修 士 論 文

雑音除去自己符号化器を用いた
多様なテキストからの敵対的サンプル生成手法

Generating Adversarial Examples from Diverse Text using Denoising Autoencoder

東京大学 大学院情報理工学系研究科
電子情報学専攻

氏 名 **48–176435** 保田 和彦

指導教員 喜連川 優 教授

提 出 日 平成 **31** 年 **1** 月 **31** 日

概 要

深層学習に基づくモデルは，入力に小さな変化を加えるだけでモデルの出力が（人間の感覚に反し）大きく変化することがあり，この不安定な挙動がモデルの解釈性と頑健性を確保する上で大きな問題となる．そこで，入力に小さな変化（摂動）を人工的に加えてモデルの出力が大きく変化する事例（敵対的サンプル）を生成し，この敵対的サンプルを深層学習モデルの解釈や頑健性の改善に用いる手法が近年研究されている．敵対的サンプルは通常，入力データの近傍では出力が変わらないという前提に立ち摂動を加えることで生成される．しかしながら，入力が離散かつ可変長系列となる自然言語処理タスクでは，近傍の入力（文）を見つけることが困難であるため，単語レベルで編集する単純な手法が主流である．結果，これらの手法で生成される敵対的サンプルは多様性に乏しく，特にモデルの解釈に用いるには十分ではない．

そこで，本研究では雑音除去自己符号化器を用いて文生成モデルを訓練し，文生成時にモデルの隠れ状態に摂動を加えることで多様な敵対的サンプルを生成する手法を提案する．この時，隠れ状態に意味表現学習を加えることにより，摂動を加えた時の意味の変化を抑制した．極性分類問題において，学習モデルが正解したデータに対して敵対的サンプルを生成出来た割合，及び追加学習時の精度で評価及び人手によるアノテーションを行い，生成したサンプルに関して主観的・客観的分析を行う．

目 次

図 目 次

表 目 次

第1章　はじめに

1.1　敵対的サンプルによる深層学習モデルの理解

深層学習モデルは画像処理や自然言語処理を始めとする様々な分野のタスクで目覚ましい性能向上を達成しているが，実応用に用いる上ではモデルの入力データに対する敏感性が大きな障害となっている．人間では区別がつかない入力の些細な変更であっても，大きく結果を変化することがある [2]，そのため学習したモデルがどのように出力を行うのかを理解することが深層学習モデルを実応用で運用する際に重要になっている．

深層学習モデルの評価は，通常の機械学習と同様に対象タスクに沿ったテストセットにおける精度などの数値指標を用いて行われる。前述のモデルの振る舞いを把握するという観点でテストセットとして必要となる要件は，定量的に評価可能で，タスクで起きうる事象に対して網羅的かつ誤った正解が付与されたサンプルがないことである．しかし，広大な入力空間に対して網羅的なテストセットを効率的に作ることは難しい．大規模なデータセットを機械的にもしくはクラウドワーカを使い作成することにより量によって網羅性を解決しようとしたとしても，見た目以上に簡単な問題に帰着してしまったり [3]，アノテーションを行う人間が同じ傾向の問題を作成してしまう [4] などのアノテーションバイアスが入り込んでしまう．網羅的でないテストセットでは，モデルが誤って予測する例について取りこぼしている例が存在する．

この問題に対し，与えられた深層学習モデルに対してその挙動を理解するためのサンプル（敵対的サンプル）を生成する研究が近年行われている。敵対的サンプルは深層学習モデルの学習に用いた事例を元に，モデルが予測を誤る例を効率的に生

成したものである．敵対的サンプルは画像処理分野で発案され，近年自然言語処理分野でも多く研究がされはじめている．しかし，画像処理と異なり自然言語処理タスクは離散的な入力をとるため，画像処理における敵対的サンプル生成手法と同じ手法を使うことは出来ず，単語の入替えを行う単純な手法が主流である．そのため，生成できる敵対的サンプルは元の文から大きく変化することはなく，敵対的サンプルとして使用できる例が少ない．

1.2 本研究の目的と貢献

本研究では，より多くの入力テキストに対する敵対的サンプルの生成に取り組む．そのため，雑音除去自己符号化器 [5,6] を用いた文生成モデルを訓練し，文生成時にモデルの隠れ状態に摂動を加えることで敵対的サンプルを生成する手法を提案する．また，敵対的サンプルの比較評価に関して客観的評価と主観的評価の両方を行うことにより，各手法の違いを明らかにする．客観的評価では実際に対象モデルを騙せた割合，文の自然さ，共通単語数による原文との類似度を測定し，主観的評価では生成した敵対的サンプルの流暢さ，原文とラベルが変化していないかの評価を行う．実験の結果，提案手法は原文と表層的な類似度が高く，流暢度の高い敵対的サンプルの生成が可能であることを確認した．

1.3 本論文の構成

本論文の構成は以下の通りである．

第 2 章 敵対的サンプルの定義と設定条件について述べる．

第 3 章 自然言語分野における敵対的サンプル生成手法についてまとめ，提案手法と比較を行う手法に関して詳細に説明を行う．

第 4 章 提案手法である雑音除去自己符号化器を用いた生成モデルとそれを用いて敵対的サンプル生成する方法について述べる．

第 5 章　生成した敵対的サンプルの評価を行うための尺度について述べる.

