敵対的生成ネットワークを用いた異常検知とその改良手法の評価

平内 雅則[†1]

概要： 機械学習を活用した異常検知において教師なし学習の手法を用いる有効性は高い．急速に増加する様々なデータに応じて検知精度を高めるための研究開発が行われている．また，敵対的生成ネットワーク(GAN)による異常検知手法が提案されているが，先行研究におけるネットワーク構成ではマッピングが 1 つに定まらず再構成能力の低下につながることが指摘されている． 本論文では，GAN に基づく異常検知手法において，データのマッピングを一貫的に制約することによる改良手法を提案し，汎用データセットを用いた実験を通じて異常検知精度を改善できることを確認した．

キーワード： ディープラーニング，敵対的生成ネットワーク，異常検知

Anomaly Detection via Generative Adversarial Networks and Approach for Improvement

Hirauchi Masanori[†1]

Abstract: The more the amount of data due to sensor devices increases and information technology develops, the higher necessities and attempts to discover anomalies existing in data by means of machine learning are emerging. For its accuracy improvement, many methods have been developed and recently a novel approach applying Generative Adversarial Networks (GAN) to anomaly detection has been proposed. On these methods based on GAN, this paper proposes a new approach to leverage the methods by enabling consistent data mappings and shows improvement of detection accuracy through general dataset experiments.

Keywords: Deep Learning, Generative Adversarial Networks, Anomaly Detection

1. はじめに

近年, AI の要素技術として機械学習の発展が著しい．特に Deep Learning は，画像分類，物体検出，セグメンテーションなど画像関連や，自然言語処理，音声認識といった分野にまで広く応用され,従来の機械学習手法を凌ぐ結果と活発な応用研究やビジネス適用が行われている．

その中でも，特に 2014 年に初めて提案され，近年注目されている手法の 1 つに敵対的生成ネットワーク(Generative Adversarial Networks, GAN)がある．GAN は生成モデルの一種であり，データから特徴を学習することで，実在しないデータを生成することや，存在するデータの特徴を変換することができる．GAN は正解データを与えることなく特徴を学習する教師なし(ラベルなし)学習の一手法であり，そのアーキテクチャーの柔軟性から，幅広い応用研究や理論的研究が急速に進み，今後の発展が大いに期待されている．

Deep Learning の活用シーンの 1 つに画像や時系列データにおける異常パターンの検知がある．しかし，ラベル付けをして学習させる教師あり学習は，異常パターンが無数にある場合には適用が難しい．その場合，ラベルを使用せず正常例のみを学習データとする教師なし学習が効果的である．教師なし学習手法として GAN を用い，従来の機械学習の手法に比べて異常検知精度の改善を報告している研究がある[1][2]．本稿では，2018 年に提案された GAN による異常検知手法を改良する新たな手法を提案し，従来手法との比較実験を行うことでその有効性を確認する．

2. 異常検知と教師なし学習

2.1. 異常検知

センサー技術の発達や大量データの蓄積が可能

提出日： 2018 年 08 月 29 日

†1 : ISE, コグニティブ・ソリューションセンター，アナリティクス・ソリューション(ISE, Cognitive Solution Center, Analytics Solution)

になったことによって，機械学習を用いてセンサーデータの中から自動的に異常を検知する仕組みが発展している．適用分野例として,防犯カメラに映る不審者の検知や，製品の製造過程におけるキズ・異物の検知，生体センサーデータにおける異常兆候の検知などが挙げられる．異常検知には変化点検知や外れ値検知といったカテゴリーが存在するが，本稿では，あらかじめ正常と定義した状態に対してそれ以外の状態を異常として判別する異常検知の精度向上について論じる．

2.2. 教師あり学習と教師なし学習

機械学習は大量のデータの中から，潜在的な規則や特徴を見つけ出し，データの識別や予測を行うためのルールをモデルとして構築する手段であり，基本的な分類として教師あり学習と教師なし学習がある．教師あり学習は，データとそれに対応する教師データとなる正解データとラベルを用意してその関係性を学習させる方法である．教師あり学習では未知のデータは，学習ラベルのいずれかに分類される．対して教師なし学習は教師データを与えることなしに特徴を抽出する方法であり，教師データを作成する必要がないことが利点の 1 つである．

