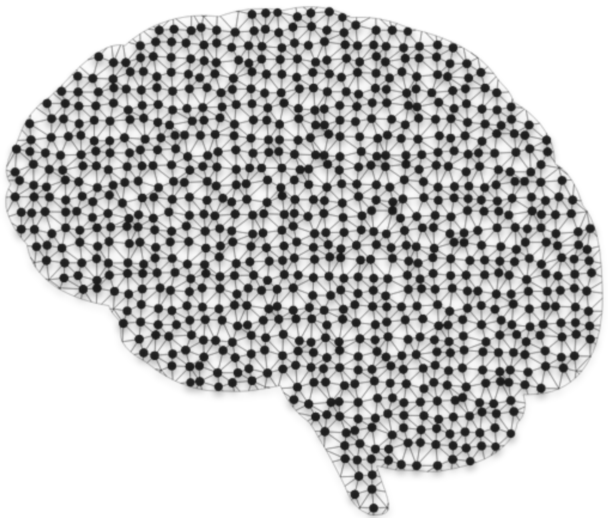


RoBERTa

A Robustly Optimized BERT Pretraining Approach

Table of Contents

- Introduction
- Data & Evaluation
- Implementation & Training
- Results Analysis
- Related Work
- Conclusion

1

Introduction

Original BERT and Motivation

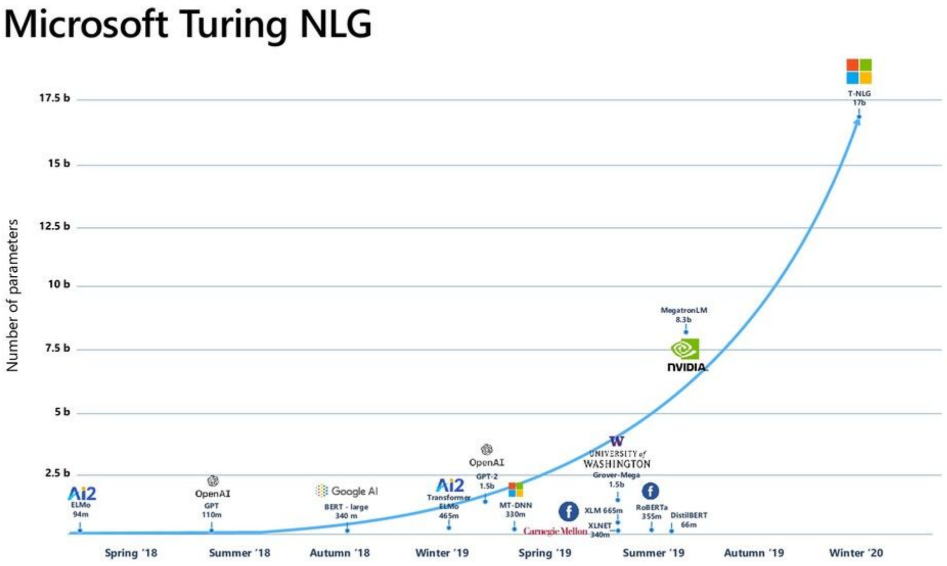


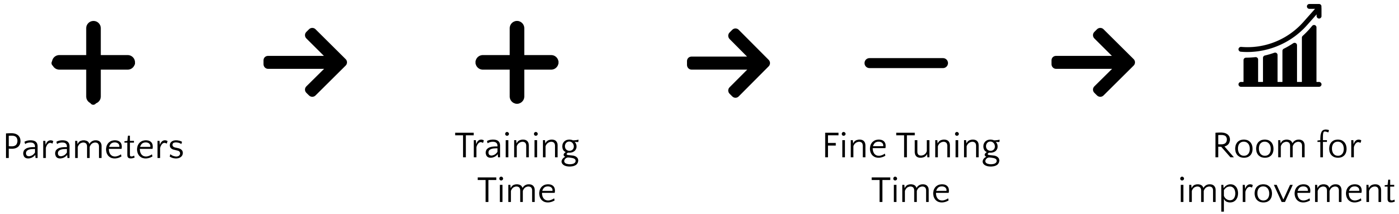
Fig. 1: Development of Number of Parameters in NLP Transformer models with time. [Source]