第 6 章　分類タスクに関して敵対的サンプルの生成を行い，客観的評価・主観的評価を行った結果をまとめ考察を行う.

第 7 章　全体のまとめと今後の課題について述べる.

第2章　基礎知識

この章では，本研究のテーマである敵対的サンプル及び提案手法で使用する自己符号化器について述べる．まず，敵対的サンプルの定義・敵対的サンプルの生成を行うときに使用可能な情報や前提条件の明確化を行う．また，今後の手法や関連研究で使用されている自己符号器に関する基礎的な知識に関して説明する．

2.1　敵対的サンプル

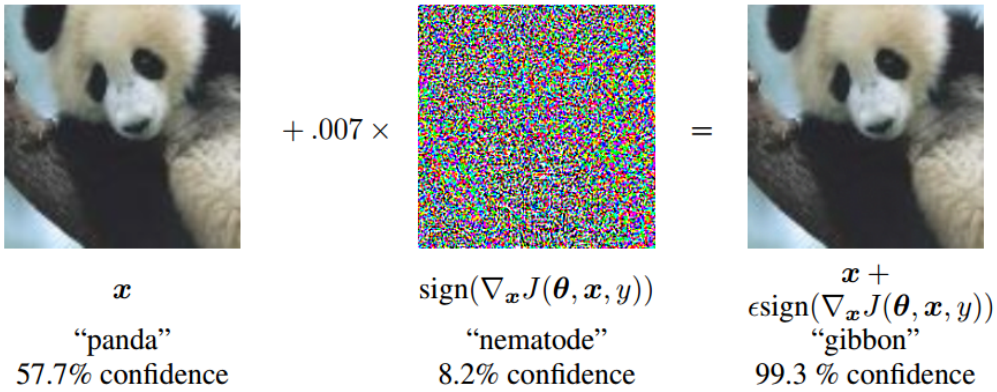

図 2.1: Goodfellow らによる敵対的サンプル例（Figure 1）[2]．左の画像では正しく「パンダ (panda)」と分類出来ているが，微小な摂動を加えた右の画像は「テナガザル (gibbon)」と分類されている．

深層学習モデルは人間では区別出来ない小さな摂動によって大きく予測結果を変えることが [2] で明らかになり，深層学習の評価を行う際に問題となる（図 2.1）．敵対的サンプルは図 2.1 のような小さな変化によってモデルが誤答してしまう入力の

ことを指す．深層学習モデルは入力の微小な変化による影響が大きく，テストデータで高い精度を示す一方で，実世界で運用する時に人間の直観に反した精度低下に陥ることがある．一例として，一時停止標識の一部に変更を加えることで，自動運転のために標識を認識するシステムの誤作動を引き起こすことが可能である [7]．自動運転やスマートスピーカーなど深層学習技術が現実世界で応用されるに従い，それらのモデルに対する攻撃が可能となるため，モデルの脆弱性の理解と対応が今後重要になってくる．敵対的サンプルは画像分野においては多くの研究がなされているが，言語分野ではまだ特定のタスクやヒューリスティックに基づいた手法が多く，汎用的な敵対的サンプル生成手法や分析が求められている．

本論文では敵対的サンプルの生成において [8] を参考に次の前提条件を定めた．

前提条件 1 攻撃者が敵対的サンプルを生成するにあたってテストステージに置いてのみ攻撃を行える．つまり対象については訓練済みのモデルのみ与えられ，訓練データや訓練方法について知ることや関与することは出来ない． また攻撃者は対象のモデルの情報（構造やパラメータ）を使っても良い．

前提条件 2 SOTA のモデルは深層学習が多く占めるため，深層学習モデルに対する攻撃のみ考える．従来の機械学習モデルや単純なニューラルネットワークを対象としないが，同様の敵対的サンプルが効果を示すことが知られている [9,10]．

前提条件 3 敵対的サンプルか否かはモデルが誤答したかでのみ評価を行う．訓練データの漏洩 [11] など他の脆弱性については考慮せず，モデルの精度に直結する指標のみ考える．

以上の前提において攻撃つまり敵対的サンプルの生成は，訓練済みのモデル f と入力 x が与えられた時，入力に近いが出力結果が元とは違う x' を求める問題として

定式化できる．

$$\min_{x'} \mathrm{dist}(x - x')$$
$$\text{s.t. } f(x) \neq f(x')$$

モデルの目的関数が分かっている場合は入力に目的関数を最大化する方向に摂動を加えることで敵対的サンプルを作ることが出来る [2]．これは勾配が計算出来る場合汎用的に使用可能な手法である．J を損失関数とすると，摂動 η を次のようにして求めることが出来る．

$$\eta = \epsilon \mathrm{sign}(\nabla_x J(x, l))$$
$$x' = x + \eta, f(x') = l'$$
$$\text{s.t. } f(x) = l, l \neq l'$$