2.3. 異常検知における教師なし学習の活用

異常検知において教師あり学習は検出したい異常パターンが限られている場合には有効である．しかし，正常以外を異常と考えるような場合，出現しうるすべての異常パターンを学習データとして獲得することは容易でない．

一方で異常検知の教師なし学習は異常データを準備することなく学習でき，かつ未知の異常パターンに対しても対応できる．モデル構築では正常と定義したデータを学習させることで正常な状態の特徴を捉える．このモデルは，未知データとして正常データが与えられた場合は，捉えた特徴と近いので正常と識別し，異常データが与えられた場合は，捉えた特徴と異なっていることで正常でない状態として識別することを期待する．ここで識別されるのは，正常か異常であり，それぞれのクラスの中にそれ以上の分類パターンは存在しない．

3. 敵対的生成ネットワークの仕組み

3.1. GAN の仕組みとアーキテクチャー

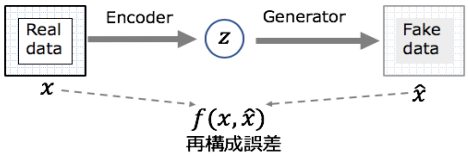

図 2 : GAN による再構成誤差のイメージ

GAN の構成には様々なアーキテクチャーが提案されているが，本稿で考察する GAN は Generator, Discriminator に Encoder を加えた 3 つのニューラルネットワーク(図 1)による構成である．通常 GAN の基本構成は 2 つのニューラルネットワークで構成される. 1 つは Generator であり，データを生成する．Generator は生成データの特徴を圧縮した低次元表現に相当するランダムノイズ(図ではzで表記)を入力することで，このノイズを所望のデータに近づけるようにマッピングを行う構造を持つ．もう 1 つは Discriminator であり, Generator が生成した偽物のデータと，本物のデータが与えられ，その真偽を判定する役割をもつ．この 2 つのネットワークを交互に競合させ学習を進めることで, Generator は本物のデータに近い偽物データを生成できるようになる．

GAN はこの 2 つのニューラルネットワークによる構成が基本だが，異常検知の先行研究[2]におい

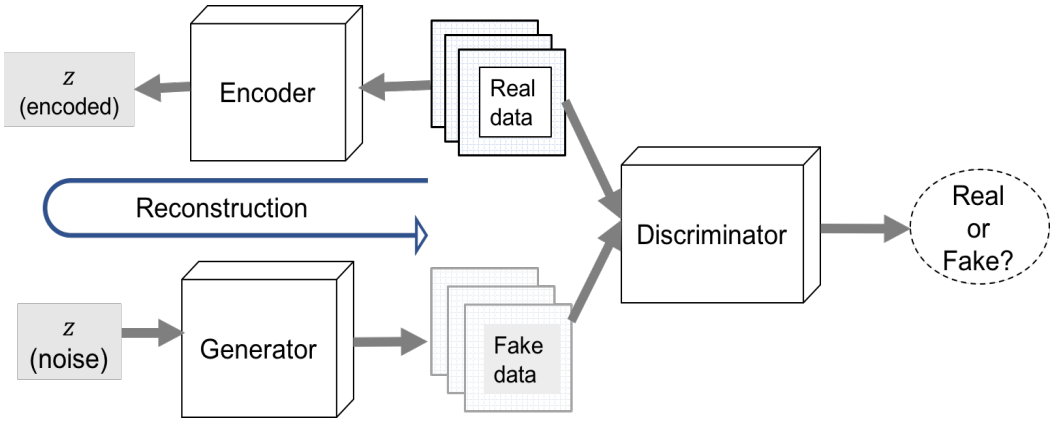

図 1 : GAN のアーキテクチャー構成

て, 3 つ目のネットワークとして Encoder を導入したアーキテクチャーが提案されている. この Encoder は Generator と逆方向のマッピング, すなわち, 与えられたデータから低次元表現のzへの写像を与える. Encoder と合わせて 3 つのネットワークを学習させていくと, 最終的に Encoder は与えられたデータに対応するzを生成できるようになる. Encoder は類似したデータを与えたとき, zの空間(以後, 潜在空間とよぶ)においても近しい位置にマップする, つまり潜在空間においてもデータの類似性を測ることができると考えられる. ここで適切に Encoder のマッピングができていれば, データからエンコードされたzを Generator に与えてデータを生成すると, Encoder に与えた元データに近いデータに復元されると考えられる.