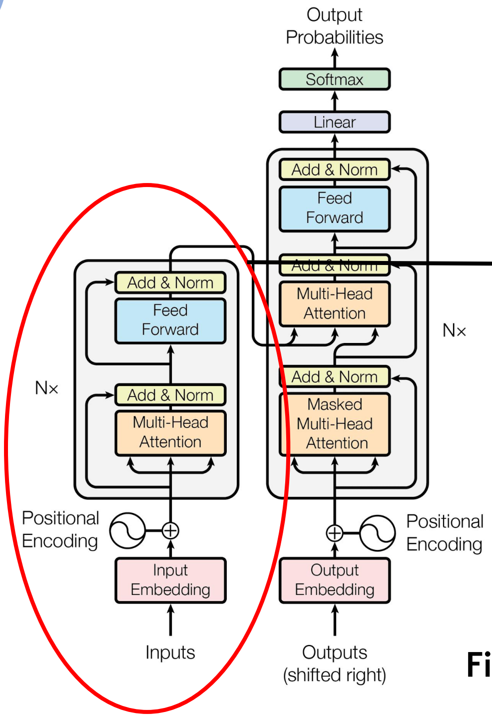
Transformer based models were a **game changer.**

However, they **require** huge training datasets and **billions of parameters**.

Motivation

The **effort required for (pre) training** leads to less fine tuning and **less time for experimenting overall**, which leads to **suboptimal models**.





Bidirectional

Encoder

Representations from

Transformers

Fig. 2: Development of Number of Parameters in NLP Transformer models with time. Source

BERT In Sum

Book Corpus: 11,000 unpublished books scraped from the Internet (~800M words)

English Wikipedia (~2500M words)

Autoencoding model

[base(large)]:

12(24) layers

12(16) attention heads

Optimization: ADAM optimizer

Input: 2 segments (> 1 natural sentence)

Tasks: **NSP** (Next Sentence Prediction); **MLM** Masked Language Model

RoBERTa’s goal

The researchers behind RoBERTa hypothesize that **BERT was severely undertrained** and propose an **improved recipe for training BERT** models

More Intense Training

No NSP Task + Different Approach for MLM

Longer Sequences + Different Encoding

2

Data & Evaluation

Datasets and benchmarks used

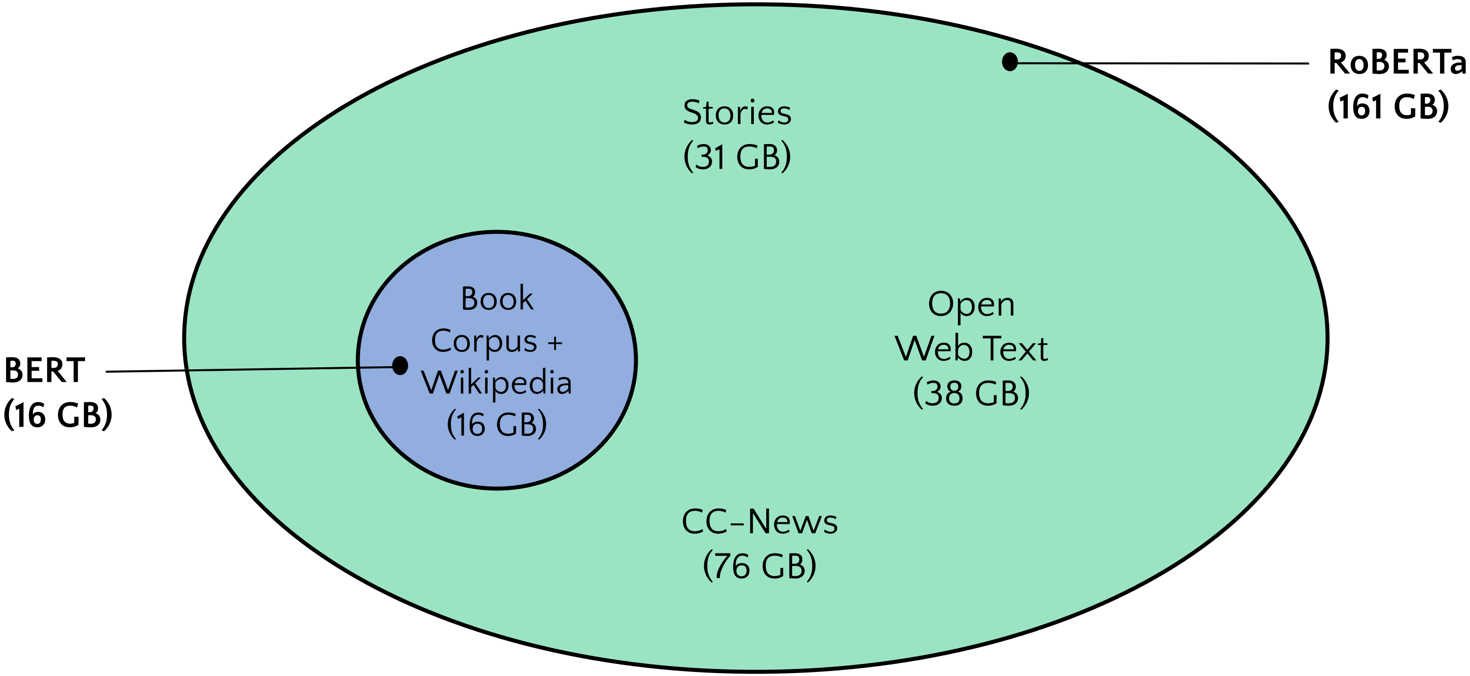
RoBERTa Datasets: BERT + ...

