ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators

1 Contribution

The work has proposed a novel pre-training method for transformer network. Unlike its predecessors which uses generative pre-training, the work has used discriminative training to solve the following two drawbacks of masked language model:

1. Since in masked language model only a small subset (15%) of the input tokens are masked and then the network is trained to identify only those tokens, the network learns from only 15% of the input tokens. This results into slower convergence during pre-training.

2. In masked language modeling, the inputs are corrupted by an artificial mask token which does not present on the fine-tuning data. This causes pre-training – fine-tuning discrepancy.

The paper proposed a novel pre-training task called *replaced token detection* to solve the above two problems. In *replaced token detection* task, the input is corrupted by replacing some tokens with samples from a proposal distribution, which is typically the output of a small masked language model. The pre-training task is to learn to distinguish real input tokens from the corrupted tokens. The illustration of the *replaced token detection* task is shown in Figure 1. After pre-training, only the discriminator is used for fine-tuning on downstream tasks.

2 The Pre-training

In *replaced token detection* task, both the generator and discriminator are simultaneously trained during pre-training. The generator is trained using masked language modeling objective and the discriminator is trained using the two-class classification objective.

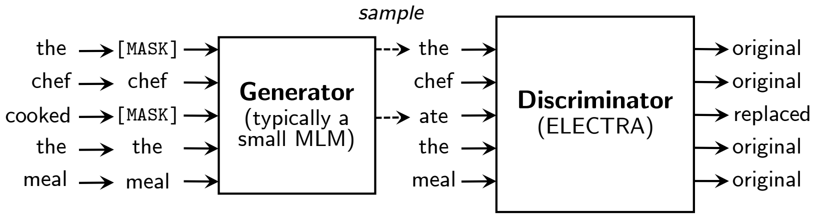

Figure 2: An overview of replaced token detection. The generator can be any model that produces an output distribution over tokens, but we usually use a small masked language model that is trained jointly with the discriminator. Although the models are structured like in a GAN, we train the generator with maximum likelihood rather than adversarially due to the difficulty of applying GANs to text. After pre-training, we throw out the generator and only fine-tune the discriminator (the ELECTRA model) on downstream tasks.

Figure 1: An illustration of the *replaced token detection* task.

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

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

and $\mathcal{L}_{\text{pre-training}}(\mathbf{x};\theta_G,\theta_D) = \mathcal{L}_{\text{MLM}}(\mathbf{x};\theta_G) + \lambda\mathcal{L}_{\text{Disc}}(\mathbf{x};\theta_D)$

3 The Model

In ELECTRA, both the generator and discriminator architecture are same as BERT. During experiments, size of the generator model has been tuned for better results. The best results are obtained with generators 1/4-1/2 the size of the discriminator. After training, the generator is thrown away and the discriminator is kept as the pre-trained model.

4 Results

Empirically, the paper has shown that ELECTRA pre-training converges much faster than that of BERT (please refer to Figure 2) since the task is defined over all input tokens rather than just the small subset which is masked out. As a result, ELECTRA substantially outperforms BERT given the same model size, data, and compute. The gains are particularly strong for small models. The larger model also 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.

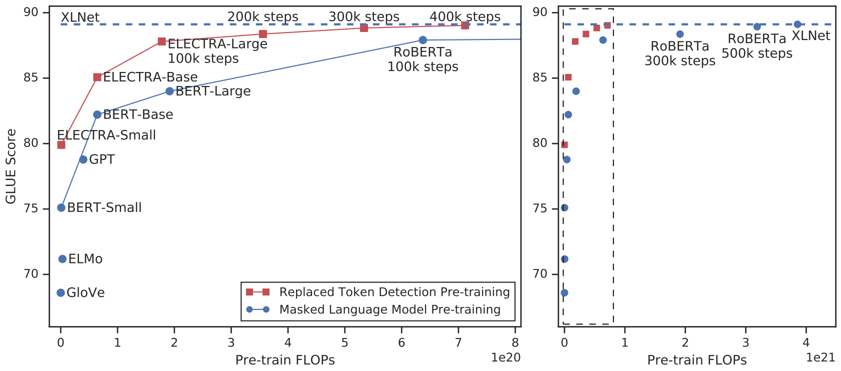

Figure 1: Replaced token detection pre-training consistently outperforms masked language model pre-training given the same compute budget. The left figure is a zoomed-in view of the dashed box.

Figure 2: Comparison of the rate of convergence of ELECTRA and BERT