本稿では以後, データから Encoder でzを取り出し, それを Generator に与えてデータ生成することで元のデータの復元を行うことを再構成とよぶ.

3.2. GAN による異常検知

次に上述した GAN のアーキテクチャーを利用して異常検知を行う方法について述べる.

3.2.1. 学習方法と異常検知までのプロセス

学習から異常パターンの検知までは以下のプロセスを順に行う. 模式図を図 2 に示す.

(1) 正常と定義したデータを GAN (Generator/Discriminator/Encoder) で教師なしで学習させる. ここで異常データは用いない
(2) 本物データを Encoder に与えて, データに対応するzを取得する
(3) (2)で得たzを Generator に与えることで, zから対応するデータを再構成する
(4) (2)の本物データと(3)で再構成されたデータを比較することで再構成誤差(後述)を計算する
(5) 再構成誤差に対して閾値を定めて, 閾値以上を異常として検知する.

※GAN においては, データに対する再構成誤差の取得が目的であり, (5)の閾値の決定は行わない.

3.2.2. 再構成誤差

GAN は, 学習に用いたデータの特徴を抽出して学習することで, その特徴を持ったデータを生成できるようになる. 反対に, 学習させていないデータの特徴はモデルが獲得していないので, そのようなデータの生成は難しくなる.

この仮定に基づいたものが再構成誤差であり, 正常データのみで学習を行った場合, 正常データを与えると再構成誤差は小さくなり, 異常データを与えると再構成がうまくいかず誤差が大きくなる.

再構成誤差は, 与えられたデータの異常度を表すスコアに相当し, 値が大きいほど異常度が高い(学習させた正常データと乖離している)と言える.
※再構成誤差の定義は Zenati らの提案方法[2]の 3 節を参照のこと

4. 先行研究

本節では, GAN を用いた異常検知手法の先行研究について述べる.

Schlegl ら[1]は GAN を用いた異常検知として AnoGAN を提案した. Generator と Discriminator の 2 つのニューラルネットワークの構成を用い, 教師なしで異常検知を実現している.

Zenati ら[2]は AnoGAN を発展させ, データから潜在空間へマッピングする手法として Encoder を用いた方法を提案した. Encoder を用いることでよりデータとそれに対応するzの間で, 精度の高いマッピングが実現でき, 異常検知精度が改善につながることを一般的なデータセットによる実験を通じて示している. この論文が筆者の知る限りにおいて最新の結果である. 本稿では, Zenati ら[2]の提案手法を Efficient-AnoGAN と呼ぶ.

5. 提案手法

本稿では先行研究[2]に対する改善手法としてデータと対応する潜在空間の間のマッピングの改善方法を提案する.

Efficient-AnoGAN をはじめとする Encoder を用いた構成は, データに対応する潜在空間上のzが 1 つに定まらない可能性があり, マップが一貫的にならず再構成能力の低下につながることが指摘されている[3].

Li ら[4]は GAN のコスト関数にマッピングの制約項を設けることで, 一貫的なマッピングに改善できることを示している. これは特定のタスクやアーキテクチャーに限った手法でなく, 一般に Encoder が存在する場合に適用できる手法である. 本稿ではこの手法を Efficient-AnoGAN に取り入れて,コスト関数にマッピングの制約項を設けることで, 異常検知精度の改善につながると考えた.