しかしながら，入力が言語の場合，言語データは離散的な記号（単語）であるため，勾配から計算した連続な摂動を入力に加えることが出来ない．そのため，言語分野の敵対的サンプルの生成は他の手法が必要となってくる．具体的な既存手法については 3 章で説明する．

敵対的サンプルの生成（攻撃）を行う時の一連の流れを図 2.2 に表す．攻撃の対象（target）が存在し，テストデータ x が与えらた時に，敵対的サンプル x' を生成する．この時 x' はもとの x とモデルの出力が同じとなっていた場合，攻撃は失敗である．また，x' が “自然” でなければ，生成失敗とみなす．“自然” とはそのサンプルが実世界のデータに似ているということと，元のサンプル x に似ている（つまり target の出力が変化するべきでない）ことが要求される．

2.2 敵対的サンプルの分類

敵対的サンプルは生成を行う時の前提条件によって，white-box / black-box と target / non-target 攻撃に分類される．white-box / black-box は敵対的サンプルの

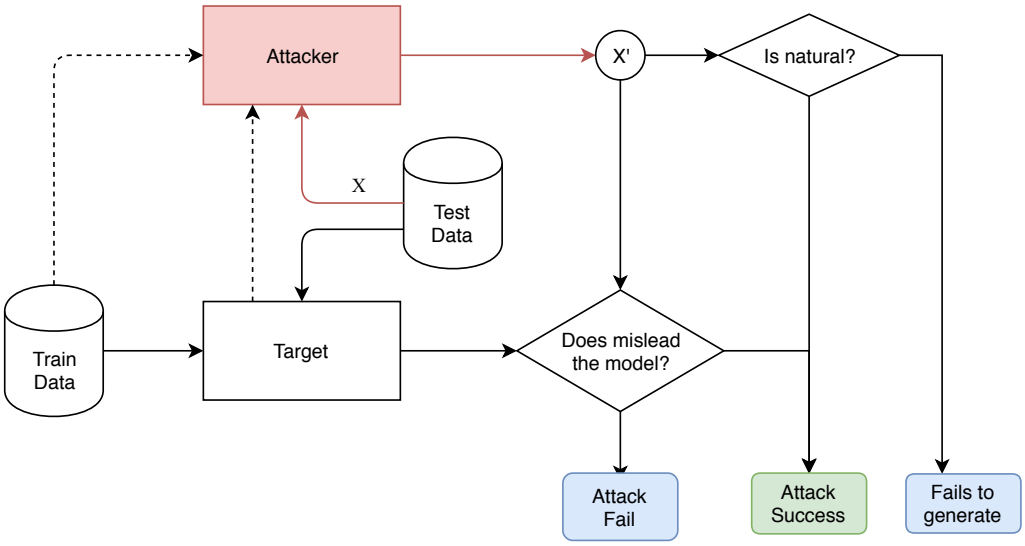

図 2.2: 敵対的サンプルの生成フロー

生成時に使用可能な情報の量によって分類される．white-box は対象のモデルの構造やパラメータなどすべての情報を収得出来るという設定で生成を行う．式 (2.1) のように勾配を計算する手法の場合は white-box である．一方で black-box は対象のモデルの入出力のペアのみが分かるという設定で生成を行う．一般に black-box の方が情報が少ないため，難しい問題となるが，入力が言語では前述の通り勾配を計算する手法が使えないため，モデル内の情報にアクセスする利点が少なく，black-box の設定で行われる研究が多い．white-box の研究の例として，単語の埋め込み表現で摂動を加える手法 [12] や，最も結果への寄与が大きい単語や文字を他のものと入れ替える手法 [13, 14] が存在する．一方で，black-box では入力となる文や文章を “編集” して敵対的サンプルを生成する手法が多い．typo のような人間では元の単語を予測出来る変換 [15]，単語の入替え [16] や統語上の言い換え [17] を使った手法が存在する．

敵対的サンプルを入力した時のモデルの出力結果によって，target / non-target 攻撃に区別される．non-target 攻撃はモデルが正解とは違う出力をすれば，攻撃成功とみなす．target 攻撃はモデルが違う出力をするだけではなく，攻撃の成功に特定の

出力になることを要求する．たとえば，分類器を対象とする場合は正解ラベルとは違う出力をすれば，non-target 攻撃に成功したと認められ，特定のクラスに誤分類した場合は target 攻撃に成功したと認められる．誤答するだけでは成功とならないので target 攻撃の方が難易度の高い設定となっている．言語分野の真のラベルを変更せずに敵対的サンプルの生成することが難しいため，未だに分類タスクの non-target 攻撃に関するものが多く target 攻撃に関してはまださほど研究がなされていない．

本研究では，black-box の設定での non-target 攻撃に関する敵対的サンプルを対象とする．black-box を対象とする理由は攻撃対象のモデルに依存せずに生成可能な手法の方が重要であると考えるためである．自全言語における target 攻撃は今後の課題とする．