- **Stories:** CommonCrawl data filtered to match the story-like style of Winograd schemas – *Trinh and Le, 2018*

- **Open Web Text:** web content extracted from URLs shared on Reddit with at least three upvotes – *Gokaslan and Cohen, 2019*

- **CC-News:** collected from the English portion of the CommonCrawl Newsdataset; 63 million English news articles crawled between 2016 and 2019 – *Nagel, 2016*

Comparison



RoBERTa
(161 GB)
Stories
(31 GB)
Book
Corpus +
Wikipedia
(16 GB)
Open
Web Text
(38 GB)
BERT
(16 GB)
CC-News
(76 GB)

Evaluation Benchmarks

GLUE

General Language Understanding Evaluation

Origin: 9 datasets:

CoLA, SST-2, MNLI, QNLI, WNLI, RTE, MRPC, QQP, STS-B

Single and sentence-pair classification;

SQuAD

Stanford Question Answering Dataset

Origin: questions posed by crowdworkers on a set of Wikipedia articles

SQuAD 1.1: 100k answerable questions

SQuAD 2.0: +50k unanswerable questions

RACE

Reading and Comprehension from Examinations

Origin: English exams for middle and high school Chinese students

28,000 passages;

~ 100,000 questions

3

Implementation and Training

Preprocessing, training method, implementation, etc.

Implementation & Hyperparameters

- Reimplemented BERT in Fairseq

- Peak Learning Rate and No. of Warmup Sets – vary for each setting

- Found training to be very sensitive to epsilon term

Training Details

Unlike BERT:

- pretrain with sequences < T = 512 tokens
- do **not randomly inject short sequences**
- do **not train with a reduced sequence length** for the first 90% of updates. Train only with full-length sequences.

Trained on DGX-1 machines, each with
8 × 32GB Nvidia V100 GPUs

Masked Language Model

I would love to go to the cinema with you. – Training

Masking

I would love to go to the cinema you. – Testing

Prediction

I would love to go to the cinema with you.

generated text

Masking - BERT vs RoBERTa

BERT: masking once during data preprocessing – single static mask. To **avoid using the same mask** for each training instance **in every epoch**, training **data was duplicated 10 times** –> each training sequence was seen with the same mask 4 times during training.

