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

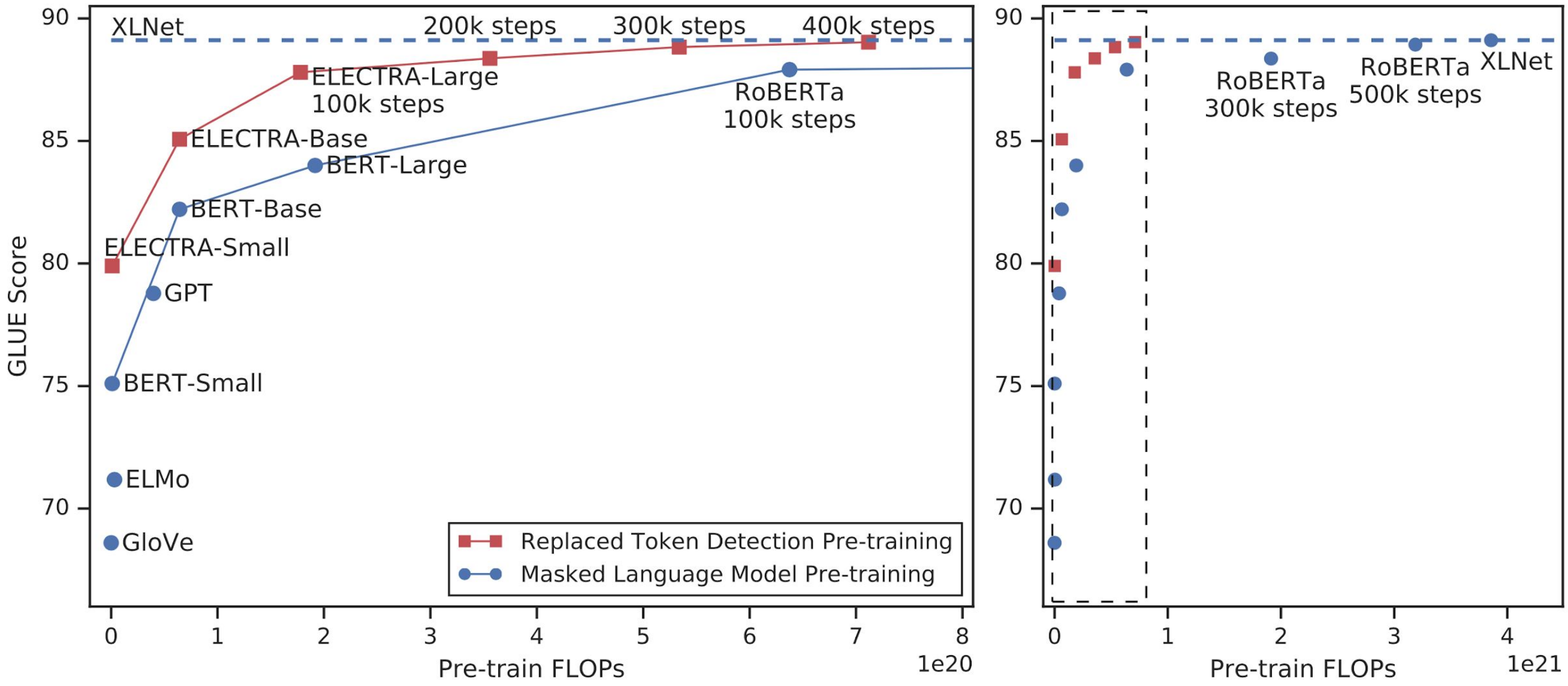
Kevin Clark
Stanford University
kevclark@cs.stanford.edu

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

Fakanov Pavel, AMI 171
HSE Research Seminar
November 2020


90
85
80
75
70
GLUE Score
XLNet
200k steps
300k steps
400k steps
ELECTRA-Large
100k steps
RoBERTa
100k steps
ELECTRA-Base
BERT-Large
BERT-Base
ELECTRA-Small
GPT
BERT-Small
ELMo
GloVe
Replaced Token Detection Pre-training
Masked Language Model Pre-training
0
1
2
3
4
5
6
7
8
Pre-train FLOPs
1e20
90
85
80
75
70
RoBERTa
300k steps
RoBERTa
500k steps
XLNet
0
1
2
3
4
Pre-train FLOPs
1e21