2.3 自己符号化器

自己符号化器とは入力をそのまま出力するモデルで，画像処理分野などで事前学習に使えることで注目を浴びている表現学習の一種である [18]．自己符号化器を定式化すると次のようになる．入力 x，入力層 f，復元層 g を用いて，損失関数の最小化を行う．

$$
\begin{aligned}
y &= f_\theta(x) \\
x' &= g_{\theta'}(y) \\
\min_{\theta,\theta'} \sum_i & L(x_i, x'_i)
\end{aligned}
$$

この時，入力層の出力 y を中間表現，f をエンコーダ，g をデコーダという．損失関数は交差エントロピー損失がよく使われる．自己符号化器によってデータ空間が学習され，他の教師あり学習に転移できる．また，エンコーダとデコーダを用いたモデルは汎用性が高く様々なタスクに使用されている（機械翻訳，画像生成，キャプション生成 etc.）

言語処理では，エンコーダ・デコーダとして再帰型ニューラルネットワーク（RNN）及びその亜種がよく使われる（図 2.3）．RNN は可変長の系列を扱うことが出来るモ

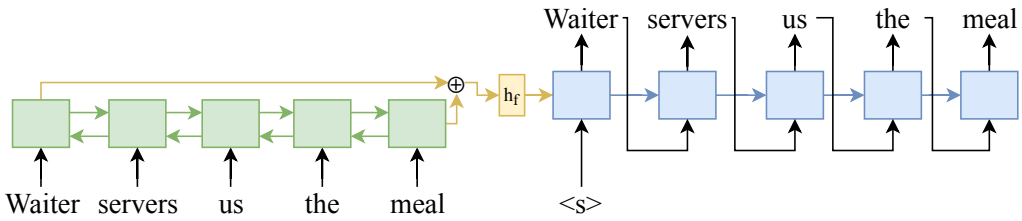

図 2.3: 双方向 RNN を用いた自己符号化器

デルで，文のように可変長のトークン列からなる入力を扱うことが出来る．デコーダでは各ステップごとに 1 トークン出力し，出力したトークンを入力として次のトークンの出力を行う．RNN は長い系列を与えると最初の情報を忘れてしまうという問題があるため，双方向 RNN が提案されている [19]

2.4 雑音除去自己符号化器

中間表現が入力の空間より広い表現能力を持っていた場合，AE は恒等写像を学習してしまうという問題がある．例えば文を入力とする AE では，中間表現の次元が入力の系列長より大きかった場合，中間表現の i 次元目に i 番目の単語の id を保存してしまえば、入力を完全に保存することが出来るが，良い表現学習になっているとは言えない．この問題を抑制するために, 中間表現の次元を制限したり正則化項を追加する場合がある．DAE は正則化と違うアプローチで表現学習を行うために提案された AE の一種である [5]．DAE は入力に破壊的な変更を加えることで，恒等写像が最適な応答にならないようにしている．入力にノイズを加え，破壊的な変更を加えることでデータが分布している空間から外れるので，DAE はノイズを取り除き適切な分布上に戻すように学習を行っていると考えられる．入力が文の場合, 不完全な文から正しい文に直す学習を行うため，文法などの統語情報が学習されると考えられる．

Lample ら [6] による文にノイズを加える手法を説明する．文中の単語の順序の入替えと単語の削除を行うことで破壊的変更を加える．まず，単語の順序を効率良く少量の変更を加えるために，次の操作を各文に行う．文を単語に分割し，各単語に

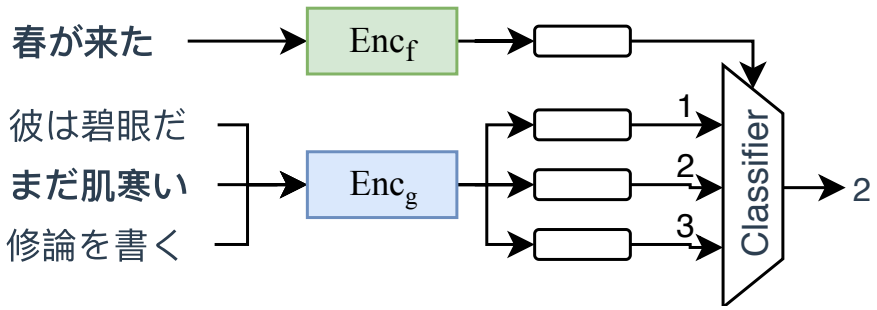

図 2.4: Quick Thoughts が行うコンテキスト予測タスク

文頭から順に連番をつける．その番号に一様分布 $\mathcal{U}(0, k+1)$ から抽出した乱数を足し，各単語を番号が昇順になるように並べることで，単語の順序を入れ替える．次に各単語を確率 p で取り除いたものをはノイズを加えた入力とする．

2.5 Quick Thoughts

Quick Thoughts（QT）は文表現を効率的に学習するために提案された手法である [20]．文表現とは離散的な文を低次元で連続的なベクトルで表したものであり，学習した文表現は分類問題などに活用される．QT は同じコンテキストの文を推定するタスクの学習を行うことで文表現を獲得する（図 2.4）．これにより文の表層のみならず意味を考慮したベクトル表現が学習される．目的の文（s）を入力とする Encoder f と コンテキストの候補文（c_i）を入力するエンコーダ g を用意する．c_i のベクトル表現（$\boldsymbol{Y}_i$）と文 s のベクトル表現 $\boldsymbol{x}$ の内積を取り，正しいコンテキストベクトルが最も高い値に，すなわち最も近くなるようにクロスエントロピー損失と

