RoBERTa and ELECTRA

Amin Saied

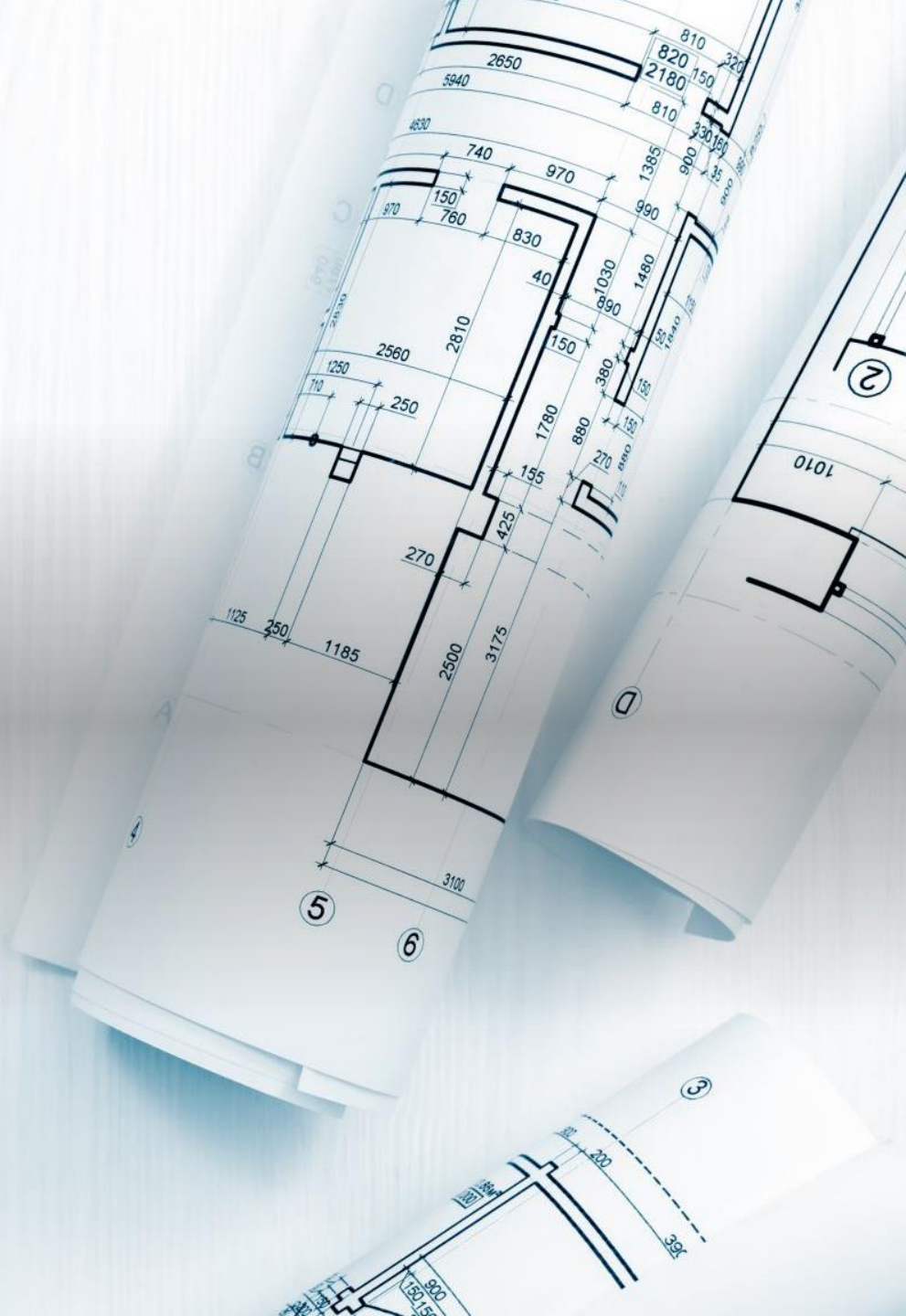
2021-09-24

Overview

- RoBERTa
  - Replication study of BERT pretraining
- ELECTRA
  - New pre-training objective: *replaced token detection*

For background click here ->

RoBERTa



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, and raise questions about the source of recently reported improvements. We release our models and code.[1]

Proposed changes

- Train model longer, larger batches
- Remove “Next sentence prediction” objective
- Increase sequence length
- Dynamic masking

More Data

- Book Corpus + English Wikipedia
  - 16 GB
  - Original used in BERT
- CC-News
  - Gathered as part of this work
  - 76 GB
  - English news articles from CommonCrawl
- Open Web Text
  - 38 GB
  - Uses reddit to filter to high quality web pages
- Stories
  - 31 GB
  - Also from CommonCrawl – more like story-like

Larger batch size

Original BERT ->

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

$$PP(p) := 2^{H(p)} = 2^{-\sum_x p(x)\log_2 p(x)} = \prod_x p(x)^{-p(x)}$$

Pretraining Objectives

- Masked Language Modeling (MLM)

  Into this wild abyss, the womb of nature, and perhaps her grave

  Into this [MASK] abyss, the womb of nature, and [MASK] her grave

- Next Sentence Prediction (NSP)

  Of neither sea, not shore, nor air, nor fire, but all these → 0 or 1?

SEGMENT-PAIR+NSP: This follows the original input format used in BERT (Devlin et al., 2019), with the NSP loss. Each input has a pair of segments, which can each contain multiple natural sentences, but the total combined length must be less than 512 tokens.

SENTENCE-PAIR+NSP: Each input contains a pair of natural *sentences*, either sampled from a contiguous portion of one document or from separate documents. Since these inputs are significantly shorter than 512 tokens, we increase the batch size so that the total number of tokens remains similar to SEGMENT-PAIR+NSP. We retain the NSP loss.

FULL-SENTENCES: Each input is packed with full sentences sampled contiguously from one or more documents, such that the total length is at most 512 tokens. Inputs may cross document boundaries. When we reach the end of one document, we begin sampling sentences from the next document and add an extra separator token between documents. We remove the NSP loss.

DOC-SENTENCES: Inputs are constructed similarly to FULL-SENTENCES, except that they may not cross document boundaries. Inputs sampled near the end of a document may be shorter than 512 tokens, so we dynamically increase the batch size in these cases to achieve a similar number of total tokens as FULL-SENTENCES. We remove 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).

Static vs Dynamic Masking

- Static: generate masks (on 15% of tokens) once, reuse.
  - Training data was duplicated 10 times to help
- Dynamic:
  - Generate masking pattern randomly each time a sentence 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

Overall Results

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.

	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.