Overview

- Replaced Token Detection
- Model architecture
- Training
- Experiment results
- Efficiency Analysis

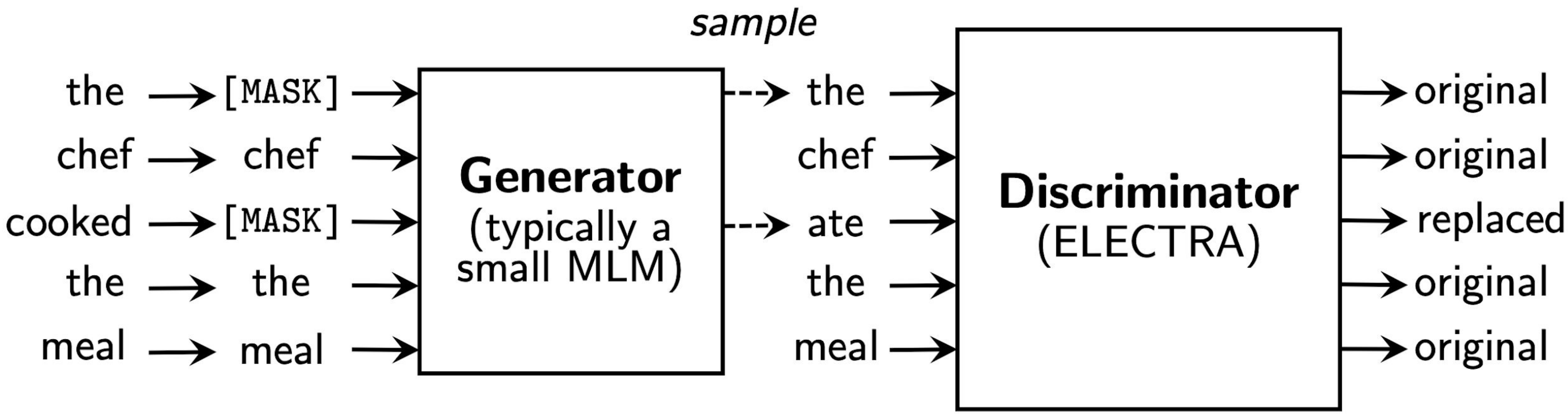
Replaced Token Detection

Problem with Masked Language Modeling

Predicts only 15% of the tokens

Solution: Predicting all inputs

Replaced Token Detection



Replaced Token Detection

- Learns from all input tokens (instead of 15%)
- More parameter-efficient
- More compute-efficient
- Improves downstream task performance

Model Architecture

Generator

It outputs a probability for a particular token x_t

$$p_G(x_t|\boldsymbol{x}) = \mathbf{exp}\left(e(x_t)^T h_G(\boldsymbol{x})_t\right) / \sum_{x'} \exp\left(e(x')^T h_G(\boldsymbol{x})_t\right)$$

Discriminator

Given a position t, it predicts whether the token x_t is real

$$D(\boldsymbol{x}, t) = \mathrm{sigmoid}(w^T h_D(\boldsymbol{x})_t)$$

Training

Steps

- MLM selects a random set of positions to mask out m = [m1, m2, …, mk]
- The generator predicts original words of the masked out tokens
- The discriminator distinguishes tokens replaced by the generator

Combined Loss

$$\min_{\theta_G, \theta_D} \sum_{\boldsymbol{x} \in \mathcal{X}} \mathcal{L}_{\text{MLM}}(\boldsymbol{x}, \theta_G) + \lambda \mathcal{L}_{\text{Disc}}(\boldsymbol{x}, \theta_D)$$

Difference from GANs

- If the generator generates the original token, is is considered real
- The generator is trained with MLM
- The generator is not trained to fool the discriminator
- We use discriminator on downstream tasks
- No noise vector

Experiments

Datasets

- GLUE (General Language Understanding Evaluation)
- (sentiment, textual similarity, entailment)
- Metrics on 9 tasks, the result is the average