RoBERTa: dynamic masking – generate 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: Dynamic vs Static Masking. Source

Dynamic Masking: better effectiveness (better results shown in table 1) and better efficiency (faster) -> Researchers' choice

Next Sentence Prediction

Sentence 1. Sentence 2. – Training

Sentence 1. – Testing

Prediction

Sentence 1. Sentence 2.

generated text

Hypothesis: the NSP task might **not** be **useful** and **may** actually **hinder performance,** (Lample & Conneau, 2019; Yang et.al, 2019)

Input Text Sequences and NSP

4 Different Setups Experimented: **With NSP**

SEGMENT– PAIR + NSP: pair of segments, each contain multiple natural sentences, total combined length < 512 token – <u>original</u>

SENTENCE – PAIR + NSP: pair of natural sentences, contiguous portion of one document or separate documents; inputs significantly shorter -> increase the batch size -> tot. no. tokens remains similar

Input Text Sequences and NSP

4 Different Setups Experimented: **Without NSP**

FULL – SENTENCES: full sentences sampled contiguously from one or more documents, total length < 512 tokens.

DOC – SENTENCES: similar to FULL – SENTENCES, except that they may not cross document boundaries; when shorter than 512 -> dynamically increase the batch size

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: Comparison of Input Sequences. Pretrained over BOOK CORPUS and WIKIPEDIA. F1 for SQuAD and accuracy for MNLI-m, SST-2 and RACE. Source

Input and NSP Decision

NSP Strategies were outperformed:

DOC-SENTENCES: Best Result but needs variable batch sizes

FULL-SENTENCES: Almost as good and fixed batch size -> Researchers' choice

Batches

Hypothesis: training with very large mini-batches can both improve optimization speed and end-task performance when the learning rate is increased appropriately *(Ott et al., 2018) (You et al., 2019)*

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 BERT base models trained over BOOK CORPUS and WIKIPEDIA with varying batch sizes (bsz). Source

Large batches:

- improved perplexity for MLM and end-task accuracy
- easier to parallelize via distributed data parallel training
- Decision: train with batches of 8K sequences.

Byte-Pair Encoding

Middle range between

- **using characters** – long sequences and less meaningful tokens
- **using words** – very large vocabulary size

Byte-Pair Encoding

In contrary to the name, classical BPE uses Unicode Characters as the base of the vocabulary.

Base BERT: Classical BPE

RoBERTa: follows technique introduced in *Radford et al. (2019)* – uses bytes instead of Unicode Characters

Byte-Pair Encoding

Unicode characters → min. Vocab. Size = all Unicode Characters (~150,000)

Bytes → min. Vocab size = 2ˆ8 = 256

4 Results Analysis

Training data and benchmarks' results

Training Data

Impact of adding training data:

- RoBERTa shows massive improvement when compared to BERT-Large which reaffirms the importance of the design choices.

Training data quality:

- Removing low quality examples and pre training the models for longer led to improvements in performance.

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 pretrained over more and more data. Source

Benchmarks Results - GLUE

- On the first case it performs better that the other models, achieving state-of-the-art results on all 9 of the GLUE task development sets
- On the other side, even though RoBERTa did not obtain the best results for each of the 9 tasks , it scored the highest average result to the date when compared to the other models (88.5%)

	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	**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 for each of the 9 tasks for single-task single models and ensembles. [Source]

Benchmarks Results - SQuAD

- On the first version RoBERTa matched the state-of-art set by XLNet. On the other hand in the second version, RoBERTa achieved a new state-of-the-art improving over XLNet by 0.4 points (EM) and 0.6 points (F1).

- RoBERTa model outperforms all but one of the single model submissions, and is the top scoring system among those that do not rely on data augmentation

Model	SQuAD 1.1 EM	SQuAD 1.1 F1	SQuAD 2.0 EM	SQuAD 2.0 F1
Single models on dev, w/o data augmentation				
$\text{BERT}_{\text{LARGE}}$	84.1	90.9	79.0	81.8
$\text{XLNet}_{\text{LARGE}}$	**89.0**	94.5	86.1	88.8
RoBERTa	88.9	**94.6**	**86.5**	**89.4**
Single models on test (as of July 25, 2019)				
$\text{XLNet}_{\text{LARGE}}$			86.3†	89.1†
RoBERTa			86.8	89.8
XLNet + SG-Net Verifier			**87.0**†	**89.9**†

Table 6: Results on SQuAD for single models for the both versions of SQuAD. [Source]

Benchmarks Results - RACE

When it comes to the RACE benchmark RoBERTa achieved the state-of-the-art results on both the Middle and High School settings

Model	Accuracy	Middle	High
Single models on test (as of July 25, 2019)			
$\text{BERT}_{\text{LARGE}}$	72.0	76.6	70.1
$\text{XLNet}_{\text{LARGE}}$	81.7	85.4	80.2
RoBERTa	**83.2**	**86.5**	**81.3**

Table 7: Results on RACE for single models for both Middle and High School settings. Source

5 Related Work

How much have new models improved until now?

