website Quora: Given two questions, the task is to determine whether they are semantically equivalent

  - https://docs.google.com/spreadsheets/d/1BrOdjJgky7FfeiwC_VDURZuRPUFUAz_jfczPPT35P00/edit#gid=0

RoBERTa

"RoBERTa: A Robustly Optimized BERT Pretraining Approach," 2019

RoBERTa

ROBOT INTELLIGENCE LAB

Robustly optimized BERT approach (RoBERTa)

RoBERTa

ROBOT INTELLIGENCE LAB

- This paper presents an improved recipe for training BERT models, named RoBERTa.
  - training the model longer, with bigger batches, over more data
  - removing the next sentence prediction objective
  - training on longer sequences
  - dynamically changing the making pattern applied to the training data
- A large new dataset (CC-News) is collected.

Dataset

ROBOT INTELLIGENCE LAB

- BERT-style pre-training crucially relies on large quantities of text. Five English-language corpora of varying sizes and domains, totaling over 160GB of uncompressed text, are used.
  - **BookCorpus** plus **English Wikipedia**: the original data used to train BERT (16GB)
  - **CC-News**: English portion of the CommonCrawl News dataset. The data contains 63 million English news articles crawled between Sep. 2016 and Feb. 2019 (76GB after filtering)
  - **OpenWebText**: The text is web content extracted from URLs shared on Reddit with at least three upvotes (38GB), Recreation of WebText corpora for training GPT-2
  - **Stories**: A subset of CommonCrawl filtered to match the story-like style of Winograd schemas (31GB)

Implementation Details

ROBOT INTELLIGENCE LAB

- **BERT$_{BASE}$** ($L = 12, H = 768, A = 12, 110M$ params) configuration
  - **Static** vs. **Dynamic Masking**
    - The original BERT implementation performed masking once during data preprocessing, resulting in a single static mask.
    - Dynamic masking generates the masking pattern every time a sequence is fed to the model.

Masking	SQuAD 2.0	MNLI-m	SST-2
reference	76.3	84.3	92.8
Our reimplementation:			
static	78.3	84.3	92.5
dynamic	78.7	84.0	92.9

Table 1: Comparison between static and dynamic masking for BERT$_{BASE}$. We report F1 for SQuAD and accuracy for MNLI-m and SST-2. Reported results are medians over 5 random initializations (seeds). Reference results are from Yang et al. (2019).

Implementation Details

ROBOT INTELLIGENCE LAB

- **$\text{BERT}_{\text{BASE}}$** ($L = 12, H = 768, A = 12, 110M$ params) configuration
  - **Nest Sentence Prediction (NSP)**
    - The original BERT model emphasized the NSP objectives.
    - However, some recent work questioned the necessity of the NSP loss.

Model	SQuAD 1.1/2.0	MNLI-m	SST-2	RACE
Our reimplementation (with NSP loss):				
SEGMENT-PAIR	90.4/78.7	84.0	92.9	64.2
SENTENCE-PAIR	88.7/76.2	82.9	92.1	63.0
Our reimplementation (without NSP loss):				
FULL-SENTENCES	90.4/79.1	84.7	92.5	64.8
DOC-SENTENCES	90.6/79.7	84.7	92.7	65.6
$\text{BERT}_{\text{BASE}}$	88.5/76.3	84.3	92.8	64.3
$\text{XLNet}_{\text{BASE}}$ (K = 7)	–/81.3	85.8	92.7	66.1
$\text{XLNet}_{\text{BASE}}$ (K = 6)	–/81.0	85.6	93.4	66.7

Table 2: Development set results for base models pretrained over BOOKCORPUS and WIKIPEDIA. All models are trained for 1M steps with a batch size of 256 sequences. We report F1 for SQuAD and accuracy for MNLI-m, SST-2 and RACE. Reported results are medians over five random initializations (seeds). Results for $\text{BERT}_{\text{BASE}}$ and $\text{XLNet}_{\text{BASE}}$ are from Yang et al. (2019).

