私のブックマーク

言語処理分野における Adversarial Example †1

佐藤 元紀（株式会社 Preferred Networks）

1. は じ め に

ニューラルネットワークは，時としてわずかな入力の違いによって大きく異なる挙動になることが知られています．

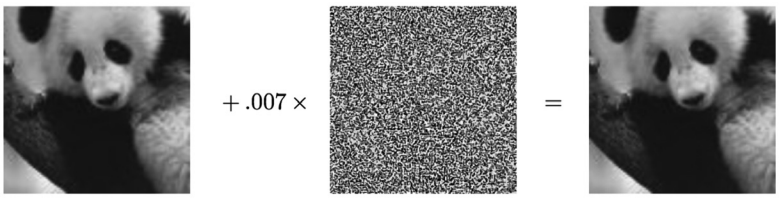

図1 画像分野における Adversarial Example の例（出典：[Goodfellow, et al.] より抜粋）

有名な例に [Goodfellow, et al.] の図1があります．

左側の画像 x に対して，微小なノイズ（Adversarial Perturbation：敵対的摂動）を足し合わせた結果が，右側の画像になっています．

左側の画像を CNN による画像認識器に入力したときには正しく「パンダ（"panda"）」と予測されるものの，右側の画像を入力すると「テナガザル（"gibbon"）」と予測されています．

同じ見た目の画像にもかかわらず，ニューラルネットワークが誤認識するこのような事例は Adversarial Example と呼ばれています．[Szegedy, et al.] や [Goodfellow, et al.] によって，このような画像の作成が微小なノイズ（Adversarial Perturbation：敵対的摂動）を足すだけで実現できることが報告され，そこから数々の研究が発表されています．

例えば，Adversarial Example を用いてモデルの正則化をする敵対的学習 [Szegedy, et al.] を半教師あり学習に拡張した研究 [Miyato, et al.] や，画像クラスに依存しない敵対的摂動を学習する Universal adversarial perturbations [Moosavi-Dezfooli, et al.] などが発表されています．

本記事では，Adversarial Example について研究をリードしている画像処理の分野から述べ，そこから自然言語処理（Natural Language Processing：NLP）に応用した際の難しさや関連研究を下記の文献などを踏まえて紹介します．

- [Goodfellow, et al.][1]
- [Szegedy, et al.][2]
- [Miyato, et al.][3]
- [Moosavi-Dezfooli, et al.][4]

2. 定 義 ・ 設 定

- 敵対的サンプル（Adversarial Example）：

  ニューラルネットワーク f の出力 y を y' に変えるような入力：

  $$y = f(x);$$

†1 http://www.ai-gakkai.or.jp/my-bookmark_vol34-no5

$$y' = f(x');\ x' = x + p$$

で定義される x' のサンプルを Adversarial Example と呼びます．p は微小なベクトルとします．

画像の分野では，見た目は x とほぼ変わらない（人が見ると y クラス）が，ニューラルネットワークに入れると出力が y' になるような画像 x' のことを指します．

・**敵対的摂動**（Adversarial Perturbation）：

モデルの出力を変えるようなノイズベクトル p（perturbation：摂動）．この摂動ベクトルはニューラルネットワークの予測が間違うように意図的に作成されます．どのように作成するかは，モデルが手元にある場合（White-box），モデルが手元にない場合（Black-box）にも依存します．

代表的な二つの設定について説明します．

・White-box：

モデル知識がある場合の設定です．モデル知識とは，モデルのネットワーク構造やパラメータ，ロス関数，活性化関数，訓練データのことです．これらのモデル知識にアクセスできる設定を White-box と呼びます．White-box での代表的な攻撃方法で Goodfellow, et al. が提案した Fast Gradient Sign Method（FGSM）があります．

モデルが手元にあるため，損失関数 Loss の計算によって入力 x に関する勾配を求めます．

$$x' = x + \varepsilon\,\mathrm{sign}(\nabla x\ \mathrm{Loss}(x, y))$$

ここで，sign 関数は正の値を＋1，負の値を－1にする関数であり，ε はどの程度，敵対的摂動を加えるかのパラメータです．つまり x の値を損失関数を高くするような方向へ ε だけ更新しています．