提案手法のマッピングの制約を模式的に表したものを図 3 に示す. 従来手法においては, Encoder から Generator へのマッピングが破線のマップとなった場合, 与えられたデータと異なるデータへマップされ, 再構成が正しく行われない. 提案手法ではこ

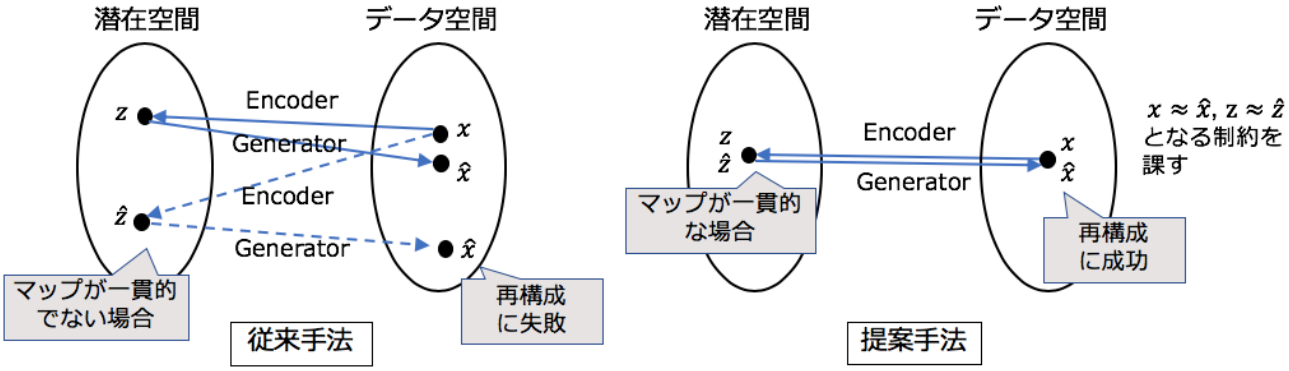

図 3：提案手法のマッピング制約　模式図

のマップが元のデータに戻るように制約するためにコスト関数を変更する(付録 1 を参照). そのために, データが存在する空間と潜在空間それぞれで, $x \approx \hat{x}$, $z \approx \hat{z}$となるような制約を課す. この制約は, Encoder と Generator によるxとzの間のマッピングに制約をつけることを意味する. データとzの間のマッピングが改善されることで, 学習させた正常データの再構成がより忠実になり, 正常データと異常データの再構成誤差の差がより顕著に現れることで検知精度の改善につながると考えられる. また, 提案手法はコスト関数だけを変更しネットワーク構成に変更がないため, 推論時間は Efficient-AnoGAN と同程度である. そのため, 本稿では有効性に関して異常検知精度を対象に検証を行う.

理論的詳細および厳密な解説は Li ら[3]の 3 節または Zhu ら[4]を参照されたい.

6. 実験

本節では, 提案手法の有効性を確かめるため, 一般に公開されているデータセットで実験を行う.

6.1. 実験用データセット

実験で使用したデータセットは以下の 2 つである.

(1) CIFAR-10[5]
airplane, automobile 等 10 カテゴリーの RGB 画像データが計 60,000 枚収集されている. 画像を扱う機械学習手法のベンチマークとして用いられることが多い.

(2) KDD CUP 99[6]
ネットワークの侵入検知に関するデータマイニングコンペティションで提供されたデータセットであり, 各レコードに正常と攻撃(異常)のラベル付けがなされている. データは, 数値データとカテゴリーデータが混合されており, 正常・異常ラベルを含む 42 項目で構成される(カテゴリーデータをダミー変数化した場合, 計 121 項目). 本実験では, 全体の 10%のデータ数で提供されている約 50 万件のサブセットを使用する.

6.2. 実験方法と精度指標

(1) CIFAR-10
10 カテゴリーの内 1 カテゴリーを正常, 残り 9 カテゴリーを異常とする. 分割された正常データは 6,000 枚存在し, その内 5000 枚を学習データ, 残りをテストデータとした. また, 他カテゴリーから異常データをランダムに 100 枚を選択し, テストデータに加える. 異常データの復元抽出を 10 回繰り返し, テストデータを 10 パターン構成する

精度は ROC(Receiver Operating Characteristic)曲線の占める面積である AUC(Area Under Curve)で評価する. ROC は異常検知手法を定量的に評価する一般的な指標であり, 横軸に偽陽性率(False Positive Rate, FPR), 縦軸を真陽性率(True Positive Rate, TPR)としたグラフで表現される.

