H-012

# 認識モデルクローン手法の一般化と評価
# A General Method for Cloning of Recognition Model and its Evaluation

伊藤 千紘† 安藤 申将† 工藤 航† 酒造 正樹† 前田 英作†
Chihiro Ito Nobumasa Ando Wataru Kudo Masaki Shuzo Eisaku Maeda

## 1. はじめに

近年、Amazon Machine Learning、Google Cloud Platform、Azure Machine Learning Studio など機械学習を利用して画像等の認識を行うサービス（MLaaS）が広く公開されている。サービス内部の識別器はブラックボックス化されているものの、公開 API に対する大量の入出力結果を得て学習すれば、識別モデルクローンが作成可能であることを示唆されている [1]。このモデル抽出攻撃（model extraction attack）の防御策を講じる上で、攻撃者側の視点に立ち、現状の技術でどの程度クローンを作成できるかを検討することには意義があり、近年様々な報告がされている [2-4]。

モデル抽出攻撃は、サンプリング工程と学習工程に大別される。前者において、ターゲット識別器に入力された特徴量と出力されたラベルに基づき、ターゲット識別器に入力すべき適切な特徴量を選択する。後者では、上記で得た特徴量とラベルを使って学習を行い、クローン識別器を生成する。本論文においてはサンプリング工程を中心に議論を行うこととする。

一般に、サンプリング回数が増加するほど学習に使えるデータが増えるため、より正確なサービス内部識別器のクローンが可能になる。しかし、モデル抽出攻撃を行う際には、サンプリングを無限回行うことは現実的でなく、なるべく少ない回数でより正確に識別器をクローンすることが重要となる。

本論文の目的は、識別器のクローン可能性を検証することである。識別器のクローンを独立に複数行いその差分を用いることで、より少ないサンプリング回数で効率的にモデル抽出攻撃を行う手法を提案し評価した。

## 2. 課題設定

サービス提供者と攻撃者がいると仮定する。まず、サービス提供者はターゲットとなる識別器を作成し、ターゲット識別器への入出力インタフェースを攻撃者に提供する。ここで、攻撃者はターゲット識別器の作成に用いられたオリジナルデータの性質（入力データの定義域およびクラスラベルの集合など）を理解するものの、データ自体を知ることができない。

次に、試行回数 $n$ が与えられ、攻撃者はターゲット識別器に特徴量を入力し、識別結果のラベルの取得を繰り返す。次いで、攻撃者は取得した特徴量とラベルを使って学習を行い、クローン識別器を得る。本論文において、両識別器に同じ入力を行い、その出力が一致しているほどクローンがうまくいっているものと見做す。計算機性能に依存する処理時間については言及しないが、なるべく少ないサンプリング回数でクローン識別器を作成するものとする。

† 東京電機大学 Tokyo Denki University

## 3. ターゲット認識器に入力する特徴量選択方法

### 3.1 方針

よい入力特徴量の特徴（図 1）として、第一に入力特徴量同士の特徴空間における距離が離れていることである。第二に入力特徴量がターゲット識別器の識別境界付近であることが挙げられる。

後者に関して、本論文ではターゲット識別器が持つ特徴空間における特徴量の識別境界らしさを表す値 LDB（likelihood of decision boundary）を提案する。これは、異なる 2 種の認識器の出力がどの程度異なっているかを表した値であり、識別境界に近いデータほど出力が一致しにくいという知見に基づく。識別すべきラベル数 $l$ と $m$ ラベルの差の大きさ $d_m$ を用いて、以下の式に定義する。

$$LDB = \sum_{m=1}^{l} d_m / l$$

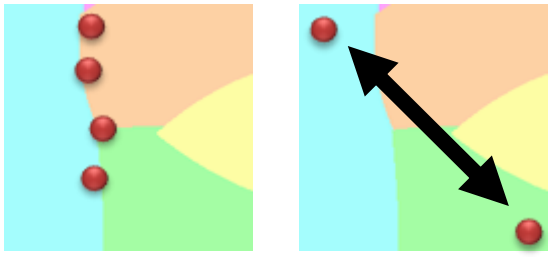
ここで、シングルラベル（$m = 1$）に対するターゲット識別器の場合は、識別結果の差は一致か不一致のいずれかであるため、LDB は 0 または 1 を取る。一方、マルチラベル（$m \geqq 2$）である場合は、0〜1 の実数値を取る。選択すべき特徴量は、候補の中から LDB が最大のものとなる。

### 3.2 サンプリングアルゴリズム

本論文で提案するサンプリングアルゴリズムを以下に示す。定義域内に入力特徴量の候補となる有限集合が存在するものとする。初期値として、1 回目の入力特徴量はランダムに決定し、ターゲット識別器の出力結果ラベルを得る。

次に、$k$ ( $\geqq 2$) 回目の入力特徴量を決定する際は、まず $k-1$ 回目までに取得済みの入力特徴量及びラベルのデータを使って、事前に用意した 2 種の比較用識別器を独立に学習させておく。続いて、学習済みの比較用識別器に、選別された入力特徴量候補の集合をそれぞれ識別させ、その結果を得る。LDB を算出し、最大となるもの以外を除外する。

続いて、$k-1$ 回目までに決定済みの特徴量と近距離にあるものを除外し、ある閾値（例えば平均距離など）以上離れたものを複数選択する。この中から 1 つをランダムに選択し、これを $k$ 回目のターゲット識別器への入力特徴量と決定する。以上を $n$ まで繰り返す（図 2）。