・Black box：

モデル知識にアクセスできない場合の設定です．アクセス可能なのは，ある入力データと出力のみアクセス可能です．例として，Web 上で提供されているモデル（Web API）の出力を変える場合が現実的に考えられます．White-box に比べ，より現実的な攻撃に近い設定です．

代表的な攻撃方法である [Papernot, et al.] が提案した手法について説明します．[5]

White-box の設定と違い勾配にアクセスすることができないため，まず少数の入力画像（数百程度）を用意し，攻撃対象となるモデルに入力画像を入力し，出力集合を得ます（Jacobian ベースの手法で入力画像の Data Augmentation を行います）．

この入力画像と出力集合のペアから，手元でニューラルネットワークを訓練します（この手元で訓練するモデルは，攻撃対象となるモデルの構造とは構造は異なります）．その後，手元のモデルに対しては勾配にアクセスできるため White-box の設定と同様に FGSM などで敵対的摂動を求め，Adversarial Example を作成します．

この Adversarial Example を攻撃対象のモデルに入力すると，不思議なことに出力を間違えてしまうことが報告されています．

これは Adversarial Example は，あるモデルで求めたものが，他のモデルに対しても誤分類する性質 Transferability（転移性）があることが知られており，Black-box の設定で使われています．

・**敵対的攻撃**（Adversarial Attack）：

Adversarial Example を作成し，モデルに入力し，モデルの出力を変える攻撃のこと．攻撃の達成目標に関しても二つの設定が知られています．

* Targeted attack：モデルの出力を特定の目的クラスの出力に変えるもの．
* Non-targeted attack：モデルの出力を y 以外の任意のクラスの出力に変えるもの．

言語処理の分野に限らず，画像などの分野で使われる攻撃方法について列挙します．

・L-BFGS [6]
・Fast Gradient Sign Method（FGSM）[7]

- Jacobian Saliency Map Adversary（JSMA）[8]
- C&W Attack [9]
- Substitute Attack [10]
- DeepFool [11]
- GAN-base Attack [12]

ここまで，Adversarial Example について説明をし，モデルを間違えさせることができることを説明しました．一方で，ただ攻撃をするだけでなく，モデルの汎化性能を上げることを目的とした学習方法，Adversarial Training（敵対的学習）に応用されています．

- 敵対的学習（Adversarial Training）：
  Adversarial Example を訓練データとして学習データに混ぜることで，敵対的攻撃への防御をすることができます．また一方で防御だけでなく，モデル正則化として応用されています [Szegedy, et al.]．Adversarial Training は正解ラベル y を使うため，Supervised の設定でのみ利用可能でしたが，KL Loss を用いて正解ラベル情報を不要にし，Semi-supervised の設定に拡張した VAT という手法も提案されています [Miyato, et al.].

- 敵対的生成ネットワーク（Generative Adversarial Network：GAN）[13]：
  二つのネットワーク（Generator と Discriminator）を競わせながら学習させるアーキテクチャとして提案されました．本物らしい画像生成が可能になったことで知られています．Adversarial Example は，モデルに入力したときに誤分類する事例をつくることが目的なのに対して，GAN は生成モデルの学習方法であるため概念的に異なります．本記事では GAN については焦点を当てず，NLP における Adversarial Example，および汎化性能を上げる Adversarial Training についての研究を紹介します．

3. NLP 分野における Adversarial Example と難しさ

まず，Adversarial Example を NLP の分野で作成する際にどのような違い・難しさがあるかを説明します．[Zhang, et al.] で紹介されている NLP 分野で Adversarial Example を作成する際の難しさは，以下のとおりです．

1. Discrete（離散的）：画像では入力の各画素値は連続的な値で表現されています．一方 NLP では入力が文，単語，文字などの記号であり離散的です．そのため画像ドメインで使えた手法をそのまま適用できない場合があります．
2. Perceivable（知覚的）：一つ目の項目と関連しますが，単語や文字の「挿入・削除・編集」は，人の目からは認識しやすく，文法的な違いなどが明らかになります．画像分野ではノイズベクトルを足しただけでは見た目は変わらないことが多いですが，一方，NLP の場合は入力文に対する操作が知覚されやすい性質があります．
3. Semantic（意味的）：入力文に対して何かしら単語や文字の入れ替えを行った場合，意味がそもそも変わってしまったり，意味の破綻した文ができてしまう場合があります．