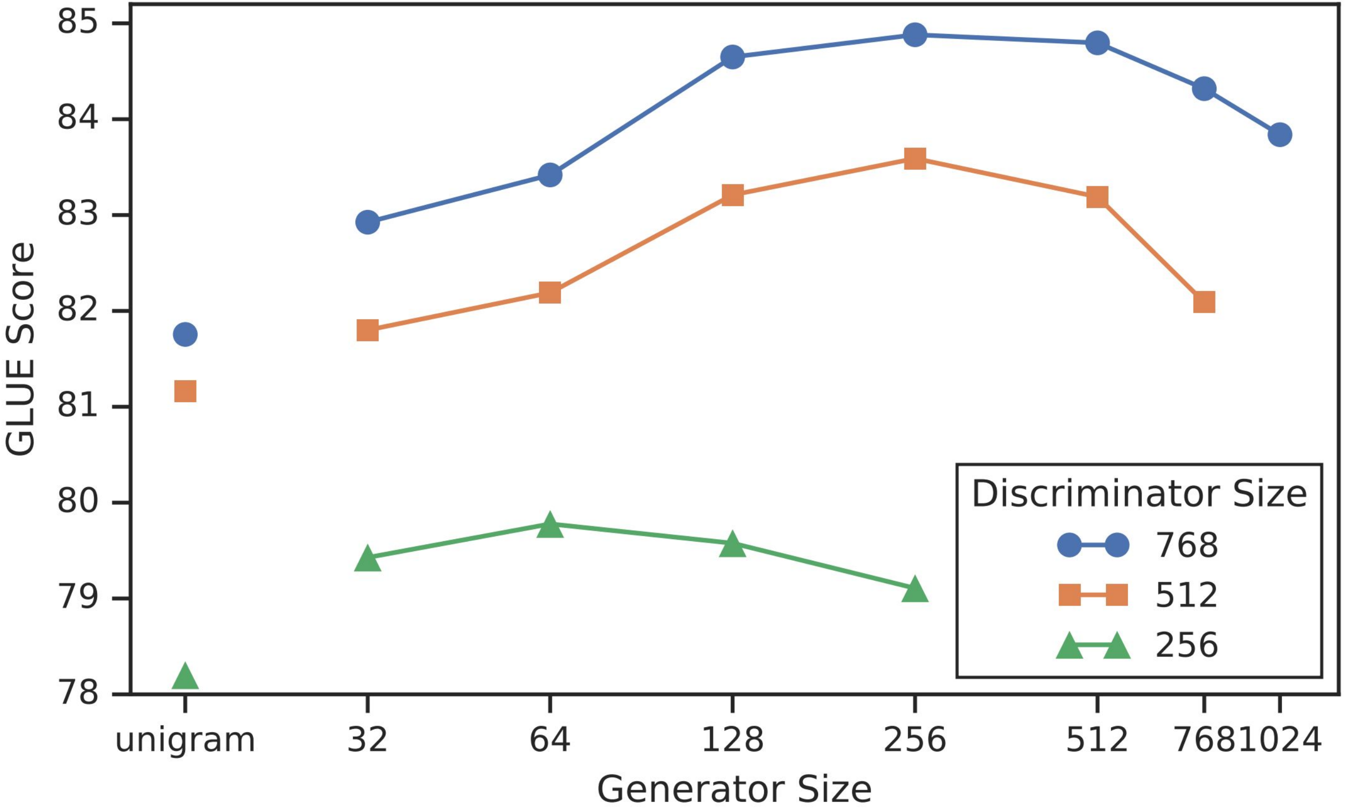
Datasets

- SQuAD (Stanford Question Answering Dataset)
- Question Answering
- Exact-Match and F1 scores