識別境界付近　　大きな入力データ間距離

図 1　効果的な入力特徴量

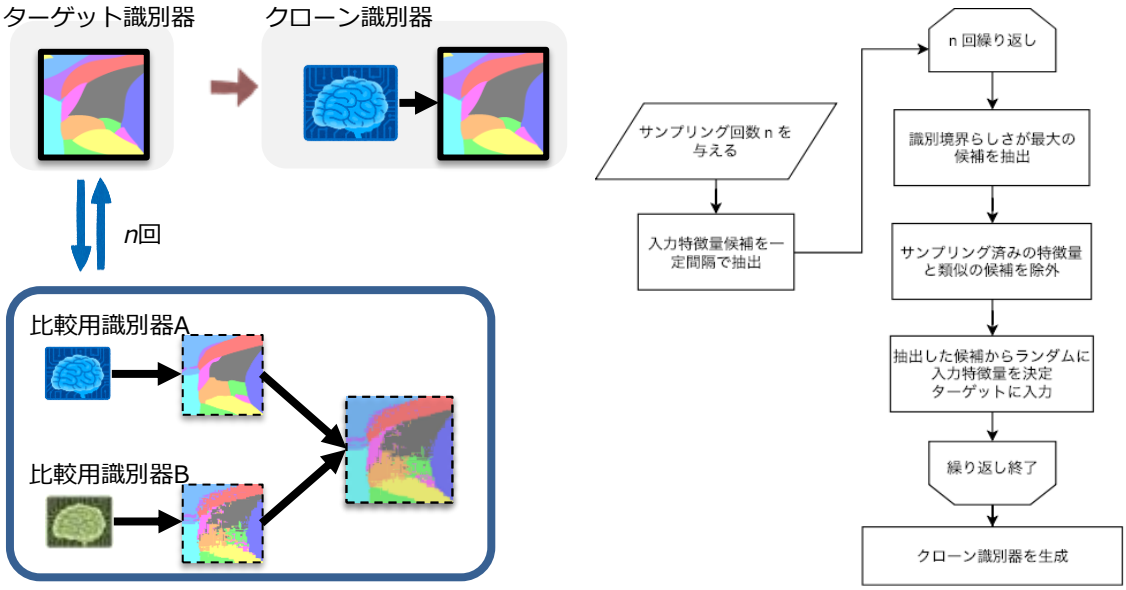



図 2　サンプリングアルゴリズム

表 1　使用したデータセット

| データセット | クラス数 | サンプル数 | 特徴次元 |
|---|---|---|---|
| Iris[1] | 3 | 150 | 4 |
| Wine[2] | 3 | 178 | 13 |
| Breast cancer[3] | 2 | 569 | 30 |
| Digits[4] | 10 | 1797 | 64 |

1 http://archive.ics.uci.edu/ml/datasets/Iris
2 http://archive.ics.uci.edu/ml/datasets/Wine
3 http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
4 http://archive.ics.uci.edu/ml/datasets/optical+recognition+of +handwritten+digits

表 2　ターゲット識別器に対する一致率

| データセット | サンプリング工程 | |
|---|---|---|
| | ランダム | 提案手法 |
| Iris | 0.912 | 0.947 |
| Wine | 0.965 | 0.997 |
| Breast cancer | 0.968 | 0.992 |
| Digits | 0.404 | 0.461 |

## 4. クローン識別器の性能評価

### 4.1 評価方法

上述サンプリング手法によるクローン識別器の性能を、ランダムサンプリングによるものと比較して行う。ランダムサンプリングは、与えられたサンプリング回数 $n$ に対して、入力候補の集合の中からランダムに $n$ 個を選択する特徴量選択方法である。両サンプリング手法により得たクローン識別器とターゲット識別器の結果の一致率を評価する。

### 4.2 評価実験

データセットとして、識別問題において一般的用いられる iris、wine、breast cancer、digits の 4 種（表 1）を用いた。全てシングルラベルが与えられている。これらのデータセットを 2 分割し、ターゲット識別器の訓練用と入力特徴量の候補用とした。

ターゲット識別器の学習アルゴリズムは、線形 SVM（support vector machine）とした。一方、クローン識別器の学習アルゴリズムは k 近傍法とし、サンプリング工程のみの性能を比較するため、サンプリング回数を 15 に固定した。サンプリング工程における比較用識別器には、決定木と非線形 SVM を用いた。

10 回交差検定による識別一致率の結果を表 2 に示す。データセットの次元によらず、全てにおいてランダムサンプリングより提案手法によるサンプリングの方が高い一致率を示していることがわかる。

## 5. おわりに

以上の結果から、少数の入力特徴量とターゲット識別器から取得したラベルを用いて 2 種の比較用識別器を学習し、その差分を用いることで、効果的なサンプリングができることを示した。本論文における評価はシングルラベルに対応するターゲット識別器を扱ったが、マルチラベルに対応する識別器にも有効であり、今後の検証項目とする。

2 種の比較用識別器に関して、1 つのサンプリングを行う度に学習を行う必要があるため、機械学習モデルとしては軽量なものを用いることが望ましい。本論文においては、出力差異が大きい手法として決定木と非線形 SVM を比較用識別器に用いた。その他の比較用識別器を用いた場合の検証も今後の課題とする。

### 参考文献

[1] F. Tramer *et al.*, “Stealing Machine Learning Models via Prediction APIs,” in Proc. 25th USENIX Security Symposium, pp. 601–618, 2016.
[2] S. Pal *et al.*, “A Framework for the Extraction of Deep Neural Networks by Leveraging Public Data,” arXiv preprint, arXiv:1905.09165, 2019.
[3] I. Unceta *et al.*, “Copying Machine Learning Classifiers,” arXiv preprint, arXiv:1903.01879, 2019.
[4] X. Hu *et al.*, “Neural Network Model Extraction Attacks in Edge Devices by Hearing Architectural Hints,” arXiv preprint, arXiv:1903.03916, 2019.