4. NLP における敵対的攻撃

NLP の分野における攻撃について，White-box・Black-box のそれぞれに分類し研究を紹介します．まず White-box の設定の研究を紹介します．同時期に行われた ACL 2018 および IJCAI 2018 で発表された [Ebrahimi, et al.],[Sato, et al.],[Liang, et al.] の 3 件の論文です．モデルの損失関数に対する勾配を用いて，入力文の文字操作をしています．文書分類などのタスクで分類器を騙すような入力文を作成しています．

4·1 White-box attack

- HotFlip：White-Box Adversarial Examples for Text, ACL 2018 [Ebrahimi, et al.][14]
  NLP における White-box の設定で敵対的攻撃をする代表的な論文です．文書分類をする文字 CNN モデル [Kim, et al.] に対して攻撃を行います．
  文字列の操作
  flip（置換え）：（例）mood → mooP
  insert（挿入）：（例）past → pas!t
  delete（削除）：（例）ships → hips
  という操作をし，Adversarial Example を探します．flip, insert, delete のそれぞれの文字操作を one-hot ベク

トルの操作として定義しました．one-hot ベクトルとは NLP で使われるベクトル表現で，文字種を |V| としたとき，入力された文の各文字をその文字に該当する座標のみが 1 でほかが 0 であるような |V| 次元のベクトルのことです．基本的なアイディアは，one-hot ベクトルに対して損失関数 Loss(x, y) の勾配を用いて現状のモデルが間違えやすい入力文字列を探索するというアイディアです．

結果として，NLP においても画像と同様に Adversarial Example に対してモデルがぜい弱性をもつことを示しました．提案手法が 90％以上の成功率で Adversarial Example をつくれることを示しました．この Adversarial Example は入力文字の 10％までを変更できる制約での実験です．文字編集の際に，編集後の単語が語彙に存在しないように制約を加えています．これは Semantic が変わらないようにすることが目的です．

また文字だけでなく単語レベルでの flip, insert, delete も実験を行っています．Semantic が変わらないように，（1）単語ベクトルの類似度が 0.8 以上であること，（2）同じ品詞であること（名詞は名詞にしか置き換えない），など置換え時にいくつか工夫を加えています．

・Interpretable Adversarial Perturbation in Input Embedding Space for Text, IJCAI 2018 [Sato, et al.][15]

単語ベクトル w に対して，損失関数 Loss(x, y) の勾配を計算すると，敵対的摂動（Perturbation）は，単語ベクトルの点が存在しない方向を向いてしまうことがあります（図 2）．そこで，勾配を求める際に単語ベクトル点が存在する方向に勾配を制限することで，敵対的摂動が単語の置換えに対応することを目指した研究です．

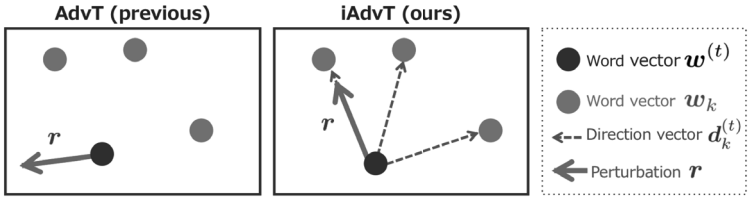

図 2　提案手法：単語が存在する方向に敵対的摂動を制限する（出典：[Sato, et al.]）

結果として，ポジティブ・ネガティブを分類する極性分類タスクで，「better → worse」と単語置換えをすることで出力が変わることや，文法的に誤っている部分を検出する文法誤り検出（Grammatical Error Detection）タスクで，検出器が間違えるような入力文をつくることが可能なことを示しました．

同時期に発表された [Ebrahimi, et al.] と比べると，[Sato, et al.] は敵対的学習に提案手法を用いることで，分類器の性能は同等以上のまま，敵対的摂動が単語の置換えと解釈できるため，解釈性が上がることを示した点が大きな違いです．

・Deep Text Classification Can be Fooled, IJCAI 2018 [Liang, et al.][16]

[Ebrahimi, et al.] と同様に文書分類の分類器に対して Adversarial Example を生成することを目指した研究です．

アイディアとしては，単語ベクトルに対する損失関数 Loss(x, y) の勾配を求め，勾配の大きさをもとにどの単語が最も予測に寄与しているかを求めます．

各予測クラスに対して，以下のように予測に寄与する単語辞書を構築します．

（例）“Building”クラス
“historic”（7 279 回）
“building”（4 954 回）
“church”（3 978 回）
...