確率的勾配法で学習を行う.

$$
\begin{aligned}
\boldsymbol{x} &= f(s) \in \mathbb{R}^d \\
\boldsymbol{Y} &= \begin{pmatrix} g(c_1) \\ g(c_2) \\ \vdots \\ g(c_n) \end{pmatrix} \in \mathbb{R}^{n \times d} \\
\text{score} &= \boldsymbol{Y}\boldsymbol{x}^t \in \mathbb{R}^n
\end{aligned}
$$

第3章　関連研究

この章では，自然言語の敵対的サンプルを生成する手法について，方法論ごとにまとめ，2 つの代表的な手法 [1,16] について提案手法と比較を行うために詳しく述べ，それぞれの問題点を述べる．

3.1　自然言語分野における敵対的サンプル生成手法

2.1 節で述べたように言語分野では入力が離散な値となるため，勾配から計算した連続な摂動を加えることが出来ない．結果として，自然言語処理における敵対的サンプルの生成は記号（単語）のレベルで入力を置き換えること手法が多く提案されている．しかし，言語では少しの離散的な変化によって大きく意味が変わってしまう可能性があるため（e.g. “good” → “bad”）変化の量で距離を測ることが難しいという問題がある．これらの問題に対して，意味が変化しないような “編集” 方法に制限したり [17]，離散な入力を連続値に変換するアプローチが取られている [12–14]．離散的な入力を連続な値にするにあたって，単語レベルと文レベルの 2 つの手法が考えられる．単語を実ベクトルに変換する単語分散表現 [21] を用いて連続的な値に変換してから，画像分野と同じ様に摂動を加え最も埋め込み表現が近い入力単語で置き換える手法がある [14]．摂動を勾配から計算が可能なため，効率的に敵対的サンプルを生成でき，本質的に出来上がる敵対的サンプルについては単語を入れ替えるのと同じである．また，文自体を連続的な値に変換し編集する手法も考えられる．文を実ベクトルに変換してから，摂動を加えて生成モデルを使って復元を行うことで敵対的サンプルを得る手法が提案されている [1,22]

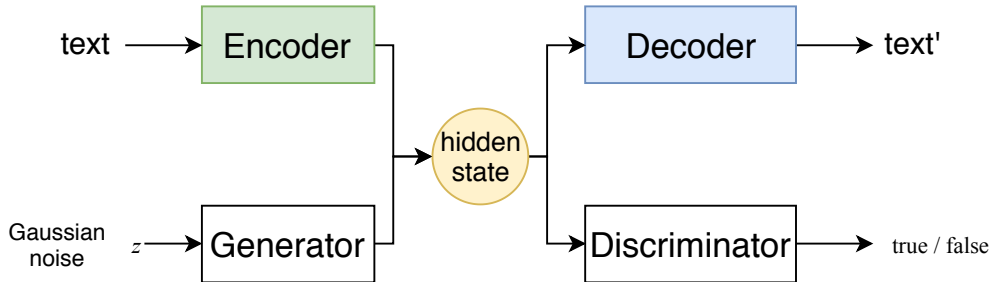

図 3.1: ARAE

次に具体的な自然言語分野における敵対的サンプル生成手法として，単語入替えによる手法と文生成の手法をそれぞれ紹介する．

3.2 単語の入替えによる敵対的サンプルの生成

Alzantot らは単語の入替えによって敵対的サンプルを生成する手法を提案している．入力文章中から単語を 1 つ選び，GloVe [23] と counter-fitting method [24] を使い類義語を見つけ置き換える．文章中の各単語ごとに置き換えを行ったサンプルを作り，対象の分類器の尤度を低くするものを残す．これを遺伝的アルゴリズムを使い複数世代置き換えを繰り返すことで，少ない置き換えで敵対的サンプルの生成を行う手法である．敵対的サンプル間の距離を単語の置き換え数として考えることができ，単語単位で類義語への変換のみを行うことでモデルが扱うタスクに依らず正解のラベルが変化しないように担保している．

3.3 生成モデルによる敵対的サンプルの生成

Zhao らは Generative adversarial networks (GAN) を使った制約を Autoencoder の潜在変数に加えた生成モデル（ARAE）[25] を用いて敵対的サンプルを生成する手法を提案している．文を入力とする素の Autoencoder は中間表現の次元が十分に大きければ，入力空間は精々語彙の文長乗しかないので入力から中間表現へ単射可能な表現の学習が可能となる．これでは入力空間（自然言語）の特徴（統語的情報