ALBERT

The ALBERT method focuses heavily on the parameter efficiency. Some of the introduced changes:

- **Cross-Layer Parameter Sharing**: Layers can share parameters between them, opposing the previous paradigma in which each of these had its own set of parameters.

- **Sentence-Order Prediction**: Pre-training task to predict permuted sentences order. Improves the model ability to find reasoning between multiple sentences.

ALBERT

Comparison to BERT and RoBERTa:

Published in 2020, achieved SOTA results and is generally better than RoBERTa.

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 10: 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.

Glue Leaderboard Today

Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP
1	Microsoft Alexander v-team	Turing ULR v6		91.3	73.3	97.5	94.2/92.3	93.5/93.1	76.4/90.9
2	JDExplore d-team	Vega v1		91.3	73.8	97.9	94.5/92.6	93.5/93.1	76.7/91.1
3	Microsoft Alexander v-team	Turing NLR v5		91.2	72.6	97.6	93.8/91.7	93.7/93.3	76.4/91.1
4	DIRL Team	DeBERTa + CLEVER		91.1	74.7	97.6	93.3/91.1	93.4/93.1	76.5/91.0
5	ERNIE Team - Baidu	ERNIE		91.1	75.5	97.8	93.9/91.8	93.0/92.6	75.2/90.9
6	AliceMind & DIRL	StructBERT + CLEVER		91.0	75.3	97.7	93.9/91.9	93.5/93.1	75.6/90.8
7	DeBERTa Team - Microsoft	DeBERTa / TuringNLRv4		90.8	71.5	97.5	94.0/92.0	92.9/92.6	76.2/90.8
8	HFL iFLYTEK	MacALBERT + DKM		90.7	74.8	97.0	94.5/92.6	92.8/92.6	74.7/90.6
9	PING-AN Omni-Sinitic	ALBERT + DAAF + NAS		90.6	73.5	97.2	94.0/92.0	93.0/92.4	76.1/91.0
10	T5 Team - Google	T5		90.3	71.6	97.5	92.8/90.4	93.1/92.8	75.1/90.6
21	Facebook AI	RoBERTa		88.1	67.8	96.7	92.3/89.8	92.2/91.9	74.3/90.2

Table 8: Current Glue leaderboard. Information taken from: Source

6 Conclusion

Conclusion

The key changes RoBERTa introduced are:

- Bigger batches, with more data.
- Removing NSP
- Longer sequences training
- Dynamically masking.

Conclusion

- The paper **lacks** novelty in model architecture and technical contributions,
- However, RoBERTa's proposal to change the training setup (hyperparameters, data size, etc) definitely **improves** the performance when comparing the results to previous alternatives.
- It may benefit **future search** on the topic.

Thanks!

Any *questions* ?

Processamento de Linguagem Natural – FEUP, M.EIC 22/23 – Group A

- Marcelo Henriques Couto – up201906086@up.pt
- André Lino dos Santos – up201907879@up.pt
- João Afonso Andrade – up201905589@up.pt

References

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In Proceedings of the IEEE international conference on computer vision (pp. 19-27).

Sebastian Nagel. 2016. Cc-news.

References

Aaron Gokaslan and Vanya Cohen. 2019. Openweb-text corpus.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. arXiv preprint arXiv:1806.02847.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In International Conference on Learning Representations (ICLR).

References

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In Empirical Methods in Natural Language Processing (EMNLP).

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. arXiv preprint arXiv:1704.04683.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291.

References

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. arXiv preprint arXiv:1906.08237.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

References

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations

Pengcheng He, Xiaodong Liu, Jianfeng Gao, Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with Disentangled Attention

Barun Patra, Saksham Singhal, Shaohan Huang, Zewen Chi, Li Dong, Furu Wei, Vishrav Chaudhary, Xia Song. 2022. Beyond English-Centric Bitexts for Better Multilingual Language Representation Learning.