Using the Transformer

ELECTRA (Clark et al., 2019)

Learning goals

- Replaced Token Detection task
- Interplay of Generator and Discriminator

A DIFFERENT PRE-TRAINING REGIME

ELECTRA ▸ Clark et al. (2020)

- ELECTRA consinsts of two separate models

→ (Small) generator model G + (large) Discriminator model D

→ This might resemble a GAN setup, but they are not trained in an adversarial manner

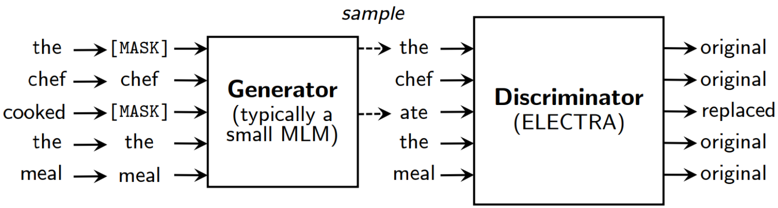
- Generator task: Masked language modeling
- Discriminator task: *Replaced token detection*

→ Predict for each token, whether it is "original" or produced by G

- ELECTRA learns from *all* of the tokens (not just from a small portion of 15%, like e.g. BERT)

ELECTRA VISUALIZED

Joint pre-training:



Source: Clark et al. (2020)

- *G* and *D* are (Transformer) encoders which are trained jointly
- *G* replaces `[MASK]`s in an input sequence
  $\rightarrow$ Passes corrupted input sequence $\vec{x}^{corrupt}$ to *D*

TRAINING DETAILS

Joint pre-training:

- Generation of samples:

$$m_i \sim \text{unif}\{1, n\} \text{ for } i = 1 \text{ to } k \qquad \boldsymbol{x}^{\text{masked}} = \text{REPLACE}(\boldsymbol{x}, \boldsymbol{m}, \texttt{[MASK]})$$
$$\hat{x}_i \sim p_G(x_i | \boldsymbol{x}^{\text{masked}}) \text{ for } i \in \boldsymbol{m} \qquad \boldsymbol{x}^{\text{corrupt}} = \text{REPLACE}(\boldsymbol{x}, \boldsymbol{m}, \hat{\boldsymbol{x}})$$

  with approx. 15% of the tokens masked out (via choice of k)

- D predicts whether $x_t,\ t \in 1, \ldots, T$ is "*real*" or generated by G
  - Softmax output layer for G (probability distr. over all words)
  - Sigmoid output layer for D (Binary classification real vs. generated)

TRAINING DETAILS

Using the masked & corrupted input sequences, the (joint) loss can be written down as follows:

Loss functions:

$$\mathcal{L}_{\text{MLM}}(\boldsymbol{x}, \theta_G) = \mathbb{E}\left(\sum_{i \in \boldsymbol{m}} -\log p_G(x_i | \boldsymbol{x}^{\text{masked}})\right)$$

$$\mathcal{L}_{\text{Disc}}(\boldsymbol{x}, \theta_D) = \mathbb{E}\left(\sum_{t=1}^{n} -\mathbb{1}(x_t^{\text{corrupt}} = x_t) \log D(\boldsymbol{x}^{\text{corrupt}}, t) - \mathbb{1}(x_t^{\text{corrupt}} \neq x_t) \log(1 - D(\boldsymbol{x}^{\text{corrupt}}, t))\right)$$

Combined:

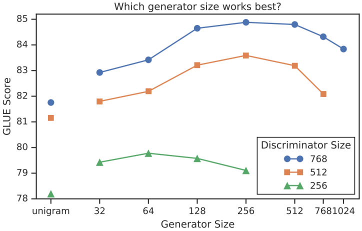
$$\min_{\theta_G, \theta_D} \sum_{\boldsymbol{x} \in \mathcal{X}} \mathcal{L}_{\text{MLM}}(\boldsymbol{x}, \theta_G) + \lambda \mathcal{L}_{\text{Disc}}(\boldsymbol{x}, \theta_D)$$

with λ set to 50, since the discriminator's loss is typically much lower than the geneator's.

TRAINING DETAILS

Generator size:

- Same size of G and D:
  - Twice the compute per training step + too challenging for D
- Smaller Generators are preferable (1/4 – 1/2 the size of D)

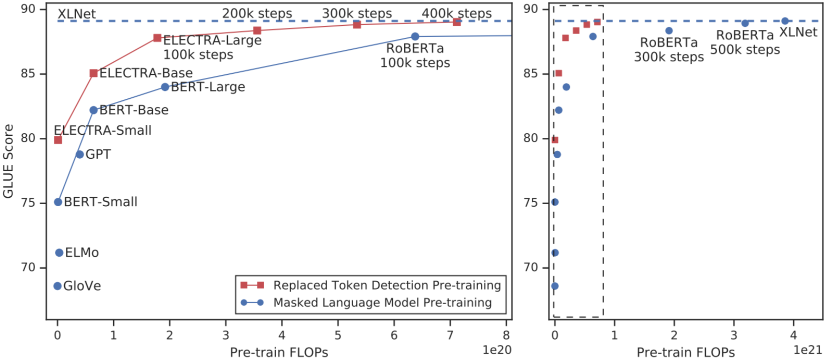


Source: Clark et al. (2020)

Weight sharing (experimental):

- Same size of G and D: All weights can be tied
- G smaller than D: Share token & positional embeddings

MODEL COMPARISON



Source: Clark et al. (2020)

Note: Different batch sizes (2k vs. 8k) for ELECTRA vs. RoBERTa/XLNet explain why same number of steps lead to approx. 1/4 of the compute for ELECTRA.

SOTA PERFORMANCE

Performance differences vs. BERT/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**

Source: Clark et al. (2020)

SOTA performance (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**

* Avg. excluding QNLI to ensure comparability

Source: Clark et al. (2020)