ELECTRA

ELECTRA: PRE-TRAINING TEXT ENCODERS AS DISCRIMINATORS RATHER THAN GENERATORS

Kevin Clark
Stanford University
kevclark@cs.stanford.edu

Minh-Thang Luong
Google Brain
thangluong@google.com

Quoc V. Le
Google Brain
qvl@google.com

Christopher D. Manning
Stanford University & CIFAR Fellow
manning@cs.stanford.edu

ABSTRACT

Masked language modeling (MLM) pre-training methods such as BERT corrupt the input by replacing some tokens with [MASK] and then train a model to reconstruct the original tokens. While they produce good results when transferred to downstream NLP tasks, they generally require large amounts of compute to be effective. As an alternative, we propose a more sample-efficient pre-training task called replaced token detection. Instead of masking the input, our approach corrupts it by replacing some tokens with plausible alternatives sampled from a small generator network. Then, instead of training a model that predicts the original identities of the corrupted tokens, we train a discriminative model that predicts whether each token in the corrupted input was replaced by a generator sample or not. Thorough experiments demonstrate this new pre-training task is more efficient than MLM because the task is defined over *all* input tokens rather than just the small subset that was masked out. As a result, the contextual representations learned by our approach substantially outperform the ones learned by BERT given the same model size, data, and compute. The gains are particularly strong for small models; for example, we train a model on one GPU for 4 days that outperforms GPT (trained using 30x more compute) on the GLUE natural language understanding benchmark. Our approach also works well at scale, where it performs comparably to RoBERTa and XLNet while using less than 1/4 of their compute and outperforms them when using the same amount of compute.

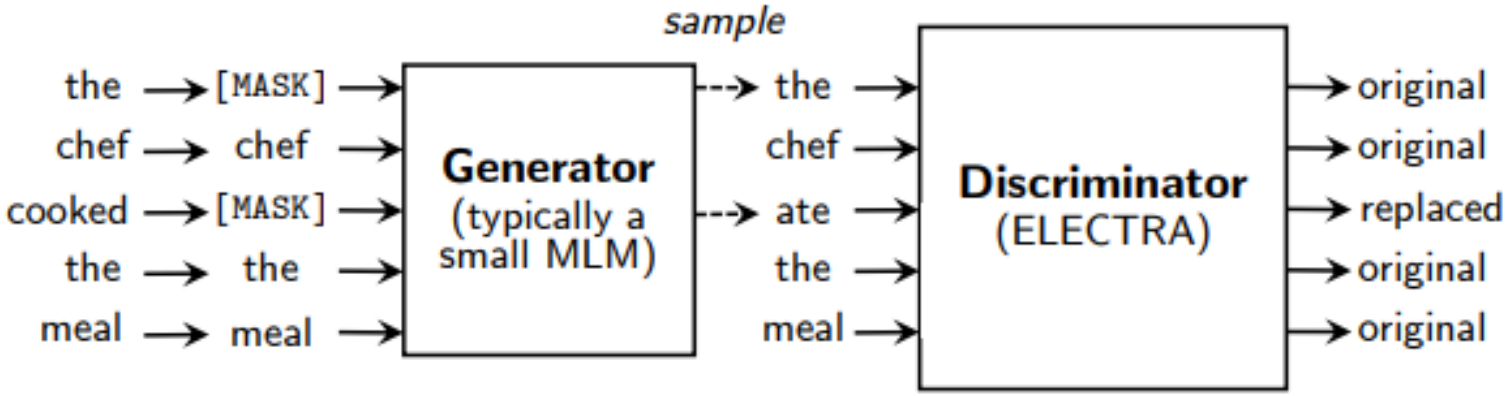
Into this wild abyss, the womb of nature, and perhaps her grave

Into this [MASK] abyss, the womb of [MASK], and perhaps her grave

Into this strange abyss, the birthplace of nature, and perhaps her demise

Replaced token detection

1. Corrupt input by replacing some tokens with samples
   - Use small LM as proposal model
2. Pre-train as a discriminator
   - For **every token** predict if it is original or replacement



Upshot

- Trains using every token, not just masked tokens
- Avoids pre-training mismatch problem
  - MASK token only seen in pretraining

- Note:
  - Similar idea to GAN
  - Key difference: the corrupting model is not adversarial!
    - The generating procedure is trained with maximum likelihood

- Note: Architecture and most hyperparameters are the same as original BERT

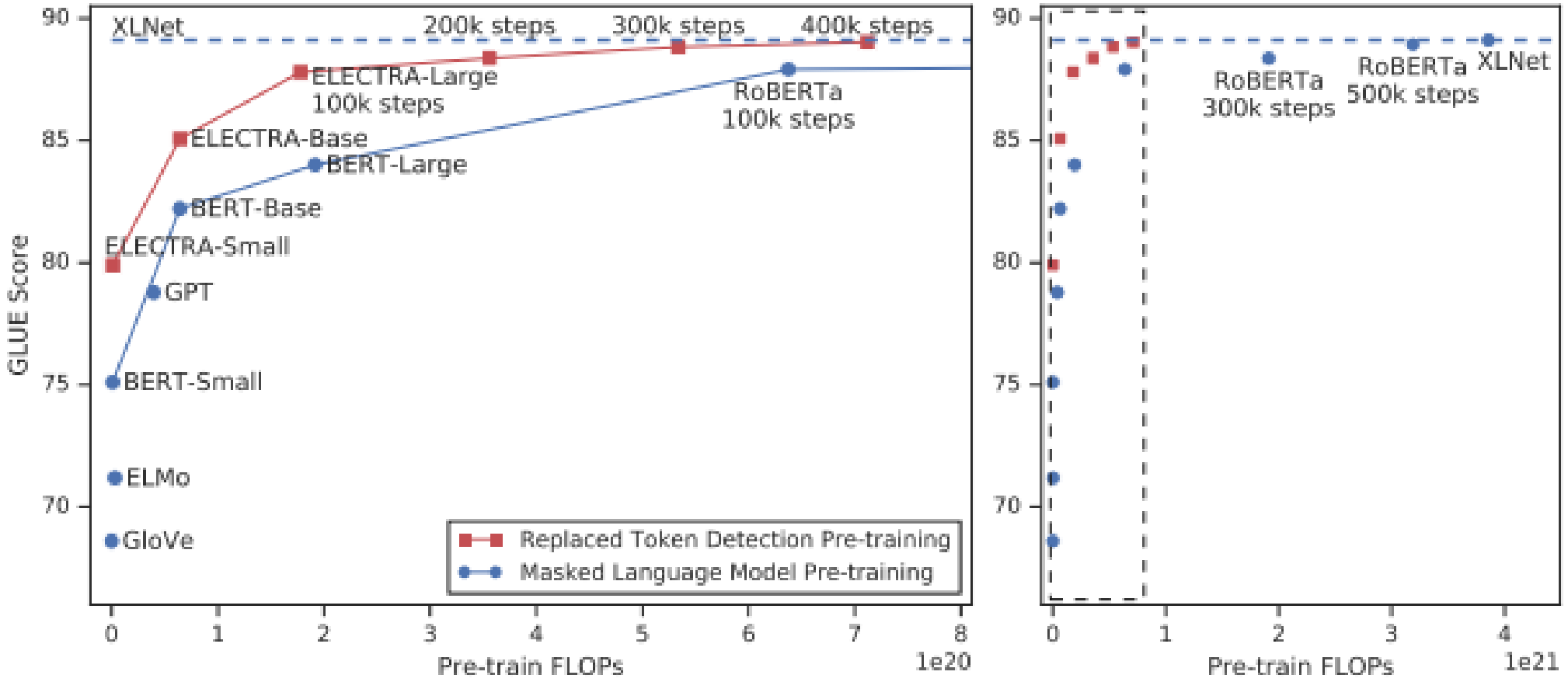
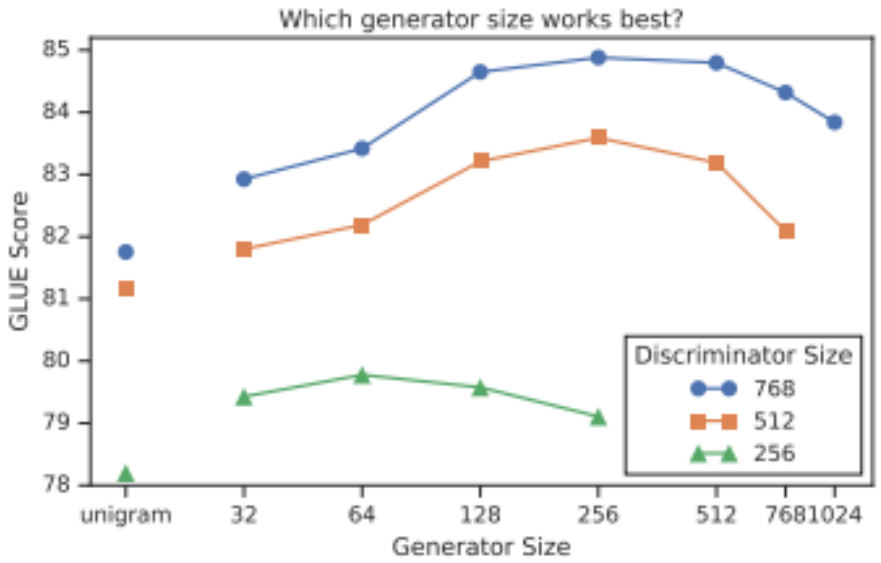