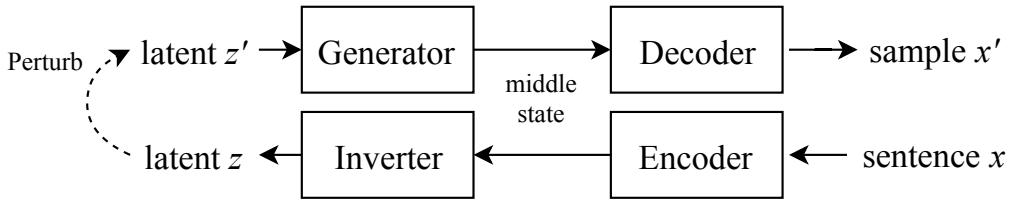

図 3.2: Zhao らの手法 [1] による敵対的サンプルの生成フロー

など）を学習することが出来ず，摂動を加えてしまうとまともな文の復元出来なくなる．そこで，中間表現に制約を加えることで，入力空間の特徴を捉えるようにしたい．制約として，中間表現が確率分布に従うようと仮定して確率分布のパラメータを学習する方法 [26] などがあるが，入力が言語場合学習が難しいことが分かっている．Zhao らは事前分布の代わりに，中間表現が本物か人工的に作られたものかを識別する識別器とその識別器を騙す生成器を敵対的に学習するモデル（ARAE）を使用した（図 3.1）．生成器は正規分布からサンプルした乱数をもとに識別器を騙す疑似的な中間表現の生成を行うことで，中間表現が入力空間の特徴を捉える様に学習されるとされる．

ARAE を使って敵対的サンプルを生成する時は，中間表現を正規分布空間に戻す逆変換器を学習する．これにより文を正規分布空間の潜在変数に落とし込めることが出来る．この潜在変数に摂動を加えることで敵対的サンプルの生成が可能となる（図 3.2）．また，敵対的サンプル間の距離は摂動の大きさとして測ることが出来る．

3.4 既存手法における課題

単語入替えを用いた手法では，多くの入力サンプルついて敵対的サンプルの生成が可能でないという問題がある．Alzantot らの手法は，可能な編集を単語の置き換えに限定しているため，元の文から多く変化せず，文法的に間違った敵対的サンプルは生成されえないという利点があるが，生成可能な敵対的サンプルの種類が少なく，元のサンプルから大きく変化しないので，短い文や類義語が少ない単語が多い文について敵対的サンプルの生成が難しいという問題がある．また，置き換え可能

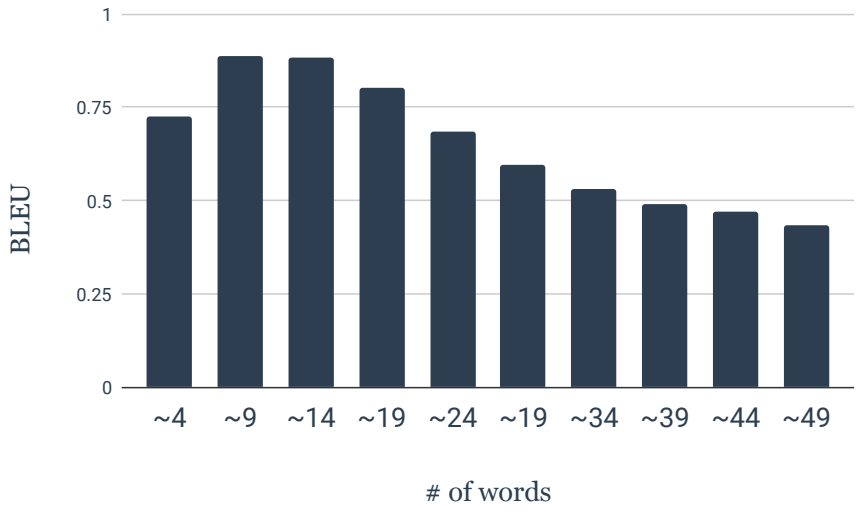

図 3.3: ARAE [1] の文長ごとの生成文精度

な単語候補に外部リソース（Glove）を使用しているため，実験により対象モデルでは未知語になるような置き換えが多く行われているということが分かった（表 6.6）．未知語への対処は問題ではあるが，Alzantot らの手法で使用している学習済みの単語分散表現を使うことによって対処出来てしまう可能性があり，単語置き換え手法の適用幅に疑問が残る．Zhao らの手法は生成モデルを使っているので，単語の置き換えと比べて削除，追加，並び替えなど多様な出力が可能である．Zhao らは中間表現を一度連続な潜在空間に写して連続的な空間から “正しい” 中間表現に戻す作業を行っているが，意味的な尺度を明示的に組み込んでいなかったため，復元した時に似た意味の文が復元される保証がない．また，潜在空間から復元まで多くの非線形変換が複数回入るので少量の摂動で大きく生成文が変化してしまう．そのため，長い複雑な文に対して敵対的サンプルを生成する場合，元の文から意味が大きく変わってしまうことがある（図 3.3）．