$$FPR = \frac{FP}{FP + TN},$$

$$TPR = \frac{TP}{TP + FN}.$$

ここで,

TP : 正解が真のデータを真と予測した数
FP : 正解が偽のデータを誤って真と予測した数
FN : 正解が真のデータを誤って偽と予測した数
TN : 正解が偽のデータを偽と予測した数

である. このモデルでは異常の有無を真偽と判断する. よって FPR は, 正解が偽(正常)のデータの内, 誤って真(異常)と予測した率であり, TPR は正解が真(異常)のデータの内, 正確に真(異常)と予測した率を指す. 一般に FPR は低く, TPR は高い値を取ることが望ましいため, AUC は高い値ほど検知精度が高いとみなすことができる. ここで AUC は[0, 1]の間の値をとる指標となる. 上述の 10 パターンのテストデータの平均の AUC を求めて実験結果とする.

(2) KDD CUP 99

本稿では Zenati ら[2]と同条件の実験を行う. 全データのうち 50%をテストデータとして分割する. そして残ったデータから正常データのみを抽出したものを学習データとする.

テストデータを対象に各手法で異常スコアを算出し, 上位 20%を異常とみなす. この基準における, 再現率, 適合率および F スコアで評価を行う. 各指標は次の式で定義される. 適合率は, 真(異常)と予測したデータのうちの正解率であり, 再現率は TPR と同じである. また, 一般に再現率と適合率はトレードオフの関係にあるため, それらの調和平均である F スコアも指標として用いる.

$$Precision = \frac{TP}{TP + FP},$$

$$Recall = \frac{TP}{TP + FN},$$

$$F - score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}.$$

6.3. 比較手法

比較対象手法として以下の(1), (2)と, 提案手法(3)を比較する. 各手法の詳細は付録 2 を参照のこと.

(1) One Class Support Vector Machine(OC-SVM)
Schölkopf ら[7]で提案された手法であり, 教師なし学習により正常と異常を判定する. 異常検知を対象としたタスクにおいて広く用いられている手法である.

データセット(1)の画像を対象とした場合では, 低次元化のため Convolutional Autoencoder を前処理として用い, 低次元データに変換したのちにモデルの学習データとして用いる.

(2) Efficient-AnoGAN
関連研究で述べた通り Zenati ら[2]で提案されている手法である.

(3) 提案手法
本稿の 5 節で提案した手法である.

6.4. 実験結果と考察

本実験の環境では, Python 3.6, TensorFlow v1.7, scikit-learn v0.19 を使用した.

(1) CIFAR-10

実験結果を表 1 に示す.

提案手法は, 8 カテゴリーにおいて OC-SVM の AUC を上回っている. また, Efficient-AnoGAN と比較した場合, 9 カテゴリーにおいて改善されている. カテゴリー全体の平均値では, 従来手法より約 5 ~ 6 ポイントの改善ができている. カテゴリーごとに改善率には差があり, 大きいものでは 15 ポイント超の改善となる.

改善率には差があり, カテゴリーによっては提案手法の精度が下がっているものも見られる. 今回用いたデータセットにおいてはその画像枚数やパタ

表1 : CIFAR-10 の実験結果(AUC).

カテゴリー	OC-SVM	Efficient-AnoGAN	提案手法
Airplane	0.6623	0.7137	**0.7939**
Automobile	0.4941	**0.4977**	0.4711
Bird	0.6345	0.6865	**0.7545**
Cat	0.5327	0.5636	**0.6310**
Deer	0.6714	0.7459	**0.7509**
Dog	0.6030	0.5433	**0.6701**
Frog	**0.7227**	0.6885	0.7026
Horse	0.5979	0.5455	**0.6025**
Ship	0.6392	0.7183	**0.8090**
Truck	**0.6381**	0.4511	0.5206
全体平均	0.6195	0.6154	**0.6706**

表 2: KDD CUP99 の実験結果

モデル	適合率	再現率	F スコア
OC-SVM*	0.7457	0.8523	0.7954
Efficient-AnoGAN*	0.9200± 0.00740	0.9582± 0.0104	0.9372± 0.0440
提案手法	**0.9524**	**0.9676**	**0.9599**

* OC-SVM, Efficient-AnoGAN の結果は, 本実験と同条件の結果のため Zetani ら[2]より引用

ーンの多さから，提案手法が有効に働かないケースを見極めるのは容易でなく，現状では実験を行って確かめる必要がある.