Implementation Details

ROBOT INTELLIGENCE LAB

- **BERT$_{BASE}$** ($L = 12, H = 768, A = 12, 110M$ params) configuration
  - **Training with large batches**
    - The original BERT trained **BERT$_{BASE}$** with 1M steps with a batch size of 256 sequences.
    - With the same computational cost, we can increase the batch size while reducing the steps.

bsz	steps	lr	ppl	MNLI-m	SST-2
256	1M	1e-4	3.99	84.7	92.7
2K	125K	7e-4	**3.68**	**85.2**	**92.9**
8K	31K	1e-3	3.77	84.6	92.8

Table 3: Perplexity on held-out training data (*ppl*) and development set accuracy for base models trained over BOOKCORPUS and WIKIPEDIA with varying batch sizes (*bsz*). We tune the learning rate (*lr*) for each setting. Models make the same number of passes over the data (epochs) and have the same computational cost.

Experiments

ROBOT INTELLIGENCE LAB

Model	data	bsz	steps	SQuAD (v1.1/2.0)	MNLI-m	SST-2
RoBERTa						
with BOOKS + WIKI	16GB	8K	100K	93.6/87.3	89.0	95.3
+ additional data (§3.2)	160GB	8K	100K	94.0/87.7	89.3	95.6
+ pretrain longer	160GB	8K	300K	94.4/88.7	90.0	96.1
+ pretrain even longer	160GB	8K	500K	**94.6/89.4**	**90.2**	**96.4**
$\text{BERT}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	90.9/81.8	86.6	93.7
$\text{XLNet}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	94.0/87.8	88.4	94.4
+ additional data	126GB	2K	500K	94.5/88.8	89.8	95.6

Table 4: Development set results for RoBERTa as we pretrain over more data (16GB → 160GB of text) and pretrain for longer (100K → 300K → 500K steps). Each row accumulates improvements from the rows above. RoBERTa matches the architecture and training objective of $\text{BERT}_{\text{LARGE}}$. Results for $\text{BERT}_{\text{LARGE}}$ and $\text{XLNet}_{\text{LARGE}}$ are from Devlin et al. (2019) and Yang et al. (2019), respectively. Complete results on all GLUE tasks can be found in the Appendix.

Experiments

ROBOT INTELLIGENCE LAB

	MNLI	QNLI	QQP	RTE	SST	MRPC	CoLA	STS	WNLI	Avg
Single-task single models on dev										
$\text{BERT}_{\text{LARGE}}$	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-	-
$\text{XLNet}_{\text{LARGE}}$	89.8/-	93.9	91.8	83.8	95.6	89.2	63.6	91.8	-	-
RoBERTa	**90.2/90.2**	**94.7**	**92.2**	**86.6**	**96.4**	**90.9**	**68.0**	**92.4**	**91.3**	-
Ensembles on test (from leaderboard as of July 25, 2019)										
ALICE	88.2/87.9	95.7	**90.7**	83.5	95.2	92.6	**68.6**	91.1	80.8	86.3
MT-DNN	87.9/87.4	96.0	89.9	86.3	96.5	92.7	68.4	91.1	89.0	87.6
XLNet	90.2/89.8	98.6	90.3	86.3	**96.8**	**93.0**	67.8	91.6	**90.4**	88.4
RoBERTa	**90.8/90.2**	**98.9**	90.2	**88.2**	96.7	92.3	67.8	**92.2**	89.0	**88.5**

Table 5: Results on GLUE. All results are based on a 24-layer architecture. $\text{BERT}_{\text{LARGE}}$ and $\text{XLNet}_{\text{LARGE}}$ results are from Devlin et al. (2019) and Yang et al. (2019), respectively. RoBERTa results on the development set are a median over five runs. RoBERTa results on the test set are ensembles of *single-task* models. For RTE, STS and MRPC we finetune starting from the MNLI model instead of the baseline pretrained model. Averages are obtained from the GLUE leaderboard.

ROBOT INTELLIGENCE LAB

ALBERT

"ALBERT: A LITE BERT FOR SELF-SUPERVISED LEARNING OF LANGUAGE REPRESENTATIONS," 2020

ALBERT

ROBOT INTELLIGENCE LAB

A Lite BERT (ALBERT)

ALBERT

- ALBERT presents two parameter-reduction techniques to lower memory consumption and increase the training speed of BERT.
  - The first one is a **factorized embedding parametrization** by decomposing the large vocabulary embedding matrix into two small matrices.
  - The second technique is **cross-layer parameter sharing** which prevents the parameter from growing with the depth of the network.
- A self-supervised loss focussing on modeling inter-sentence coherence is used.
  - A **sentence-order prediction** (SOP) focuses on inter-sentence coherence and is designed to address the ineffectiveness of the next sentence prediction (NSP) loss proposed in the original BERT.