Figure 1: Replaced token detection pre-training consistently outperforms masked language model pre-training given the same compute budget. The left figure is a zoomed-in view of the dashed box.

Small generator

- If generator was same size this would be inefficient!



- We speculate that having too strong of a generator may pose a too-challenging task for the discriminator, preventing it from learning as effectively

Appendix

- We reference previous talk by way of background

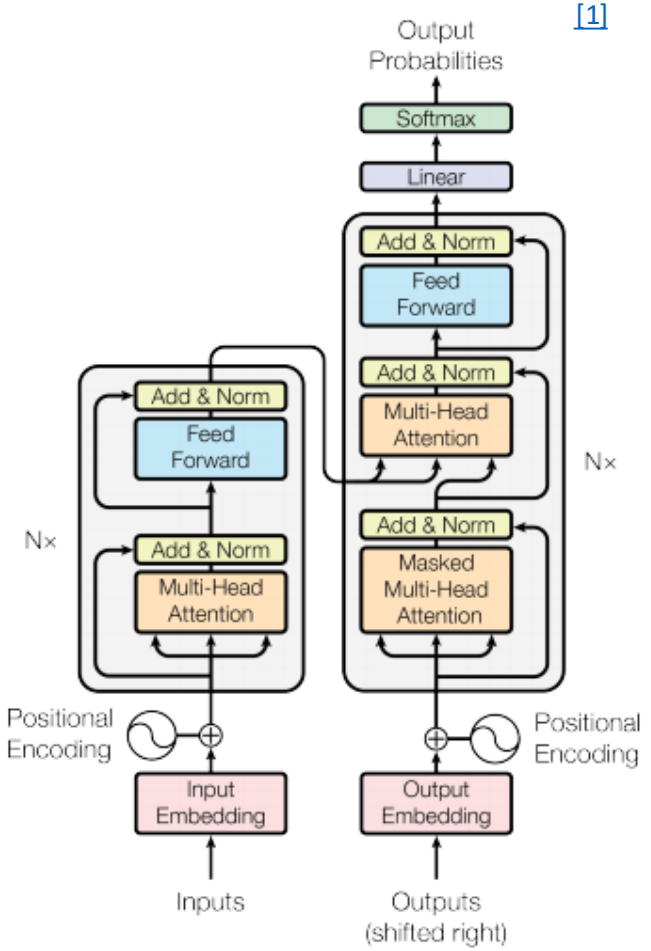
Background


Overview
RoBERTa
Replication study of BERT pretraining
ELECTRA
New pre-training objective: replaced token detection
For background click here ->

Transformers: Life before GPT-1

- Sequence-to-sequence model
- Evolution of RNNs
- Review:
  - Self-attention
  - Multi-headed attention
  - Encoder/decoder

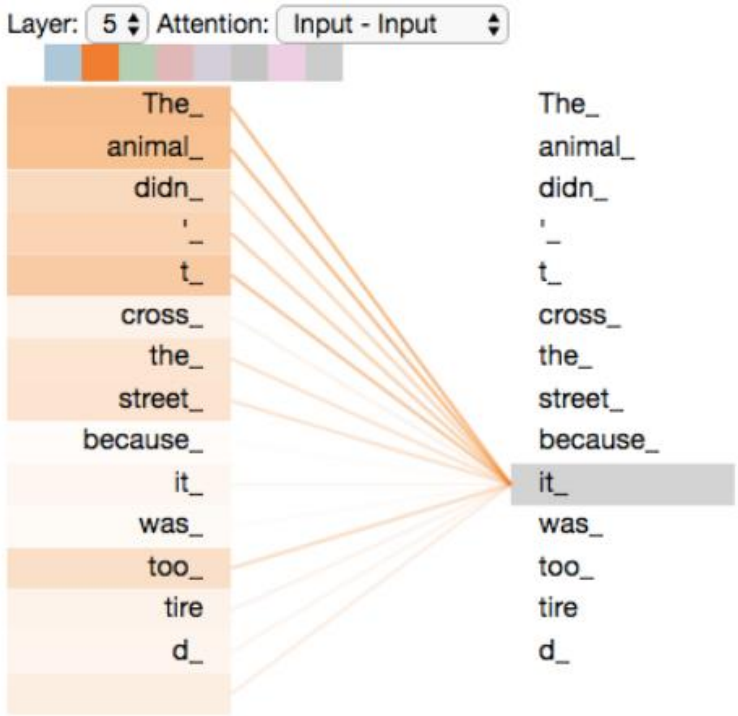
[1]



