

**USENIX SECURITY 2018 —芝加哥大学**

**Code:** <https://github.com/bolunwang/translearn>

**一、Abstract**

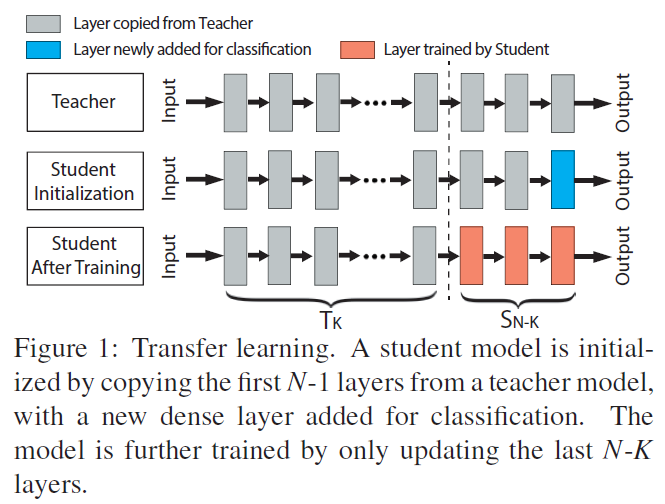
本文在迁移学习（Transfer Learning，即Student模型从Teacher模型继承模型架构和参数）下，对Student模型进行误分类攻击（Misclassification Attack，既为对抗样本攻击）。攻击者可以利用白盒（White-Box）的Teacher模型，生成对抗样本，对黑盒（Black-Box）的Student模型进行攻击。最后，文章提出防御措施，来混淆Teacher模型和Student模型的联系，抵抗误分类攻击。

**二、Introduction**

与之前的黑盒对抗样本攻击相比，本文不需要迭代地查询Student模型。本文贡献总结为：

1. 在迁移学习中提出了实际的误分类攻击
2. 开发识别技术，从多个Teacher模型中，识别给定Student模型所对应的Teacher模型。
3. 提出防御方案，包括改变训练过程、改变需要预测的数据以及引入集成模型

**三、Background**



迁移学习（Transfer Learning），如图所示，是Student模型从Teacher模型中继承前N-1层（也可能更少层）的模型架构和参数，然后再添加最后一层。之后训练最后几层的参数，例如只训练最后一层，或是训练倒数N-K层。

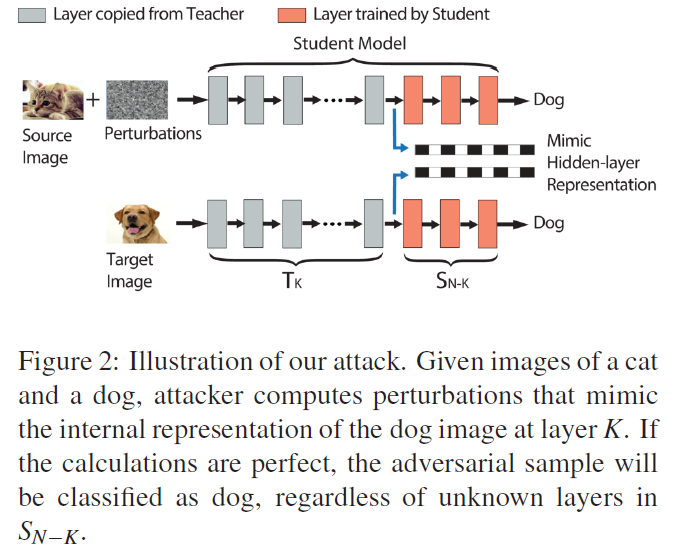
优点是：可以使用更少的训练数据和训练时资源，同时具有很好的预测精确度。

可以这样做的原因是：神经网络前面N-1层进行的是特征提取，相似的任务可以共用特征提取模型。最后一层是跟特定任务相关的分类。因此，可以只修改并训练最后一层模型参数和架构实现Student模型。

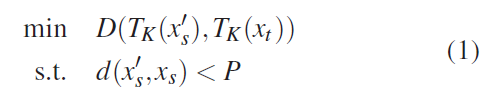
对抗样本攻击（Adversarial Sample Attacks），是让模型对一个加上少量噪音的样本，进行错误的预测。模型对原始样本可以进行正确预测，并且引入噪音的样本不容易被察觉。