このような辞書を元に，テストデータに単語を追加したり，辞書中の単語と置き換えることで新しい文を作成し探索することで，Adversarial Example を作成することを示しました．NLP の場合は文の変更などが容易にできるため，作成した文をモデルに入力し誤認識するかを総当たり的に試行することが多いです．

実験として，実際に作成した Adversarial Example に対して人手評価をし，人は正しく分類できるが，分類器が誤分類することを確認しています．

・Comparing Attention-Based Convolutional and Recurrent Neural Networks: Success and Limitations in Machine Reading Comprehension, CoNLL 2018 [17]

Attentionに着目して，CNNとRNNのロバスト性を比較した論文です．モデル内部のattention機構を利用し，文章レベルの質問応答タスクについて正しい答えを出力する際により大きなattentionが張られている文を選択してきます．そして，最もattentionが張られている単語をランダムに選び置き換えて新しい文をつくり出します．Attention自体への攻撃ではなく，モデルが重要視している単語の選定にAttention機構を用いています．

4·2 Black-box attack

次にBlack-boxの設定の研究を紹介します．

画像とは違い，NLPの場合は入力文を人手やヒューリスティックで置き換えることができるため，以下紹介する研究でも，人手で入力文を変更したり，類義語辞典（synonym）や言い換え表現（paraphrase）を用いてヒューリスティックに入力文を変更することが多いです．

・Adversarial Examples for Evaluating Reading Comprehension Systems, EMNLP 2017 [18]

Reading Comprehensionと呼ばれる読解理解のタスクで，システムの予測を間違える入力文の作成について議論した論文です．

まず読解理解とは，システムが複数文（Paragraph）を読み，そのうえでユーザが質問をしシステムが適切な返答をするタスクのことです．システムが返答を間違えるように，Paragraphの末尾にクラウドワーカが人手で作成した文を追加することで，State-of-the-artの性能を出していたシステムが簡単に予測を間違えてしまうことを示しました．

・Synthetic and Natural Noise Both Break Neural Machine Translation, ICLR 2018[19]

ニューラル機械翻訳（Neural Machine Translation）について，モデルが間違えるような入力文を作成することを目指した研究です．アイディアとしては，ランダムに入力文の1単語の文字を置換し，新しい文を作成するというとてもシンプルなものです．

実験結果として，これらのヒューリスティックな文字列の置換ルールで，翻訳元の文を編集すると，評価スコアBLEUが42.54 → 17.42（FrenchからEnglishへの翻訳タスク）というように簡単に悪化することを示しました．翻訳元の文の意味自体が変わってしまっている可能性もありますが，1単語のみの置換えでここまでBLEUが落ちることは驚きです．

・Semantically Equivalent Adversarial Rules for Debugging NLP Models, ACL 2018 [20]

言い換え表現（paraphrase）を用いて，意味的に入力文と変わらない文を作成します．また提案手法を使ってdata augmentationをかけて汎化性能が上がることを示しました．

・Generating Natural Adversarial Examples, ICLR 2018 [21]

入力文がより自然なAdversarial Exampleを生成する研究です．実際のデータ中に存在しそうな，文を生成することを目的としています．一般的にAdversarial Exampleは入力データに摂動を加えますが，GANの潜在空間上で入力に近くかつ誤識別をする文を生成できることを示しました．意味自体が異なる文を生成してしまう可能性もありますが，GANの生成モデルの潜在空間がうまく学習できていれば，意味的に似た文が生成できるだろうというアイディアです．

5. NLPにおける敵対的攻撃に対する防御

Adversarial Attackを防ぐ方法として，すべての攻撃に対応できる防御の手法は確立されてはいないのですが，一般的によく使われる方法と合わせてNLPでの研究をいくつか紹介します．

Reading comprehensionに対するAdversarial Exampleを扱った論文[Jia, et al.]では，水増ししたデータセットで学習させることで，攻撃に対して頑健になったことを示しました．しかし，別の攻撃手法に対してはぜい弱であることも示されており，あらゆる攻撃手法に対して有効ではない点も防御の難しさを示しています．

・Virtual Adversarial Training：A Regularization Method for Supervised and Semi-Supervised Learning, ICLR 2017 [22]

文書分類のタスクで VAT [Miyato, et al.] を用いることで当時の State-of-the-art を更新をしています．論文では特に防御を目的とはしていませんが，一般的に Adversarial Training が有効な手段の一つといわれています．

機械学習全般での防御方法でも議論が多くあり，ICML 2018 の Best paper [Athalye, et al.] では，ICLR 2018 で提案された論文 9 本中 7 本の論文の防御方法が攻撃可能であることが示されるなど，まだまだ研究が盛んに行われている分野です [Athalye, et al.][23].