ALBERT

ROBOT INTELLIGENCE LAB

- Factorized embedding parametrization
  - The size of the embedding matrix has the size of $V \times H$ where V is the vocabulary size and H is the size of hidden space.
  - By projecting them into a lower dimensional embedding space of size E, the embedding parameters are reduced from $O(V \times H)$ to $O(V \times E + E \times H)$. This parameter reduction is significant when $H \gg E$.
- Cross-layer parameter sharing
  - There are multiple ways to share parameters (e.g., only sharing feed-forward network parameters across layers, or only sharing attention parameters, or all parameters which is the default decision for ALBERT).
- Inter-sentence coherence loss
  - The sentence-order-prediction (SOP) task uses as positive examples the same as BERT and as negative examples the same two consecutive segments but with their order swapped.

Model

ROBOT INTELLIGENCE LAB

Model		Parameters	Layers	Hidden	Embedding	Parameter-sharing
BERT	base	108M	12	768	768	False
	large	334M	24	1024	1024	False
ALBERT	base	12M	12	768	128	True
	large	18M	24	1024	128	True
	xlarge	60M	24	2048	128	True
	xxlarge	235M	12	4096	128	True

Table 1: The configurations of the main BERT and ALBERT models analyzed in this paper.

- ALBERT-large has about 18x fewer parameters compare to BERT-large, 18M versus 334M. An ALBERT-xlarge configuration with $H = 2048$ has $233M$ parameters (i.e., around 70% of BERT-large's parameters).

Experiments

ROBOT INTELLIGENCE LAB

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3	4.7x
	large	334M	92.2/85.5	85.0/82.2	86.6	93.0	73.9	85.2	1.0
ALBERT	base	12M	89.3/82.3	80.0/77.1	81.6	90.3	64.0	80.1	5.6x
	large	18M	90.6/83.9	82.3/79.4	83.5	91.7	68.5	82.4	1.7x
	xlarge	60M	92.5/86.1	86.1/83.1	86.4	92.4	74.8	85.5	0.6x
	xxlarge	235M	**94.1/88.3**	**88.1/85.1**	**88.0**	**95.2**	**82.3**	**88.7**	0.3x

Table 2: Dev set results for models pretrained over BOOKCORPUS and Wikipedia for 125k steps. Here and everywhere else, the Avg column is computed by averaging the scores of the downstream tasks to its left (the two numbers of F1 and EM for each SQuAD are first averaged).

Experiments

ROBOT INTELLIGENCE LAB

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
ALBERT base E=768	all-shared	31M	88.6/81.5	79.2/76.6	82.0	90.6	63.3	79.8
	shared-attention	83M	89.9/82.7	80.0/77.2	84.0	91.4	67.7	81.6
	shared-FFN	57M	89.2/82.1	78.2/75.4	81.5	90.8	62.6	79.5
	not-shared	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3
ALBERT base E=128	all-shared	12M	89.3/82.3	80.0/77.1	82.0	90.3	64.0	80.1
	shared-attention	64M	89.9/82.8	80.7/77.9	83.4	91.9	67.6	81.7
	shared-FFN	38M	88.9/81.6	78.6/75.6	82.3	91.7	64.4	80.2
	not-shared	89M	89.9/82.8	80.3/77.3	83.2	91.5	67.9	81.6

Table 4: The effect of cross-layer parameter-sharing strategies, ALBERT-base configuration.

Experiments

ROBOT INTELLIGENCE LAB

	Intrinsic Tasks			Downstream Tasks					
SP tasks	MLM	NSP	SOP	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
None	54.9	52.4	53.3	88.6/81.5	78.1/75.3	81.5	89.9	61.7	79.0
NSP	54.5	90.5	52.0	88.4/81.5	77.2/74.6	81.6	**91.1**	62.3	79.2
SOP	54.0	78.9	86.5	**89.3/82.3**	**80.0/77.1**	**82.0**	90.3	**64.0**	**80.1**

Table 5: The effect of sentence-prediction loss, NSP vs. SOP, on intrinsic and downstream tasks.