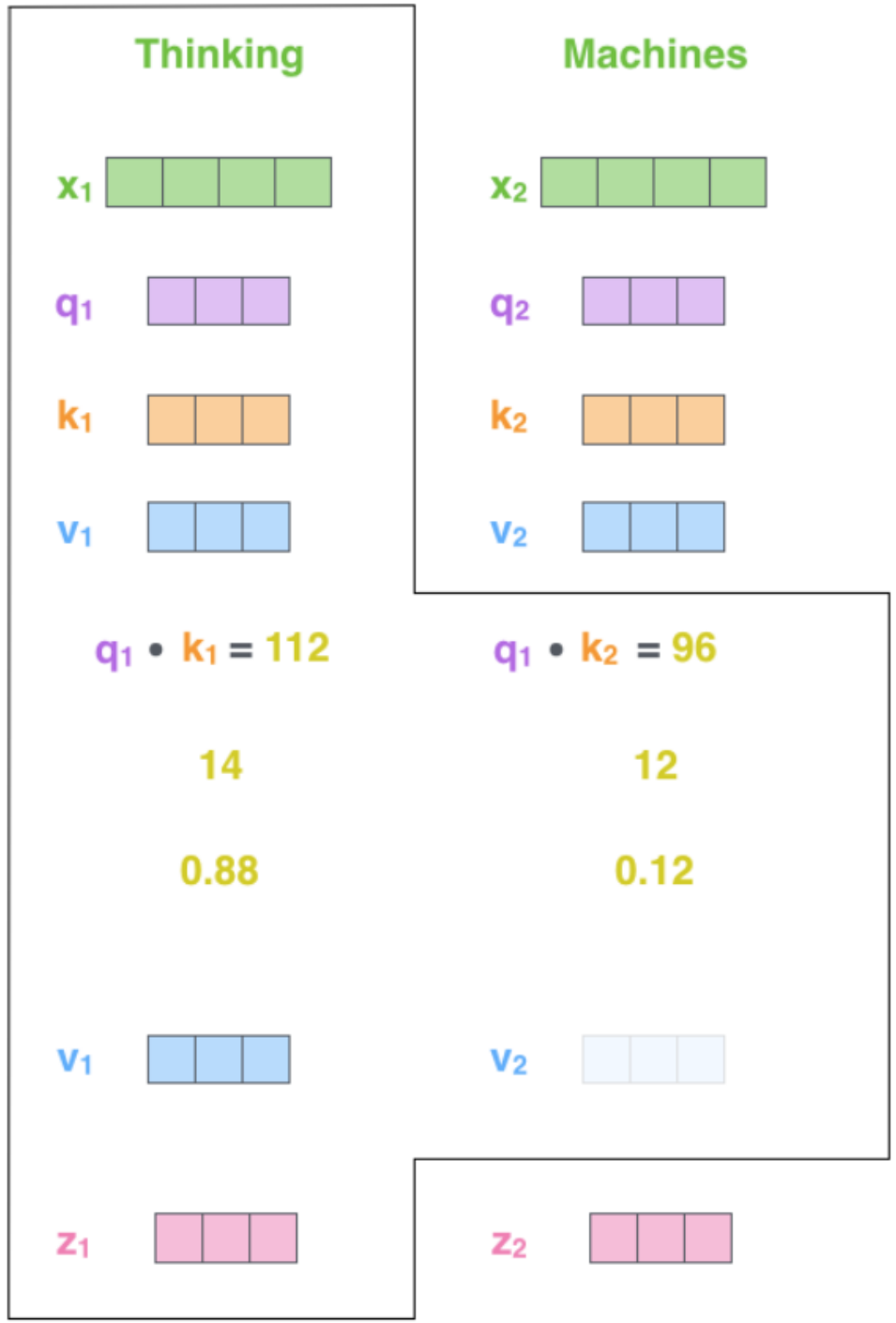
[1] Attention is all you need - Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, Polosukhin

Breaking down **self-attention**

"The animal didn't cross the street because it was too tired"



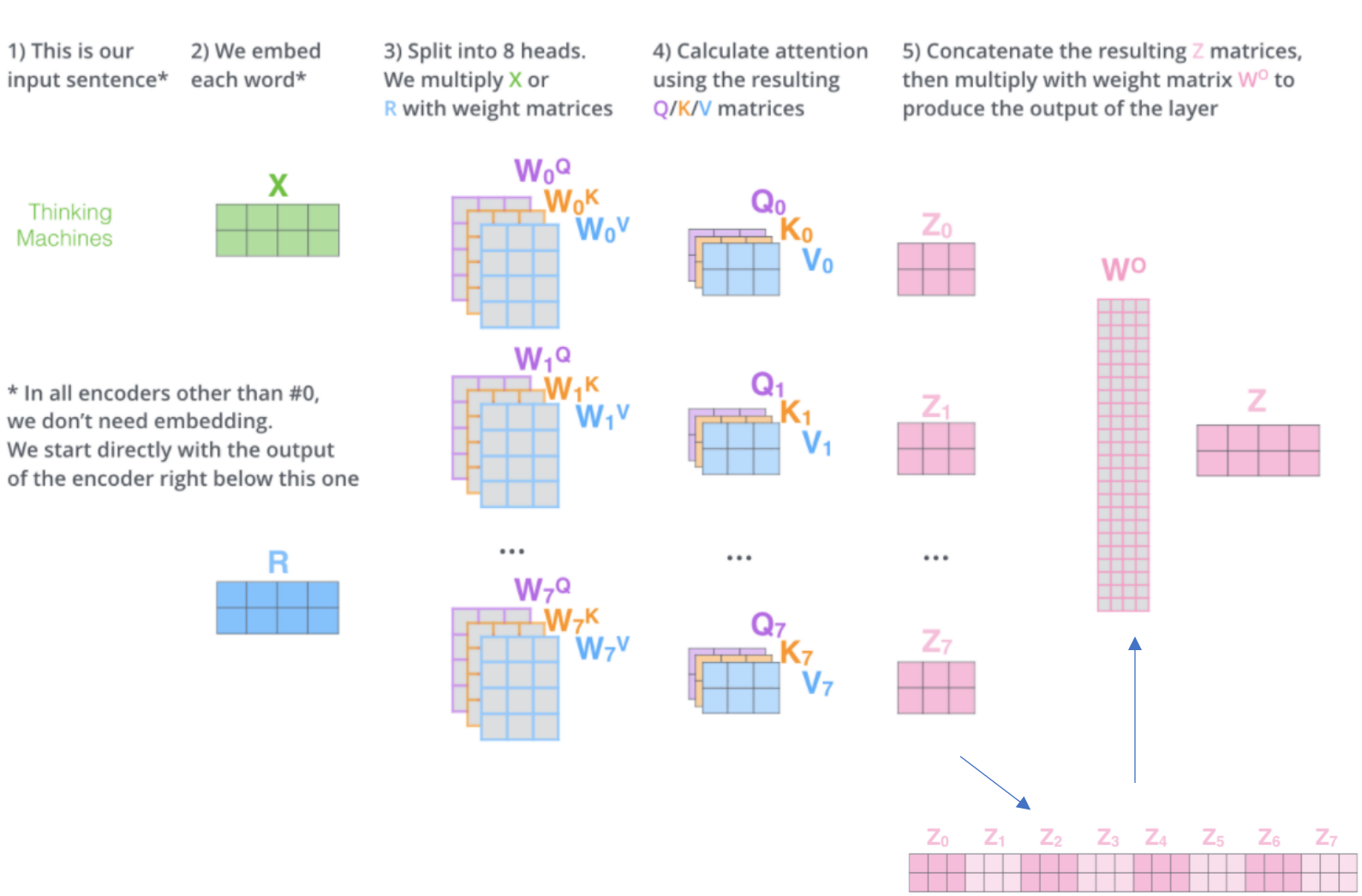
$$Z := softmax\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V$$