6. 自然言語以外のモーダルやマルチモーダルタスクへの適用

自然言語に限らず，画像とテキストのマルチモーダルタスクである Visual Question Answering（VQA）や Image Captioning のモデルの Adversarial Example に関する研究や，音声認識のモデルに対しての研究も紹介します．

・Visual Question Answering（VQA）

Fooling Vision and Language Models Despite Localization and Attention Mechanism, CVPR 2018 [24]

・Image captioning

Attacking Visual Language Grounding with Adversarial Examples：A Case Study on Neural Image Captioning, ACL 2018 [25]

・音声認識

Audio Adversarial Examples：Targeted Attacks on Speech-to-Text. [26]

7. サーベイ論文・解説記事

7·1　サーベイ論文

・Adversarial Attacks on Deep Learning Models in Natural Language Processing：A Survey [27]

・Towards a Robust Deep Neural Network in Text Domain A Survey [28]

・Analysis Methods in Neural Language Processing：A Survey [29]

7·2　解説記事 [30], [31]

7·3　論文集 [32], [33]

7·4　ツール集

・CleverHans [34]

・AdvBox [35]

8. お　わ　り　に

本記事では，自然言語処理における Adversarial Example および敵対的学習について紹介しました．

自然言語処理の場合，特に生成した文が意味的に近いのか，変わっていないかを，人手判断や言い換え辞書（paraphrase）を使うことでヒューリスティックで評価している部分も多く，まだ課題が多い分野と言えそうです．

最後に，このブックマークを通してこの分野の研究を行っていく人が増えてくれると大変嬉しく思います．

謝　辞

本記事を執筆するに当たり，アドバイスをいただいた株式会社 Preferred Networks の菊池悠太さん，小林颯介さんに感謝します．

[1] https://arxiv.org/abs/1412.6572
[2] https://arxiv.org/abs/1312.6199
[3] https://arxiv.org/abs/1507.00677
[4] https://arxiv.org/pdf/1610.08401.pdf
[5] https://arxiv.org/abs/1602.02697
[6] https://arxiv.org/abs/1312.6199
[7] https://arxiv.org/abs/1412.6572
[8] https://arxiv.org/abs/1808.07945
[9] https://arxiv.org/abs/1608.04644
[10] https://arxiv.org/abs/1602.02697
[11] https://arxiv.org/abs/1511.04599
[12] https://arxiv.org/abs/1710.11342
[13] https://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf
[14] https://www.aclweb.org/anthology/P18-2006
[15] https://www.ijcai.org/proceedings/2018/0601.pdf
[16] https://www.ijcai.org/proceedings/2018/0585.pdf
[17] https://www.aclweb.org/anthology/K18-1011
[18] http://aclweb.org/anthology/D17-1215
[19] https://openreview.net/forum?id=BJ8vJebC-
[20] https://aclweb.org/anthology/P18-1079
[21] https://openreview.net/forum?id=H1BLjgZCb
[22] https://arxiv.org/abs/1704.03976
[23] https://arxiv.org/abs/1802.00420
[24] https://arxiv.org/abs/1709.08693
[25] https://www.aclweb.org/anthology/P18-1241
[26] https://arxiv.org/abs/1801.01944
[27] https://arxiv.org/abs/1901.06796
[28] https://arxiv.org/abs/1902.07285
[29] https://arxiv.org/abs/1812.08951
[30] https://openai.com/blog/adversarial-example-research/
[31] https://openai.com/blog/robust-adversarial-inputs/
[32] https://github.com/yenchenlin/awesome-adversarial-machine-learning
[33] https://github.com/chbrian/awesome-adversarial-examples-dl
[34] https://github.com/tensorflow/cleverhans
[35] https://github.com/advboxes/AdvBox