4W-01

FaceNetに対するAdversarial Exampleによる意図的誤認識

野間口圭[1], 黒米祐馬[2], 武田圭史[3], 村井純[3]

[1]慶應義塾大学法学部法律学科
[2]慶應義塾大学環境情報学部
[3]慶應義塾大学

概 要

現在, 認証手段として, ディープラーニングを用いた顔認識技術の応用が進んでいる. その実装の例としてFaceNetがあげられる. FaceNetは, 畳み込みニューラルネット (CNN) の手法を使用しており, 認識率はほぼ99%と示しており, 現在の顔認識のstate-of-the-artである. しかし, CNNのような機械学習には, 特定のやり方で加工したAdversarial Exampleによって誤認識を起こすという特有の問題がある. 本稿では, ディープラーニングへの依存によってAdversarial Exampleに対しどの程度脆弱になるか検証する.

1 背景

社会においては, 利用者対してにサービスを提供するために, 利用者本人であることを確認する認証が必要になっている. その一つの手段として顔認識技術が研究されており, 様々な企業が使用している [1].

2006年, 深層学習の基本概念が提唱され, 画像による顔認識に応用が検討され始めた. すでに最近, 富士通がディープラーニングを用いた顔認識システムを開発すると発表した [2].

一方, ディープラーニングを用いた画像認識においては特有の問題が存在する. それは, 悪意を持って特殊に加工した画像を識別する時, 誤認識を起こすというものである. この画像をAdversarial Example[3] という. 画像認識を用いた認証方法では機械学習の技術に依存していけばいくほど, Adversarial Exampleに対して脆弱になっていく可能性がある.

2 目的

本研究では, ディープラーニングを用いた顔認識システムがAdversarial Exampleによってどの程度影響を受けるのか示す. ここでは, 現在顔認識におけるstate-of-the-artである, FaceNet[4] に, 独自に構築したCNNを用いてAdversarial Exampleを生成し, 認識させる.

2.1 FaceNet

FaceNetとは, Googleが開発した顔認識ソフトウェアである. このソフトウェアは, 二つの画像が, 同一人物か否か判定する. FaceNetは, 写真から切り取った画像からConvolutional Neural Net (CNN) を用いて特徴抽出し, 正規化をしたのち, 高次元のユークリッド空間に埋め込む. 判定は, 埋め込んだ双方の距離から判断する.

FaceNetの認識率は, 顔認識専用のデータセットである Labeled Face in the Wild (LFW)[5] を用いて, 99%であった. 現在では顔認識のstate-of-the-artとして, オープンソース化されている. またこれを利用したソフトウェアOpenfaceも公開されるなど, 応用も進んでいる. 本研究では, David Sandberg氏の公開している実装を使用した[1].

2.2 Adversarial Example

Adversarial Exampleとは, 機械学習システムの誤認識を企図して作成された画像である.

機械学習では, 損失関数を使ってCNNのバイアスや重みを更新することで, 結果的に特徴を学習していく. Adversarial Exampleは, この損失関数をなるべく最大化することで生成される. また, FaceNetの損失関数はTriplet Loss関数である. 損失関数を微分した式に画像を計算させて, 符号関数と係数 ϵ で計算した以下の式を, Fast Gradient Sign Methodという. なお, $\tilde{x}$ はAdversarial Example, xは画像, $J'(x)$ は損失関数を示す.

$$\tilde{x} = \epsilon \mathrm{sign}(J'(x))$$

3 手法

本目的を達成させるため, LFWのテストデータと, FaceNetを学習する時に使用するCNNを用いて, Adversarial Examplesを生成する.

そのためにAdversarial Exampleを作成するソフトウェアであるDeep-pwning[2]を拡張する.

このソフトウェアは, 手書き数字のデータセットMNIST, 物体のデータセットCIFAR-10に対応しているが, LFWには対応していない. 本手法ではLFWに対して, 上記に掲げた手法をDeep-pwningを使用して実現する.