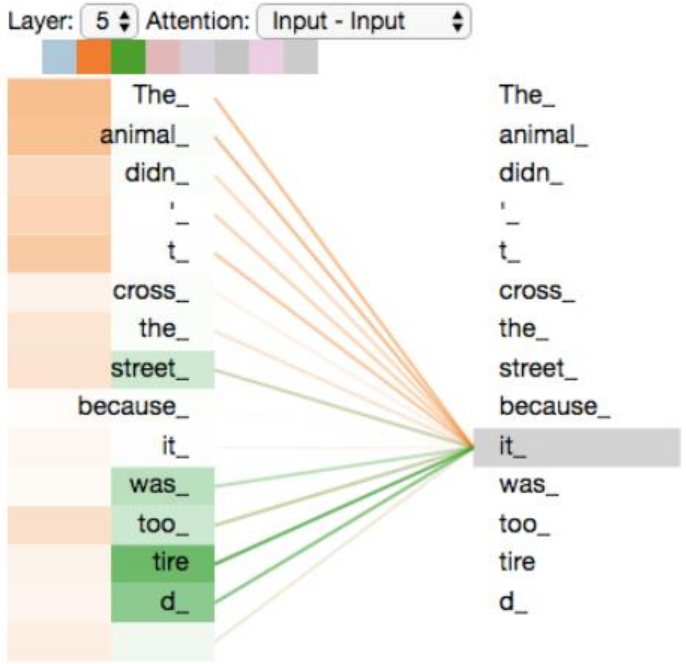


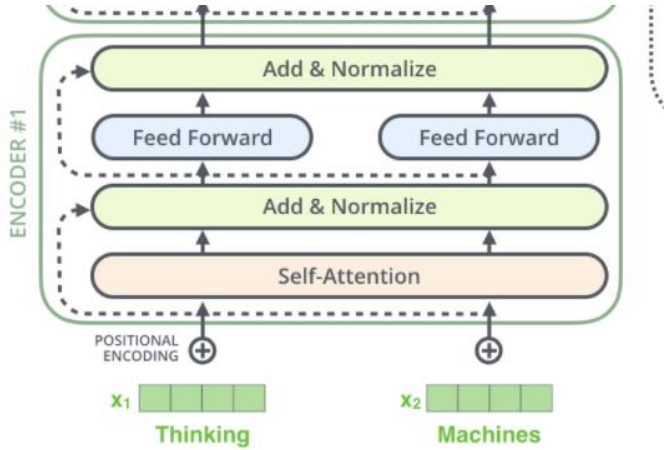
[The illustrated transformer](#) – Jay Alammar (Blog)

[Attention is all you need - Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, Polosukhin](#)

Breaking down **multi-headed** self-attention



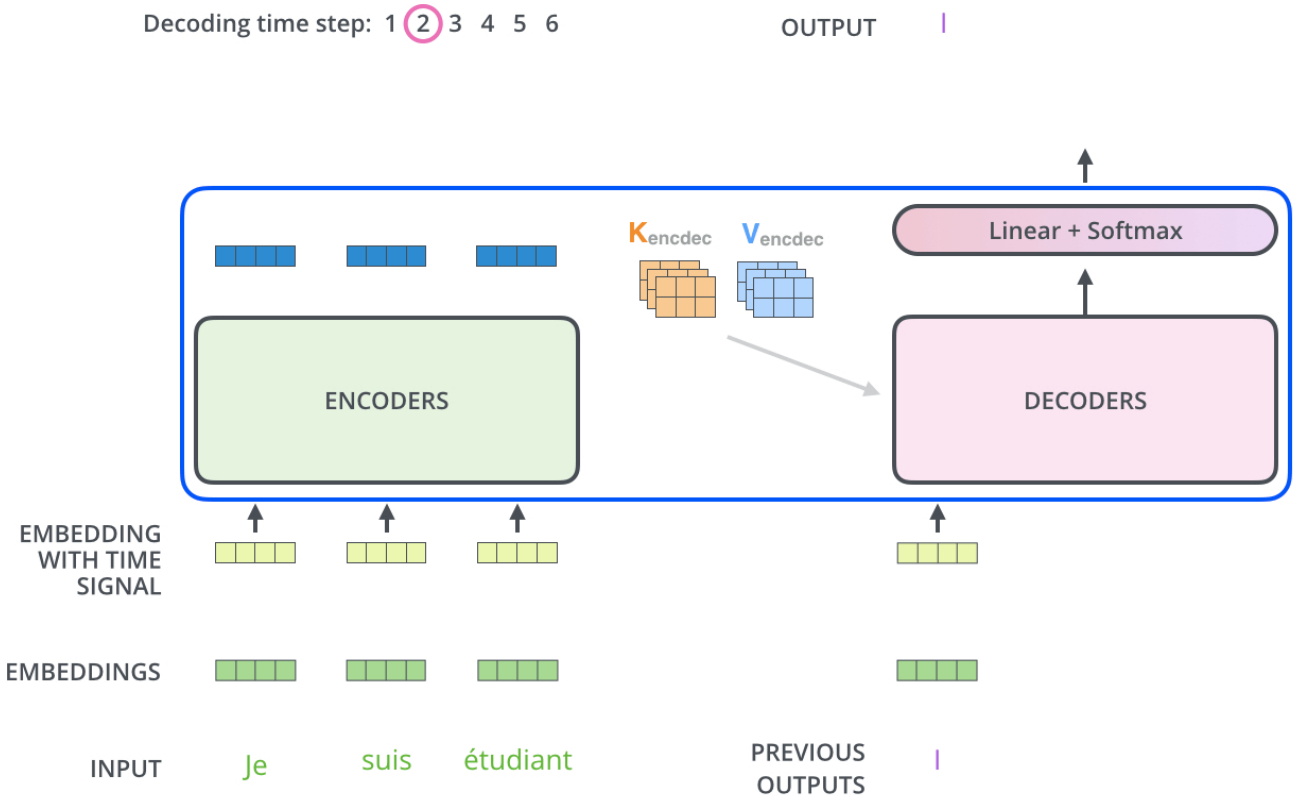




[The illustrated transformer](#) – Jay Alammar (Blog)

[Attention is all you need - Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, Polosukhin](#)

Encoder / decoder

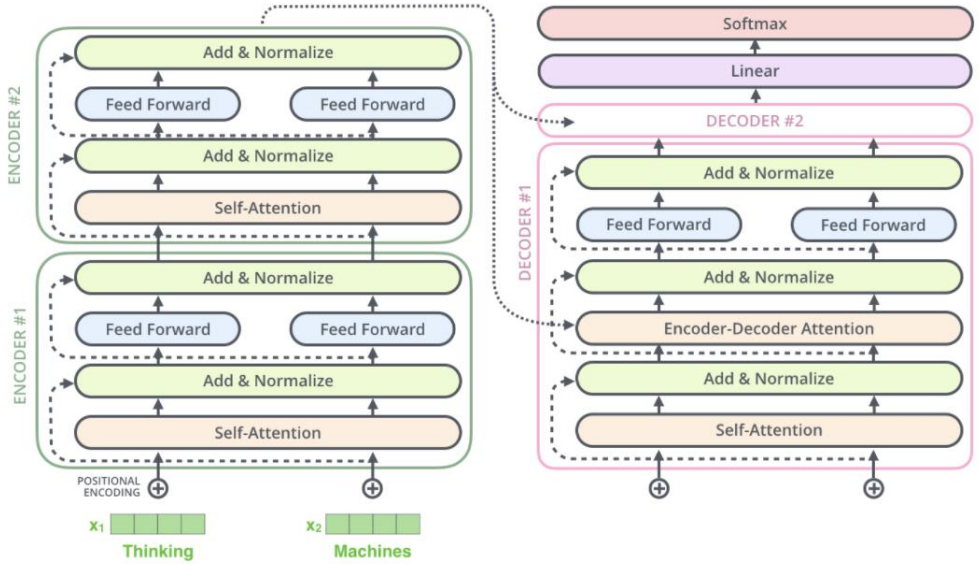


Encoder

- Full-context
- (Contextual) Word-embeddings

Decoder

- Left-context (masking)
- => predict next word



[The illustrated transformer](#) – Jay Alammar (Blog)

[Attention is all you need - Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, Polosukhin](#)

Glossary

- Tokens
- Attention (self-attention, multi-headed)
- Transformer
- Encoder / decoder


GPT 1

GPT-1

Improving Language Understanding by Generative Pre-Training – Radford et al

Key takeaways

- Semi-supervised learning with transformers
  - Pretraining / finetuning
- Decoder-only architecture
- Simplified approach to transfer learning

=>

- SOTA in 9/12 tasks studied

Language modelling (unsupervised approach)

3.1 Unsupervised pre-training

Given an unsupervised corpus of tokens $\mathcal{U} = \{u_1, \ldots, u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

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

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ. These parameters are trained using stochastic gradient descent [51].