第4章　提案手法

4.1　Denosing Autoencoder を用いた文生成モデル

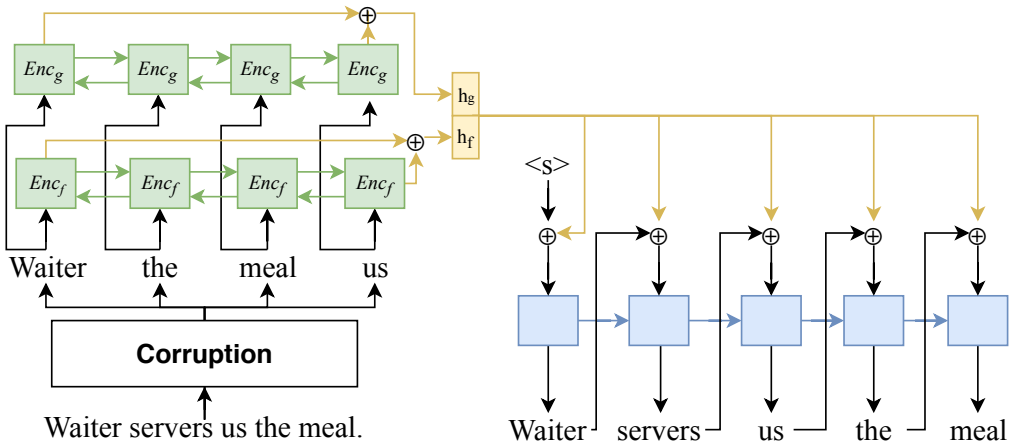

図 4.1: 提案手法の概念図

Zhao らの手法の課題であった，長文の対応と文の意味を考慮するために，本研究では雑音自己符号化器の中間表現に意味表現の学習を加えたモデルを用いて敵対的サンプルを生成する手法を提案する．通常の自己符号化器の中間表現に直接摂動を加えて復元を行うと文法的に破綻した文が出来てしまう．これは Autoencoder の中間表現からデコードするモジュールが統語的な学習をしないため，摂動を与えることにより学習された分布から中間表現が外れた時に対応できないためである．そのため，雑音除去自己符号化器を用いることでノイズに対する頑健性を上げ，摂動を加えても復元出来るようにした．提案手法は zhao らの手法と違い，中間表現を写像しないため，情報の欠落が少ないと考えられる．また，Zhao らの手法では考慮され

ていなかった意味表現を Quick Thoughts とマルチタスク学習することによって意味を考慮した生成を可能にした．

4.1.1 Quick Thoughts と DAE のマルチタスク学習

QT と DAE のマルチタスク学習を行う際は，f と g に同一の文を入力しそれぞれの中間表現を連結し線型変換して次元を落としてデコーダに入力する（図 4.1）．エンコーダ，デコーダともに RNN の一種である GRU [27] を使用した．GRU は LSTM [28] と比べて，パラメータ数が 25 %少ない上 NMT において同等の性能を示すことが分かっており，NMT と同じエンコーダ・デコーダモデルである提案モデルにも通じると考え採用した．また，長文に置いて入力情報を忘れてしまう問題を緩和するために双方向 RNN と同様に入力文を順方向と逆方向それぞれ違う GRU を用いてエンコードして最終的な出力を連結した．Zhao らと同じ実装に倣い，2 つのエンコーダの出力を連結し，線型変換をかけてデコーダの埋め込み表現と同じ次元数にしてから，埋め込み表現と連結してデコーダに入力を行った．これは隠れ状態の初期値として入力する場合と比べて学習が早くなる効果があった．

中間表現に摂動を加えた時に自然な文が生成されるために，中間表現をノイズに対して頑強にする必要がある．DAE の訓練を行う際に，中間表現にノイズを加えて学習を行った．正規分布からサンプルしたノイズに倍率 r をかけたものを中間表現に加えた．しかし，ノイズを加えた状態では自己符号化の訓練自体が難しいので，一定ステップごとに 1 以下の実数倍することで学習とともに r を小さくしていった（焼きなまし）．

4.1.2 実装及び訓練

提案手法の訓練を英語の生コーパスから抽出した文を用いて行った．各ステップごとに，DAE と QT の学習をそれぞれ 1 回ずつ行った．QT の学習の際に使用するコンテキストは入力文の前後 1 文を使用し，コンテキストの候補としてミニバッチ内の

表 4.1: モデルのハイパーパラメータ

エンコーダ・デコーダ	GRU の層数	1
	GRU の隠れ層	500 次元
	単語埋め込み表現	500 次元
	語彙数	50,000
雑音	k	3
	p	0.1
	r	0.1
	焼きなまし比	0.999
	焼きなまし間隔	100 ステップ
学習設定	学習時の最大単語長	50
	ミニバッチサイズ	128
	学習率	0.0005
	β_1	0.9
	β_2	0.999

文のコンテキストを使用した．各ミニバッチでは同一レビューの文が重複しないようにした．確率的勾配降下法の一種である，モーメンタム付き SGD [29]，Adam [30]，AMSGrad [31] を試し，AMSGrad が最も収束が早くかつ開発データにおける DAE の BLEU スコアが高かったので採用した．DAE の復元損失と QT による損失をそれぞれ足し合わせたものを損失として勾配の計算を行った．実装はすべて Pytorch (version: 0.4.1) で行った．

学習時のハイパーパラメータを表 4.1 に示す．学習時間と GPU のメモリ上限の問題から 1 文あたりの最大単語数を制限した．BLEU [32] スコアが開発データで最も高くなったモデルを敵対的サンプルの生成に使用した．