(2) KDD CUP99

実験結果を表 2 に示す．すべての指標において提案手法は他手法を上回っている. CIFAR-10 のような画像のみならず数値データにおいても，比較対象手法より良好な結果を示すことが確認できた. 適合率と再現率の両方の値が改善されていることから，提案手法で導入したマッピングの制約が再構成誤差に有効に働いていると考えることができる.

7. まとめと今後に向けての課題

本稿では, GAN を用いた異常検知手法において, 従来手法に対して検知精度の改善手法を提案した. 一般的なデータセットを用いて精度比較実験を行い，画像データおよび数値データに対して精度向上が確認できることを示した.

今後の課題として以下が挙げられる.

(1) ネットワークアーキテクチャーおよび学習プロセスの見直しによる異常検知精度の改善
(2) より実用的なデータセットによる追加実験

謝辞

本稿の執筆にあたり，多くの助言をいただいた黒川佳昭氏をはじめ，日頃の業務でお世話になっている ISE アナリティクス・ソリューションの皆さまに改めて感謝いたします.

参考文献

[1] T. Schlegl *et al.* : Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discover, *arXiv*, 2017.

[2] H. Zenati *et al.* : Efficient GAN-Based Anomaly Detection, *arXiv*, 2018.

[3] C. Li *et al.* : ALICE: Towards Understanding Adversarial Learning for Joint Distribution Matching, *Neural Information Processing Systems（NIPS）*, 2017.

[4] J. Zhu *et al. :* Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks , *arXiv*, 2017.

[5] CIFAR-10, https://www.cs.toronto.edu/~kriz/cifar.html

[6] KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[7] B. Schölkopf *et al.* : Estimating the Support of a High-Dimensional Distribution, *Neural Computation*, *3*, 7, 1443–1471, 2001.

[8] Scikit learn documentation, http://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html

[9] A. Radford *et al.* : Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, *arXiv*, 2015.

付録 1

提案手法におけるコスト関数を示す.

$$\min_{\theta,\phi} \max_{\omega} L_{adv}(\theta,\phi,\omega) + L_{ce}(\theta,\phi)$$

ここで，θ, ϕ, ω はそれぞれ Generator, Encoder, Decoder のパラメータとし，$L_{adv}(\theta,\phi,\omega)$は GAN の学習を示す項であり，$L_{ce}(\theta,\phi)$は提案手法において追加した項でありマッピング制約を表す. 本稿の実験において，$L_{ce}(\theta,\phi)$としてxと$\hat{x}$およびzと$\hat{z}$の差の L1 ノルムを使用しており, Generator と Encoder はこの項を最小化することで，$x \approx \hat{x}$, $z \approx \hat{z}$となるように学習が進む.

付録 2

比較手法の実装における詳細を示す.

A. OC-SVM

CIFAR-10 を対象にした実験では, scikit-learn の実装である OneClassSVM を使用した. この際に使用したパラメータは, kernel および γ はデフォルト設定[8]であり，それぞれ RBF カーネルと自動選択モードを使用した. また，異常データの割合の上限を指定するパラメータ ν は，0.1 から 0.9 まで 0.1 刻みで変化させたときの最も良好であった結果を実験結果として採用している.

B. Efficient-AnoGAN および提案手法

公平な比較を行うため，この 2 つの手法間ではネットワークのアーキテクチャーやその他の学習パラメータは同じ設定を用いている. KDD CUP 99 の実験については，Zetani ら[2]の Appendix に記載されているものを使用し，CIFAR-10 を対象にした実験については，DCGAN[9]ベースのアーキテクチャーを使用している. 最適化手法として学習率 0.0002 の Adam を使用し， zの次元は 200 である. また，各カテゴリーでそれぞれ 800 エポック回し, 5 エポックごとにテストを実行して AUC を計算し，最も良好であった結果を実験結果として採用している.