- Different from above: this is unsupervised!
- Training data:
  - Data: Edon Lulzim Zhegrova (born 31 March 1999) is a Kosovan professional footballer who plays as a right winger for Swiss club Basel
  - Input: Edon Lulzim Zhegrova (born 31 March 1999) is a Kosovan professional
  - Output: Edon Lulzim Zhegrova (born 31 March 1999) is a Kosovan professional footballer
- Jargon: Auto-regressive language modelling
- Transfer learning in NLP!

Improving Language Understanding by Generative Pre-Training – Radford et al

Decoder-only architecture

- Based on previous work [2] using decoder-only transformer to generate Wikipedia articles
- Key-insight [2]: convert seq-to-seq task into language modelling task
  - Seq-to-seq: $(x_1, \dots, x_m) \mapsto (y_1, \dots, y_n)$
  - LM: $(x_1, \dots, x_m, \delta, y_1, \dots, y_n)$, where δ=separator token

$$p(w^1, ..., w^{n+\eta}) = \prod_{j=1}^{n+\eta} p(w^i | w^1, ..., w^{j-1})$$

- [1]: Semi-supervised approach!

[2] P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, and N. Shazeer. Generating wikipedia by summarizing long sequences. ICLR, 2018.

[1] Improving Language Understanding by Generative Pre-Training – Radford et al

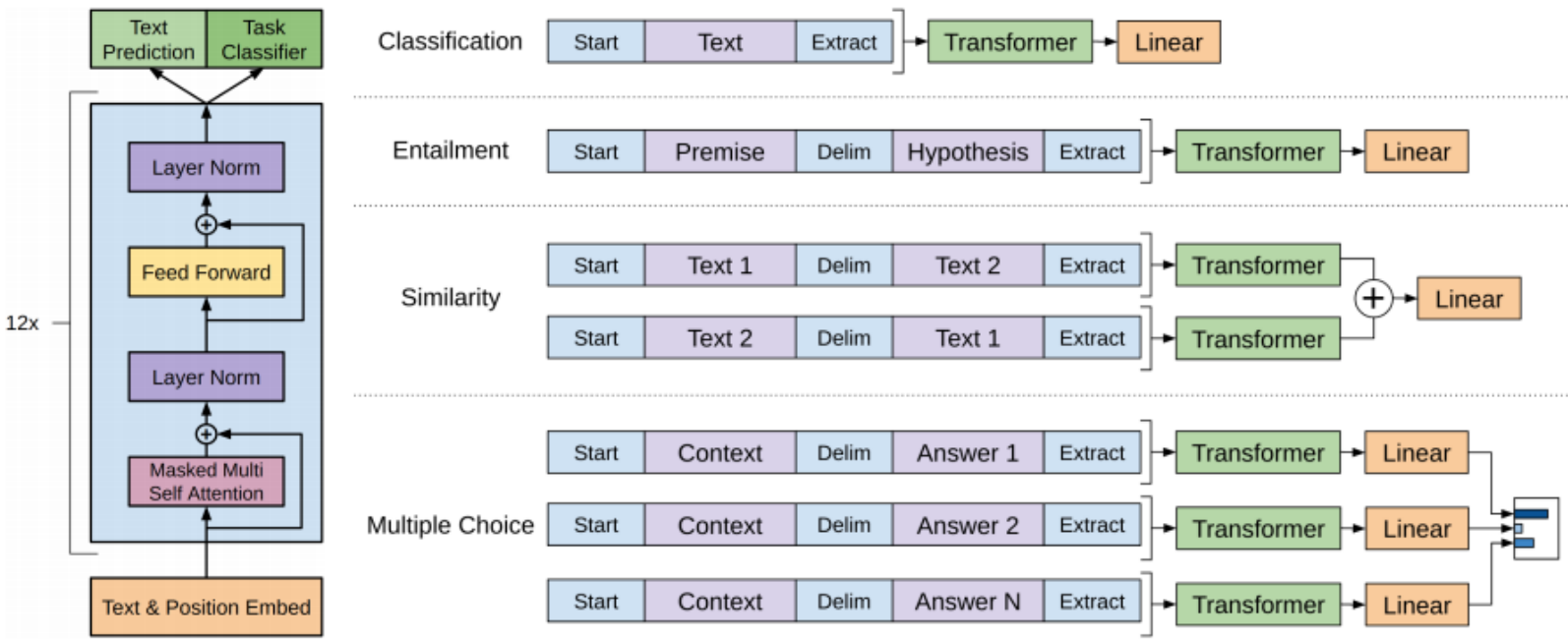
3.2 Supervised fine-tuning

After training the model with the objective in Eq. 1, we adapt the parameters to the supervised target task. We assume a labeled dataset $\mathcal{C}$, where each instance consists of a sequence of input tokens, $x^1, \ldots, x^m$, along with a label y. The inputs are passed through our pre-trained model to obtain the final transformer block's activation h_l^m, which is then fed into an added linear output layer with parameters W_y to predict y:

$$P(y|x^1, \ldots, x^m) = \texttt{softmax}(h_l^m W_y). \tag{3}$$

This gives us the following objective to maximize:

$$L_2(\mathcal{C}) = \sum_{(x,y)} \log P(y|x^1, \ldots, x^m). \tag{4}$$



Improving Language Understanding by Generative Pre-Training – Radford et al

Glossary

- Tokens
- Attention (self-attention, multi-headed)
- Transformer
- Encoder / decoder
- **Pretrain / Finetune**
- **Language modelling**
- **Auto-regressive**

Experimental Results

Model specifications Our model largely follows the original transformer work [62]. We trained a 12-layer decoder-only transformer with masked self-attention heads (768 dimensional states and 12 attention heads). For the position-wise feed-forward networks, we used 3072 dimensional inner states. We used the Adam optimization scheme [27] with a max learning rate of 2.5e-4. The learning rate was increased linearly from zero over the first 2000 updates and annealed to 0 using a cosine schedule. We train for 100 epochs on minibatches of 64 randomly sampled, contiguous sequences of 512 tokens. Since layernorm [2] is used extensively throughout the model, a simple weight initialization of $N(0, 0.02)$ was sufficient. We used a bytepair encoding (BPE) vocabulary with 40,000 merges [53] and residual, embedding, and attention dropouts with a rate of 0.1 for regularization. We also employed a modified version of L2 regularization proposed in [37], with $w = 0.01$ on all non bias or gain weights. For the activation function, we used the Gaussian Error Linear Unit (GELU) [18]. We used learned position embeddings instead of the sinusoidal version proposed in the original work. We use the *ftfy* library[2] to clean the raw text in BooksCorpus, standardize some punctuation and whitespace, and use the *spaCy* tokenizer.[3]

