Paper: BadEncoder: Backdoor Attacks to Pre-trained Encoders in Self-Supervised Learning
Venue: SP 2022

1. Summary
   The paper proposes BadEncoder, a backdoor attack method for self-supervised learning. The attack pollutes the pretrained model trained from a large number of unlabeled data with a set of *shadow dataset* so that downstream classifiers built with the pretrained model will have backdoors embedded in them. The authors formulated two loss functions, effectiveness loss (backdoor attack works) and utility loss (stealthiness), and formulate an optimization problem to solve for the pretrained model that jointly minimizes both loss functions.
2. Pros
   a. The topic is very timely. Large pretrained models are receiving a lot of attention these days.
   b. Embedding multiple backdoors for different downstream classifiers was surprising.
   c. The proposed attack was also tested against existing defenses.
3. Cons
   a. The method seems to be just formulating loss functions, combining them, and solving it with gradient descent. The attack method is not novel.
   b. High success rate achieved when shadow dataset size is larger than 20% of the pre-training dataset. That's A LOT of data.
4. Meaning of the paper
   a. The paper demonstrates that backdoors attack can be executed on models trained with self-supervised learning.
   b. The fact that such attacks are possible is important as many services nowadays use pre-trained models.
   c. This is an attack paper. The described attack is very critical as pre-trained models are widely adopted.
   d. The paper seems to be the first paper to implement a backdoor attack on self-supervised models.
5. Discussions and Questions
   a. There seems to be no penalty for attacking multiple downstream tasks simultaneously. Actually, attack success rate increased for some datasets. How is this possible?
   b. I wonder whether the attack will be successful for more complex tasks. It might have been easier to attack the tasks described in the paper because they are relatively simple and do not utilize all the features extracted from the encoder.