分类：分为白盒攻击（攻击者了解目标模型的架构和参数细节）和黑盒攻击（攻击者不了解模型细节，但是可以对目标模型进行查询，既给定一个样本，模型返回预测结果）。白盒攻击几乎可以实现100%的攻击成功率，黑盒攻击成功率较低，同时也需要迭代的模型查询。

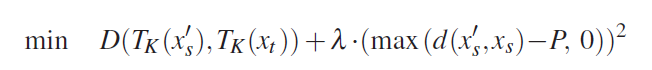
**四、Method**



方案思想是：如图所示，由于Student模型前K层参数继承自Teacher模型，不被训练，Source图像加上扰动后，在第K层的特征表示只要与Target图像的K层特征表示相同，加噪音的Source将会被预测为Target类别，导致错误预测。

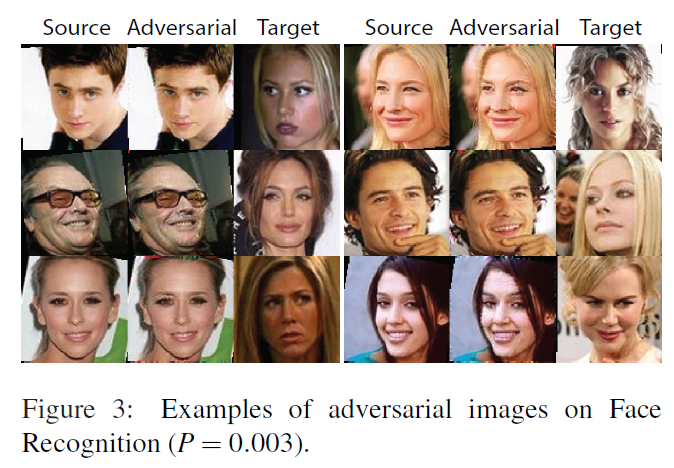


构造对抗样本需要解决上述优化问题，要让对抗样本在第K层的特征尽可能接近不同的目标类的特征。表示第K层的特征表示，表示Source样本，表示target样本，D表示范数，衡量中间特征之间的差异。S.t. 表示优化函数的限制条件，P表示一个预定值（budget），d表示衡量对抗样本和原始样本之间的差异，文中使用DSSIM，差异不能超过P。

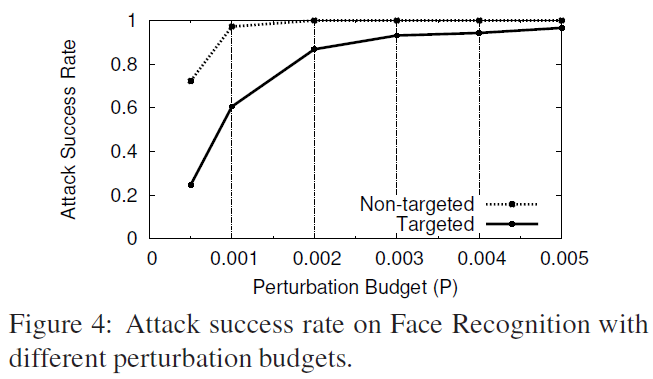


因为解决有限制条件优化问题困难，所以利用penalty method（惩罚方法）进行求解，既是将条件转移到优化函数中。如果限制条件没有满足，则优化函数不能达到最小值。

**五、Experiment**



左侧第一列为原始图像，中间为生成的对抗样本，被分类为第三列图像的类别，同时肉眼察觉不到。



不同的扰动程度对攻击成功率的影响。Budget P增加即可更大程度添加噪音，因此攻击成功率更高。

**六、Attack for real ML services**

现实ML场景中，用户会从多个Teacher模型中选择一个模型来学习Student模型，因此攻击者需要识别用户选择了哪一个模型。

识别方法：

假设Student模型只是训练最后一层，攻击者可以查询Student模型。首先针对不同Teacher模型伪造特定的对抗样本，使得倒数第二层特征值为0。因此



上述公式为最后一层神经网络运算，为权重，为偏置项。如果Student模型用与Teacher模型配对的对抗样本进行查询，则模型预测结果为。作者定义了Gini coefficient（详见论文），根据系数值的大小确定Student模型选择的Teacher模型。

**七、防御方法**

Dropout Randomization：即预测之前随机去掉部分输入像素。

Injecting Neuron Distances：训练全部模型，增大特征表示的举例。

Ensemble Models：即训练多个Student模型，预测时根据多个预测结果进行判断。