- 12 layer decoder
- 768 dim hidden states
- 12 attention heads (multi-headed attention)

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

Question Answering

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**

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**

Natural language inference

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

Improving Language Understanding by Generative Pre-Training – Radford et al

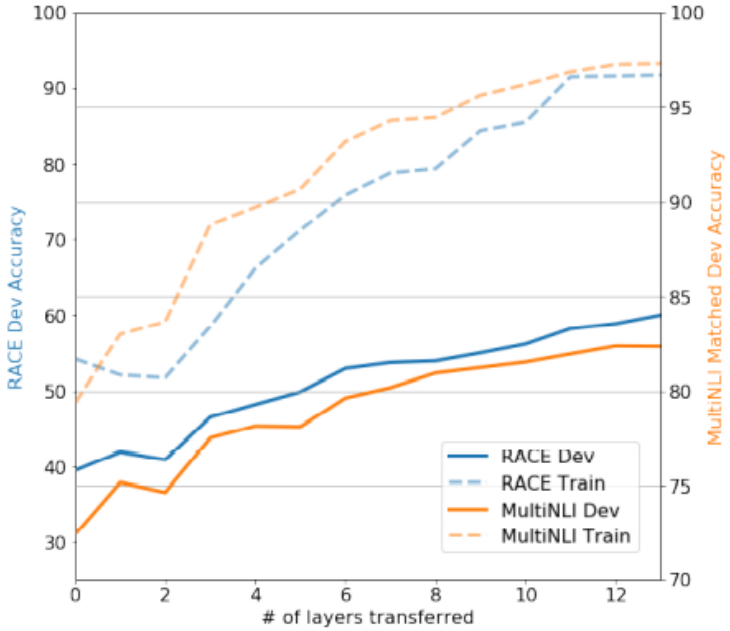
Details

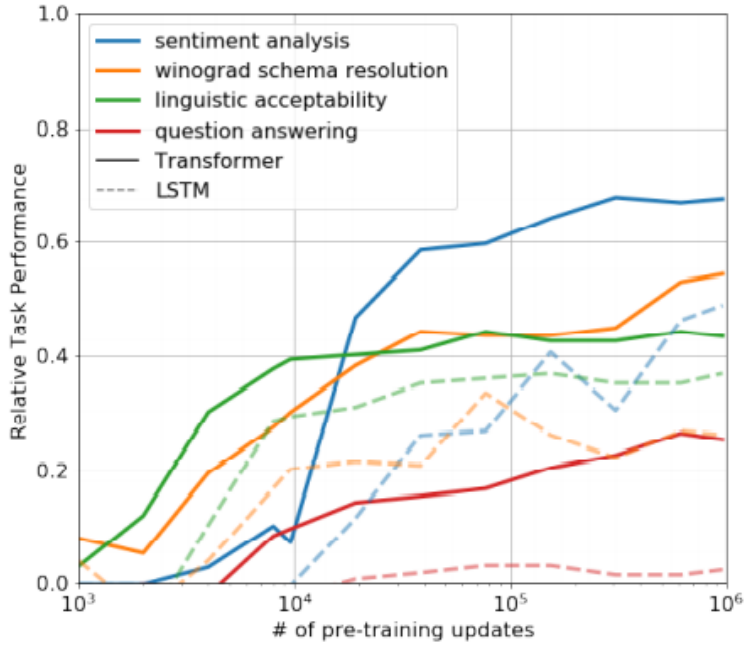
- Augmented objective function in finetuning
- More layers is better!
- Zero-shot

We additionally found that including language modeling as an auxiliary objective to the fine-tuning helped learning by (a) improving generalization of the supervised model, and (b) accelerating convergence. This is in line with prior work [50, 43], who also observed improved performance with such an auxiliary objective. Specifically, we optimize the following objective (with weight λ):

$$L_3(C) = L_2(C) + \lambda * L_1(C) \quad (5)$$

Overall, the only extra parameters we require during fine-tuning are W_y, and embeddings for delimiter tokens (described below in Section 3.3).





Improving Language Understanding by Generative Pre-Training – Radford et al


GPT 2

GPT-2

- Current paradigm => “narrow learners”
  - Don’t generalize well to out-of-distribution data
  - Hypothesis: Single task training
- Idea: Use LM and zero-shot => “general learners”
- + Make your models huge ☺

- P(output|input) → P(output|input, task)
  - (translate to french, english text, french text)
  - (answer the question, document, question, answer)

Language Models are Unsupervised Multitask Learners – Radford et al

WebText

- Common Crawl: big but low-quality
  - Don’t’ use
- WebText:
  - Outbound links from Reddit (with karma >= 3)
  - 45 million links
  - 40 GB of text
  - (Removed Wikipedia to avoid conflicts with other datasets)

Language Models are Unsupervised Multitask Learners – Radford et al

Zero-shot Language Modelling

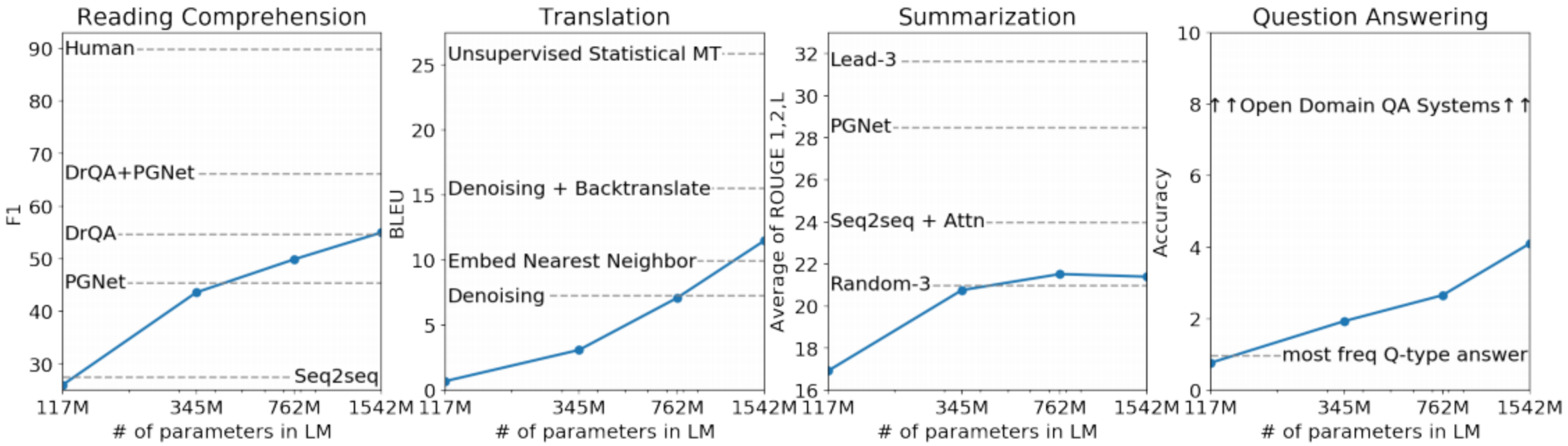
Language Models are Unsupervised Multitask Learners

	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).