Models	Steps	Time	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
BERT-large	400k	34h	93.5/87.4	86.9/84.3	87.8	94.6	77.3	87.2
ALBERT-xxlarge	125k	32h	**94.0/88.1**	**88.3/85.3**	87.8	**95.4**	**82.5**	**88.7**

Table 6: The effect of controlling for training time, BERT-large vs ALBERT-xxlarge configurations

Experiments

ROBOT INTELLIGENCE LAB

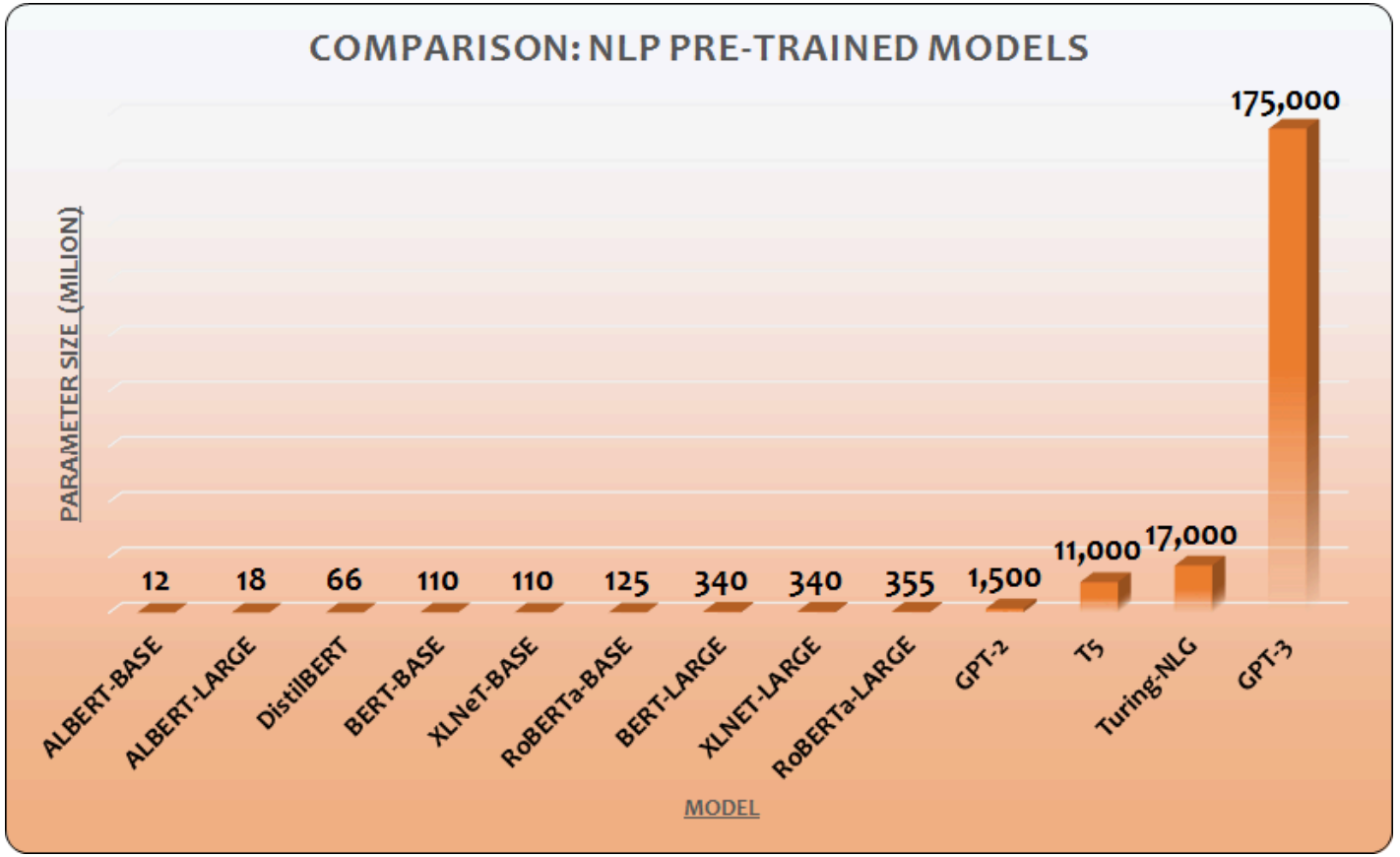
Models	MNLI	QNLI	QQP	RTE	SST	MRPC	CoLA	STS	WNLI	Avg
Single-task single models on dev										
BERT-large	86.6	92.3	91.3	70.4	93.2	88.0	60.6	90.0	-	-
XLNet-large	89.8	93.9	91.8	83.8	95.6	89.2	63.6	91.8	-	-
RoBERTa-large	90.2	94.7	**92.2**	86.6	96.4	**90.9**	68.0	92.4	-	-
ALBERT (1M)	90.4	95.2	92.0	88.1	96.8	90.2	68.7	92.7	-	-
ALBERT (1.5M)	**90.8**	**95.3**	**92.2**	**89.2**	**96.9**	**90.9**	**71.4**	**93.0**	-	-
Ensembles on test (from leaderboard as of Sept. 16, 2019)										
ALICE	88.2	95.7	**90.7**	83.5	95.2	92.6	**69.2**	91.1	80.8	87.0
MT-DNN	87.9	96.0	89.9	86.3	96.5	92.7	68.4	91.1	89.0	87.6
XLNet	90.2	98.6	90.3	86.3	96.8	93.0	67.8	91.6	90.4	88.4
RoBERTa	90.8	98.9	90.2	88.2	96.7	92.3	67.8	92.2	89.0	88.5
Adv-RoBERTa	91.1	98.8	90.3	88.7	96.8	93.1	68.0	92.4	89.0	88.8
ALBERT	**91.3**	**99.2**	90.5	**89.2**	**97.1**	**93.4**	69.1	**92.5**	**91.8**	**89.4**