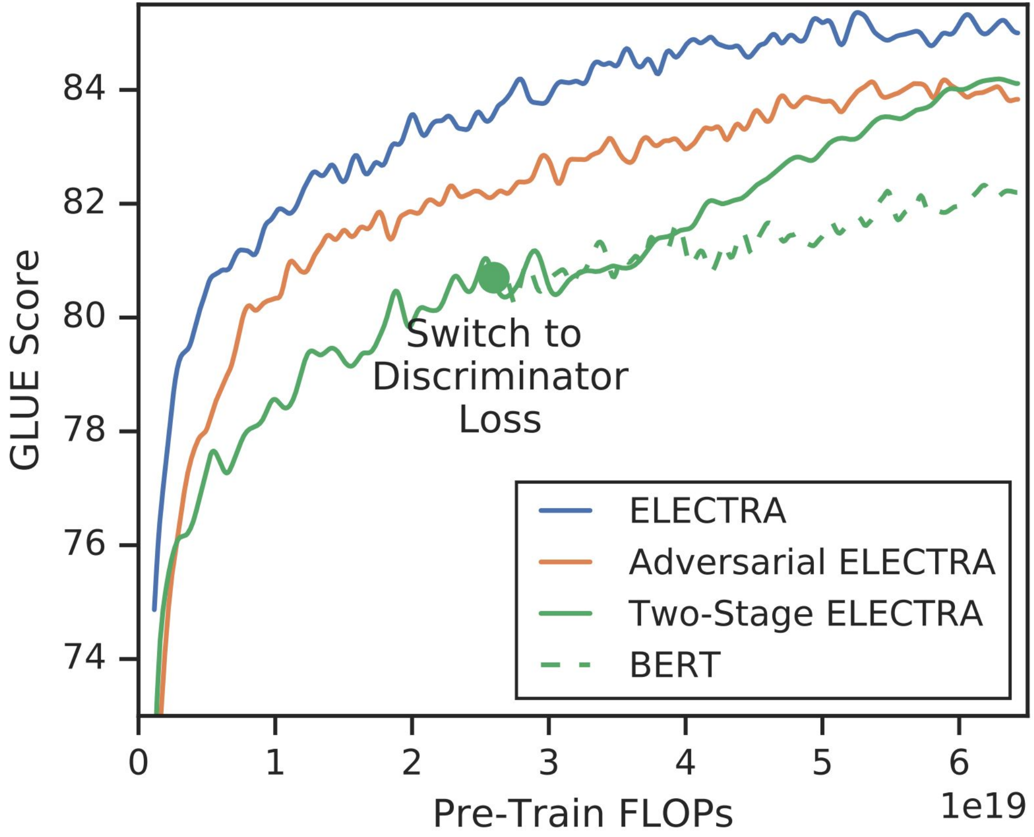
Weight sharing

- No weight sharing: **83.6**
- Embedding weight sharing: **84.3**
- All weight sharing: **84.4** (needs to type the model sizes)

Which generator size works best?



Training algorithms



ELECTRA Small Compared to BERT

- Sequence length: **512 -> 128**
- Word embedding size: **768 -> 128**
- Hidden dimension size: **768 -> 256**

ELECTRA Small Compared to BERT

Model	Train / Infer FLOPs	Speedup	Params	Train Time + Hardware	GLUE
ELMo	3.3e18 / 2.6e10	19x / 1.2x	96M	14d on 3 GTX 1080 GPUs	71.2
GPT	4.0e19 / 3.0e10	1.6x / 0.97x	117M	25d on 8 P6000 GPUs	78.8
BERT-Small	1.4e18 / 3.7e9	45x / 8x	14M	4d on 1 V100 GPU	75.1
BERT-Base	6.4e19 / 2.9e10	1x / 1x	110M	4d on 16 TPUv3s	82.2
ELECTRA-Small	1.4e18 / 3.7e9	45x / 8x	14M	4d on 1 V100 GPU	79.9
50% trained	7.1e17 / 3.7e9	90x / 8x	14M	2d on 1 V100 GPU	79.0
25% trained	3.6e17 / 3.7e9	181x / 8x	14M	1d on 1 V100 GPU	77.7
12.5% trained	1.8e17 / 3.7e9	361x / 8x	14M	12h on 1 V100 GPU	76.0
6.25% trained	8.9e16 / 3.7e9	722x / 8x	14M	6h on 1 V100 GPU	74.1
ELECTRA-Base	6.4e19 / 2.9e10	1x / 1x	110M	4d on 16 TPUv3s	85.1