Language Models are Unsupervised Multitask Learners – Radford et al

Zero-shot Downstream



- Promising and impressive (compared to expectations)
- But far from SOTA

Language Models are Unsupervised Multitask Learners – Radford et al

Example: Natural Questions

- Top 30 most-confident answers
- Question: did these show up in the training data?

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%

Language Models are Unsupervised Multitask Learners – Radford et al

Generalization vs Memorization

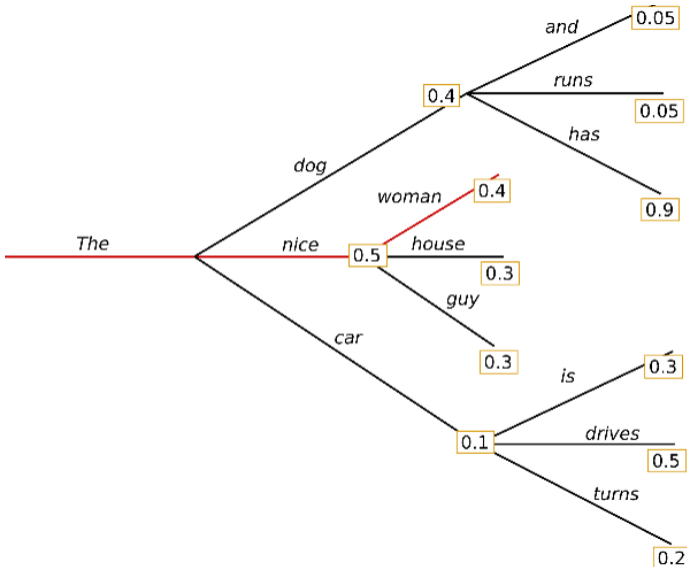
- Bloom filters with 8-grams => estimate overlap
  - Given Datasets A, B.
  - Question: What is the percentage of 8-grams from A that are also in B?
- Interesting: 1BW has overlap of ~13% with its own training set…
- TL;DR – WebText has low or no overlap with the datasets used in the studies

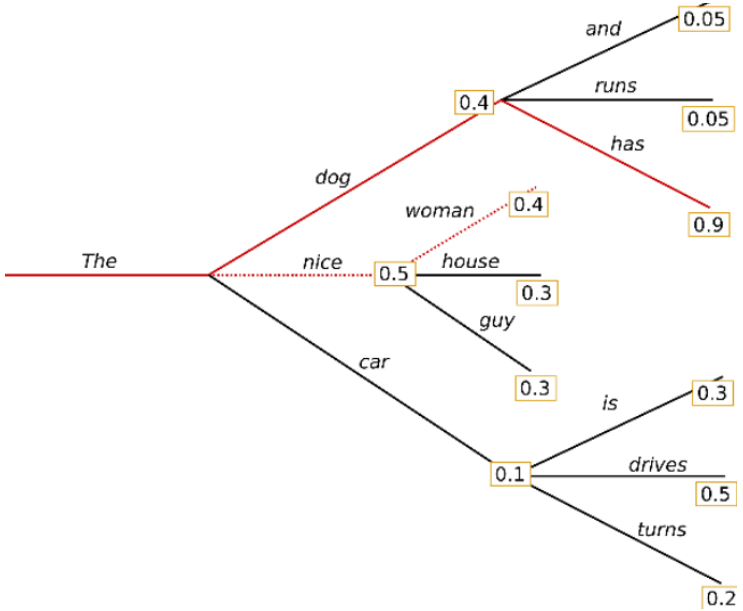
Language Models are Unsupervised Multitask Learners – Radford et al

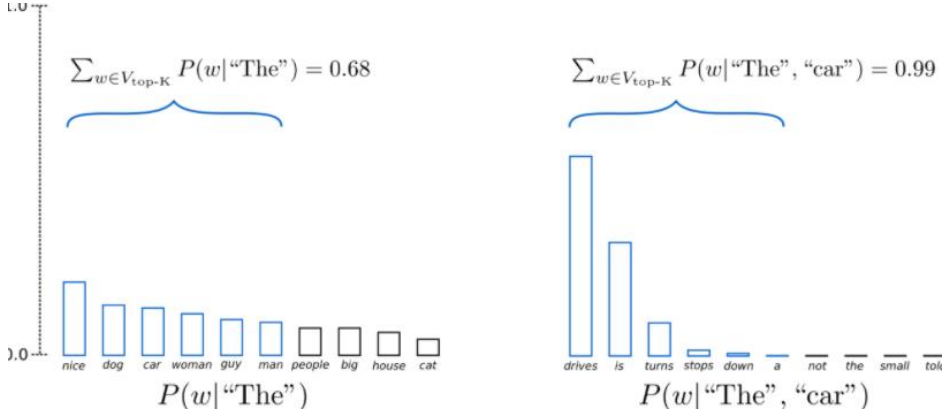
Text generation from LMs

[Huggingface blog / how to generate text]

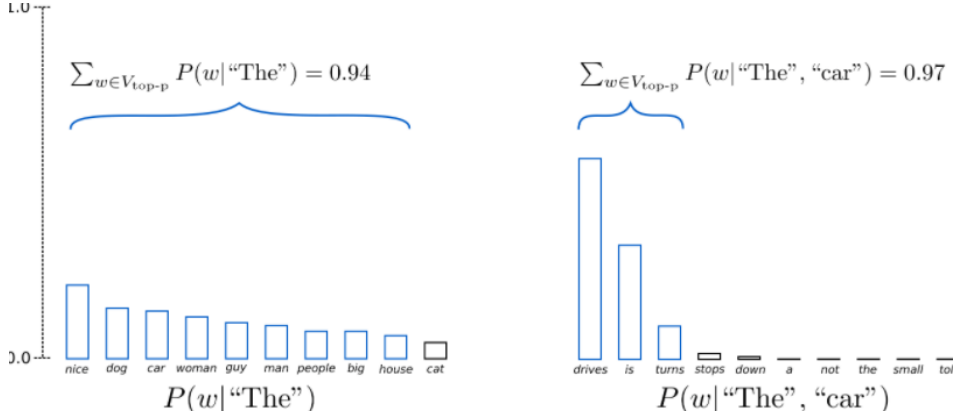
- Greedy
- Beam search
- Sampling
  - Top-k sampling
  - Top-p sampling







Top-k



Top-p

[1]

Context (human-written): 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.

GPT-2: 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.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them – they were so close they could touch their horns.

While examining these bizarre creatures the scientists discovered that the creatures also spoke some fairly regular English. Pérez stated, "We can see, for example, that they have a common 'language,' something like a dialect or dialectic."

Dr. Pérez believes that the unicorns may have originated in Argentina, where the animals were believed to be descendants of a lost race of people who lived there before the arrival of humans in those parts of South America.

While their origins are still unclear, some believe that perhaps the creatures were created when a human and a unicorn met each other in a time before human civilization. According to Pérez, "In South America, such incidents seem to be quite common."

However, Pérez also pointed out that it is likely that the only way of knowing for sure if unicorns are indeed the descendants of a lost alien race is through DNA. "But they seem to be able to communicate in English quite well, which I believe is a sign of evolution, or at least a change in social organization," said the scientist.

Language Models are Unsupervised Multitask Learners – Radford et al