Table 9: State-of-the-art results on the GLUE benchmark. For single-task single-model results, we report ALBERT at 1M steps (comparable to RoBERTa) and at 1.5M steps. The ALBERT ensemble uses models trained with 1M, 1.5M, and other numbers of steps.

GPT-3

"Language Models are Few-Shot Learners," 2020

GPT-3



https://medium.com/analytics-vidhya/openai-gpt-3-language-models-are-few-shot-learners-82531b3d3122

GPT-3

ROBOT INTELLIGENCE LAB

What is new besides the model size has increased over 10 times?

Datasets

ROBOT INTELLIGENCE LAB

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3. "Weight in training mix" refers to the fraction of examples during training that are drawn from a given dataset, which we intentionally do not make proportional to the size of the dataset. As a result, when we train for 300 billion tokens, some datasets are seen up to 3.4 times during training while other datasets are seen less than once.

Dataset

- Three different techniques are used (1) **filtering CommonCrawl**, (2) fuzzy deduplication, (3) added high-quality reference corpora to the training mix to augment CommonCrawl
  - An **automatic filtering method** is developed to remove low quality documents. A classifier is trained by using the original WebText as a proxy for high-quality documents to distinguish low quality ones from raw CommonCrawl.
  - The classifier (logistic regression) is used to re-sample CommonCrawl by prioritizing documents which were predicted by the classifier to be higher quality. The positive examples include curated datasets such as **WebText**, **Wikipedia**, and **web books corpus**, and the negative examples include **unfiltered Common Crawl**.

Dataset

ROBOT INTELLIGENCE LAB

- Three different techniques are used (1) filtering CommonCrawl, (2) **fuzzy deduplication**, (3) added high-quality reference corpora to the training mix to augment CommonCrawl
  - To improve the model quality and prevent overfitting, the authors fuzzily deduplicated documents (i.e., removed documents with high overlap with other documents). WebText was fuzzily removed from Common Crawl.
  - Overall, this decreases dataset size by an average of 10%.

In-context Learning

ROBOT INTELLIGENCE LAB

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  cheese =>                           ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← example
3  cheese =>                           ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```