4 関連研究

初めてAdversarial Exampleを生成したのは, Szegedyら [3] であり, これに関して理論的説明を与え

[1]davidsandberg/facenet. (閲覧日 2017 年 1 月 13 日). https://github.com/davidsandberg/facenet

[2]cchio/deep-pwning (閲覧日 2017 年 1 月 13 日) .https://github.com/cchio/deep-pwning

図 1: 加工をしていない画像

図 2: **Adversarial Example:** 画像にノイズが存在する.

たのが Goodfellow らである [6]. Papernot らは, 実際のサービス, Google の Cloud Prediction API と, Amazon の Amazon Machine Learnig に Adversarial Example を使用し, 高い誤検知率を示した [7].

本稿では, 実際にオープンソースとして存在する FaceNet に用いて評価する.

5 評価方法

最初に, 1 加工をしていない LFW の画像を, ランダムで二枚選択する. 同じ人だが, 種類が違う画像の組と, 違う人で, 種類が違う画像の組, それぞれ 3000 組ずつ, 計 6000 組選定する. 尚, Fast Gradient Sign Method の係数 ϵ は 0.9 で作成した.

次に, 2 そのときに使用した画像を使用して, Adversarial Example を作成する. これも同じように, 同じ人だが, 種類が違う画像の組と, 違う人で, 種類が違う画像の組, それぞれ 3000 組ずつ, 計 6000 枚ずつ選定する.

1, 2 のときの計 6000 組ずつを FaceNet に認識させて, そのときに出力される Accuracy, Validation Rate を比較する. なお, このとき, 十分割交差平均を用いた.

6 実験

以下のグラフのように, Validation Rate は, 94.389% から, 0.133%と, 急激に低下している.

なお, Validation Rate は, 二つの画像が同一人物である組の内, FaceNet が同一人物と認識できた組の割合である. Validation Rate の場合は, False Alert Rate(二つの画像が違う組の内, FaceNet が同じと判定した組の割合) が 0.103%になるような閾値を設定した.

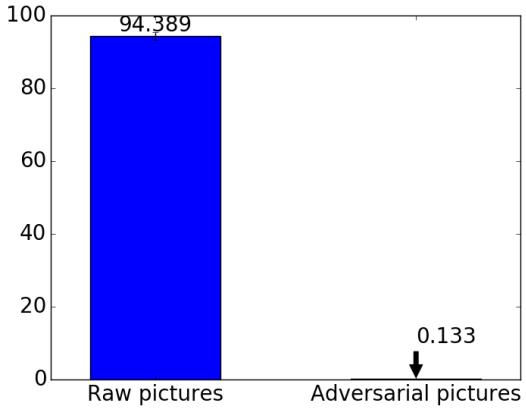

図 3: Validation Rate

7 考察

Validation Rate を算出する上で, 閾値はどちらも同じに設定している. その上で比較すると, 確率が急激に降下していることから, FaceNet が同一人物の組み合わせを, 違うものと誤認識しているということを考えることができる.

8 まとめと今後の課題

以上の実験のように, 閾値を同じに設定した上で, FaceNet において, 機械学習は Adversarial Examples の画像の組を誤認識する. これをもって, 機械学習がディープラーニングに依存すればするほど, Adversarial Example に弱くなることを示すことができた. 今後の課題として, 他の発展的なディープラーニングのシステムが, どの程度 Adversarial Example の影響を受けるのか実験, 検証していきたい.

参考文献

[1] http://jpn.nec.com/rd/research/DataAcquition/face.html

[2] http:/www.atmarkit.co.jp/ait/articles/1602/04/news140.html

[3] C. Szegedy et al. “ Intriguing properties of neural networks “ ICLR 2014.

[4] F. Schroff, et al ”FaceNet: A Unified Embedding for Face Recognition and Clustering” IEEE 2015.

[5] http://vis-www.cs.umass.edu/lfw/

[6] I. Goodfellow, et al. “ Explaining and Harnessing the Adversarial Example ” ICLR 2015.

[7] N. Papernot, et al. “ Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples ” ArXiv 2016.