ELECTRA Large

- The same size as BERT-Large
- ELECTRA-400k: ¼ the pre-training compute of RoBERTa
- ELECTRA-1.75m: similar compute to RoBERTa

GLUE Dev Set

Model	Train FLOPs	Params	CoLA	SST	MRPC	STS	QQP	MNLI	QNLI	RTE	Avg.
BERT	1.9e20 (0.27x)	335M	60.6	93.2	88.0	90.0	91.3	86.6	92.3	70.4	84.0
RoBERTa-100K	6.4e20 (0.90x)	356M	66.1	95.6	**91.4**	92.2	92.0	89.3	94.0	82.7	87.9
RoBERTa-500K	3.2e21 (4.5x)	356M	68.0	96.4	90.9	92.1	92.2	90.2	94.7	86.6	88.9
XLNet	3.9e21 (5.4x)	360M	69.0	**97.0**	90.8	92.2	92.3	90.8	94.9	85.9	89.1
BERT (ours)	7.1e20 (1x)	335M	67.0	95.9	89.1	91.2	91.5	89.6	93.5	79.5	87.2
ELECTRA-400K	7.1e20 (1x)	335M	**69.3**	96.0	90.6	92.1	**92.4**	90.5	94.5	86.8	89.0
ELECTRA-1.75M	3.1e21 (4.4x)	335M	69.1	96.9	90.8	**92.6**	**92.4**	**90.9**	**95.0**	**88.0**	**89.5**

GLUE Test Set

Model	Train FLOPs	CoLA	SST	MRPC	STS	QQP	MNLI	QNLI	RTE	WNLI	Avg.*	Score
BERT	1.9e20 (0.06x)	60.5	94.9	85.4	86.5	89.3	86.7	92.7	70.1	65.1	79.8	80.5
RoBERTa	3.2e21 (1.02x)	67.8	96.7	89.8	91.9	90.2	90.8	95.4	88.2	89.0	88.1	88.1
ALBERT	3.1e22 (10x)	69.1	**97.1**	**91.2**	92.0	90.5	**91.3**	–	89.2	91.8	89.0	–
XLNet	3.9e21 (1.26x)	70.2	**97.1**	90.5	**92.6**	90.4	90.9	–	88.5	**92.5**	89.1	–
ELECTRA	3.1e21 (1x)	**71.7**	**97.1**	90.7	92.5	**90.8**	**91.3**	**95.8**	**89.8**	**92.5**	**89.5**	**89.4**

SQUAD

Model	Train FLOPs	Params	SQuAD 1.1 dev		SQuAD 2.0 dev		SQuAD 2.0 test	
			EM	F1	EM	F1	EM	F1
BERT-Base	6.4e19 (0.09x)	110M	80.8	88.5	–	–	–	–
BERT	1.9e20 (0.27x)	335M	84.1	90.9	79.0	81.8	80.0	83.0
SpanBERT	7.1e20 (1x)	335M	88.8	94.6	85.7	88.7	85.7	88.7
XLNet-Base	6.6e19 (0.09x)	117M	81.3	–	78.5	–	–	–
XLNet	3.9e21 (5.4x)	360M	**89.7**	**95.1**	87.9	**90.6**	87.9	90.7
RoBERTa-100K	6.4e20 (0.90x)	356M	–	94.0	–	87.7	–	–
RoBERTa-500K	3.2e21 (4.5x)	356M	88.9	94.6	86.5	89.4	86.8	89.8
ALBERT	3.1e22 (44x)	235M	89.3	94.8	87.4	90.2	88.1	90.9
BERT (ours)	7.1e20 (1x)	335M	88.0	93.7	84.7	87.5	–	–
ELECTRA-Base	6.4e19 (0.09x)	110M	84.5	90.8	80.5	83.3	–	–
ELECTRA-400K	7.1e20 (1x)	335M	88.7	94.2	86.9	89.6	–	–
ELECTRA-1.75M	3.1e21 (4.4x)	335M	**89.7**	94.9	**88.0**	**90.6**	**88.7**	**91.4**