Implementation Details

ROBOT INTELLIGENCE LAB

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

Experiments

ROBOT INTELLIGENCE LAB

Title: United Methodists Agree to Historic Split
Subtitle: Those who oppose gay marriage will form their own denomination
Article: After two days of intense debate, the United Methodist Church has agreed to a historic split - one that is expected to end in the creation of a new denomination, one that will be "theologically and socially conservative," according to The Washington Post. The majority of delegates attending the church's annual General Conference in May voted to strengthen a ban on the ordination of LGBTQ clergy and to write new rules that will "discipline" clergy who officiate at same-sex weddings. But those who opposed these measures have a new plan: They say they will form a separate denomination by 2020, calling their church the Christian Methodist denomination.
The Post notes that the denomination, which claims 12.5 million members, was in the early 20th century the "largest Protestant denomination in the U.S.," but that it has been shrinking in recent decades. The new split will be the second in the church's history. The first occurred in 1968, when roughly 10 percent of the denomination left to form the Evangelical United Brethren Church. The Post notes that the proposed split "comes at a critical time for the church, which has been losing members for years," which has been "pushed toward the brink of a schism over the role of LGBTQ people in the church." Gay marriage is not the only issue that has divided the church. In 2016, the denomination was split over ordination of transgender clergy, with the North Pacific regional conference voting to ban them from serving as clergy, and the South Pacific regional conference voting to allow them.

Figure 3.14: The GPT-3 generated news article that humans had the greatest difficulty distinguishing from a human written article (accuracy: 12%).

Experiments

ROBOT INTELLIGENCE LAB

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:
We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:
One day when I was playing tag with my little sister, she got really excited and she started doing these crazy farduddles.
A "yalubalu" is a type of vegetable that looks like a big pumpkin. An example of a sentence that uses the word yalubalu is:
I was on a trip to Africa and I tried this yalubalu vegetable that was grown in a garden there. It was delicious.

A "Burringo" is a car with very fast acceleration. An example of a sentence that uses the word Burringo is:
In our garage we have a Burringo that my father drives to work every day.

A "Gigamuru" is a type of Japanese musical instrument. An example of a sentence that uses the word Gigamuru is:
I have a Gigamuru that my uncle gave me as a gift. I love to play it at home.

To "screeg" something is to swing a sword at it. An example of a sentence that uses the word screeg is:
We screeghed at each other for several minutes and then we went outside and ate ice cream.

Figure 3.16: Representative GPT-3 completions for the few-shot task of using a new word in a sentence. Boldface is GPT-3's completions, plain text is human prompts. In the first example both the prompt and the completion are provided by a human; this then serves as conditioning for subsequent examples where GPT-3 receives successive additional prompts and provides the completions. Nothing task-specific is provided to GPT-3 other than the conditioning shown here.

Experiments

ROBOT INTELLIGENCE LAB

Poor English input: I eated the purple berries.
Good English output: I ate the purple berries.
Poor English input: Thank you for picking me as your designer. I'd appreciate it.
Good English output: Thank you for choosing me as your designer. I appreciate it.
Poor English input: The mentioned changes have done. or I did the alteration that you requested. or I changed things you wanted and did the modifications.
Good English output: The requested changes have been made. or I made the alteration that you requested. or I changed things you wanted and made the modifications.
Poor English input: I'd be more than happy to work with you in another project.
Good English output: I'd be more than happy to work with you on another project.

Poor English input: Please provide me with a short brief of the design you're looking for and that'd be nice if you could share some examples or project you did before.
Good English output: Please provide me with a brief description of the design you're looking for and that would be nice if you could share some examples or projects you have done before.

Poor English input: The patient was died.
Good English output: The patient died.

Poor English input: We think that Leslie likes ourselves.
Good English output: We think that Leslie likes us.

Poor English input: Janet broke Bill on the finger.
Good English output: Janet broke Bill's finger.

Poor English input: Mary arranged for, in St. Louis, John to rent a house cheap.
Good English output: Mary arranged for John to rent a house in St. Louis.

Poor English input: Today I have went to the store to to buys some many bottle of water.
Good English output: Today I went to the store to buy some bottles of water.

Poor English input: I have tried to hit ball with bat, but my swing is has miss.
Good English output: I tried to hit the ball with the bat, but my swing missed.

Figure 3.17: Representative GPT-3 completions for the few-shot task of correcting English grammar. Boldface is GPT-3's completions, plain text is human prompts. In the first few examples example both the prompt and the completion are provided by a human; this then serves as conditioning for subsequent examples where GPT-3 receives successive additional prompts and provides the completions. Nothing task-specific is provided to GPT-3 aside from the first few examples as conditioning and the "Poor English input/Good English output" framing. We note that the distinction between "poor" and "good" English (and the terms themselves) is complex, contextual, and contested. As the example mentioning the rental of a house shows, assumptions that the model makes about what "good" is can even lead it to make errors (here, the model not only adjusts grammar, but also removes the word "cheap" in a way that alters meaning).

Thank You