Efficiency Analysis

Pre-training objectives

- ELECTRA 15%: Loss only from the 15% of the tokens that are masked
- Replace MLM: Used generated tokens for masked tokens instead of [MASK]

  To solve discrepancy between pre-training and fine-tuning
- All-Tokens MLM: Predicts the replaced tokens and other tokens

  Models tend to copy inputs for for non-masked tokens

Pre-training objectives

Input: The chef cooked the meal

Replace MLM: [The] chef [ate] the meal

All-Tokens MLM: [The] chef [ate] the meal

Pre-training objectives

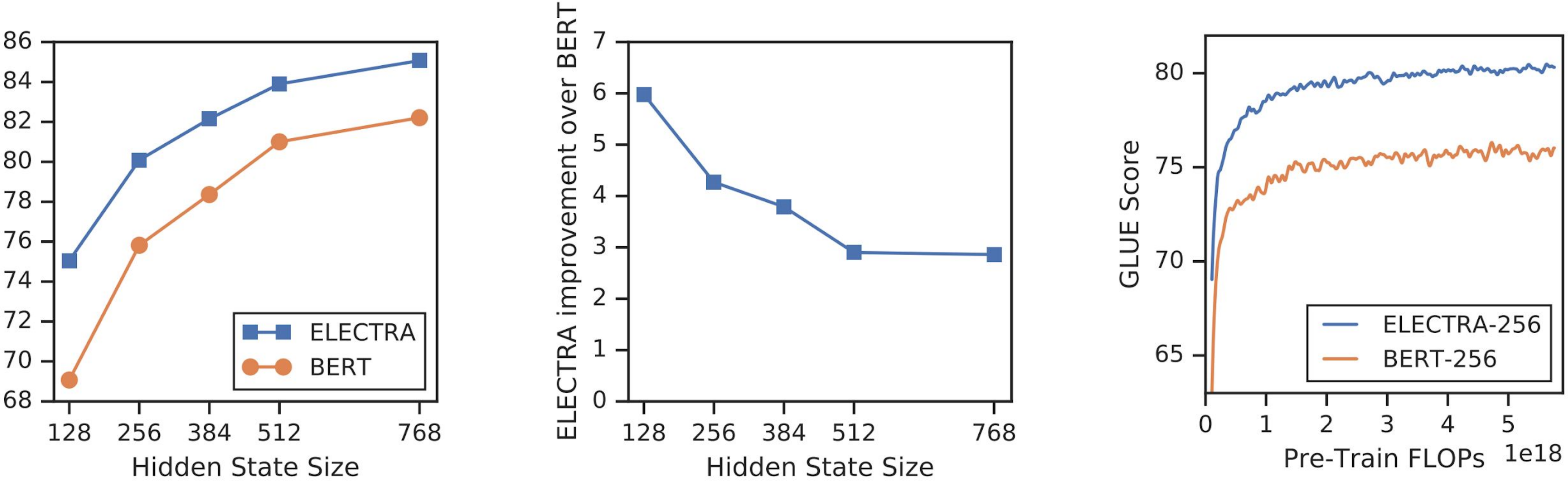
Model	ELECTRA	All-Tokens MLM	Replace MLM	ELECTRA 15%	BERT
GLUE score	85.0	84.3	82.4	82.4	82.2

Pre-training objectives

Model	ELECTRA	All-Tokens MLM	Replace MLM	ELECTRA 15%	BERT
GLUE score	85.0	84.3	82.4	82.4	82.2

- Loss over all inputs is key (most of improvement is from here)
- Removing the pre-train fine-tune mismatch is not that helpful

Gains vs. Model sizes



Conclusion

Summary

- Loss over all inputs is key
- Discriminator predicts predicts original/replacement tokens
- Better compute and parameter efficiency

Questions

- Describe replaced token detection pre-training task
- What loss is used in ELECTRA model?
- Describe another training objective authors experimented with (1 of 3)

References

- Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. ELECTRA: Pre-training text encoders as discriminators rather than generators. In International Conference on Learning Representations, 2020.

- Deep Learning Explainer