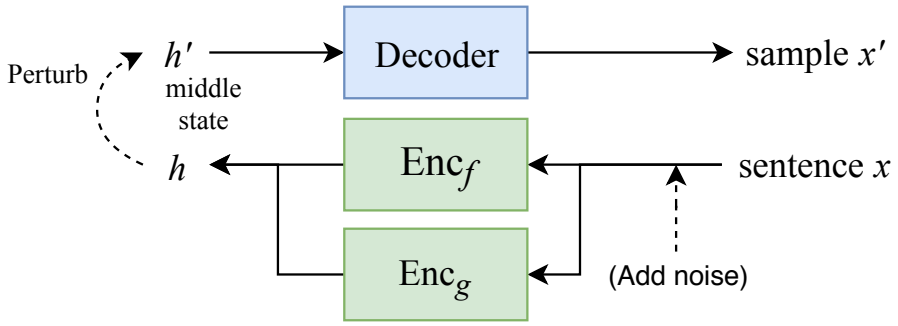

図 4.2: 提案手法による敵対的サンプルの生成フロー

4.2 敵対的サンプルの生成

敵対的サンプルを生成するために，まず上記の Autoencoder を使って候補となる類似文の生成を行った（図 4.2）．その後，候補文から敵対的な結果となる文を選択した．候補文の生成には以下の 2 つの手法を試した．

摂動の付加 (Perturbing) 中間表現に正規分布からサンプルした摂動を加えて生成を行った．文表現の学習をしているので，多少の変更を加えても意味的な変化が少ないと考えられる．摂動は Zhao らの手法を参考に，まず確率分布から N 個の摂動ベクトルをサンプルし，各摂動を中間表現に加えて復元しもし敵対的サンプルとして使えるものであったか，そうではなかったで分類した．敵対的サンプルとして使用できない摂動であった場合，破棄し新しい摂動をサンプルした．敵対的サンプルとして使用できる摂動であった場合は，最小の摂動を求めるために 2 分木探索を行った．すなわち，対象のモデルの出力が変化する摂動を半分にし，次の摂動として採用し，もしこの摂動を加えて生成されたサンプルが敵対的でなかった場合は，その摂動と元の摂動の中間を次の摂動とした．場合は摂動を保持し，変化しなかった場合は新しい摂動をサンプルをした．これを 1 回の試行として，B 回試行を繰り返した．摂動をサンプルする分布は平均 0, 分散 μ の正規分布を使用した．μ，N，B は実験ごとに設定した．

ノイズを加えた復元 (Denosing) DAE の学習によって入力中の足りない情報の復元が学習されているが，その際元の文が完全に復元出来るわけではない．例えば，“She serves us the meal.” という文から主語の “She” を取り除いて復元した場合，“Waiter serves us the meal.” のように動詞 “serves” にあう主語が学習によって復元されたが，“She” を取り除かなかった場合は同じ文が復元された．このようにノイズの加え方によって復元した時違う文となるので，ノイズつきで復元したものを候補文とした．ノイズの加え方は学習時と同じように，単語の順番の入替えと欠落によって行った．

また，Perturbation と Denoiseing とは別に，デコーダで復元時に多様な文を生成するために次の手法を試した．通常 Decoder の出力するトークンは出力層から最も高い値のものを採用するが，代わりに出力層の出力に Softmax をかけたものを確率分布として考え，そこからサンプリングを行った（**Sampling**）．これによって復元結果が途中から分岐され多様な文の生成を行うことが出来る．

つまり，以下の 4 つの方法で敵対的サンプルの候補文生成を行った．また，生成された文に未知語トークンが含まれていた場合は候補文として採用しなかった．

- Perturbing
- Perturbing + Sampling
- Denosing
- Denosing + Sampling

候補文の生成後，対象のモデルに入力し，出力が変化した場合は敵対的サンプルとして採用する．この時，複数の候補文から最も流暢なものを選択するために，yelp のレビューデータ及び wikipedia データで学習した 5-gram 言語モデルを用いて敵対的サンプルの perplexity と元の文の perplexity の比を計算して最も変化が小さいものを採用する．N-gram 言語モデルとして KenLM [33] を使用した．入力が複数の文からなる文章の場合，それぞれの文について敵対的サンプル候補を生成し，最も尤

度を下げるものを採用する．1 文の変更で目的のモデルの出力が変化しなかった場合は，1 文目の変更を固定して他の文の変更を行うことを繰り返した．

第5章　評価方法

この章では生成した敵対的サンプルを客観的及び主観的評価する方法について述べる．敵対的サンプル生成手法の評価に必要な項目として生成したサンプルの敵対性と元のサンプルの類似度がある．つまり，どれほど多くのサンプルに対して対象のモデルを騙す敵対的サンプルの生成が可能かなのかと人間が見た時の元のサンプルとの類似度やラベルの不変性を評価する必要がある．従来の敵対的サンプル生成手法の研究では，敵対的サンプルが生成出来たかのみ客観的評価で行うものが多く，ラベルの不変性などは人手で評価を行っていた．そこで，本研究では客観的評価が可能な尺度を用意し，主観的評価の結果と比較することによって敵対性と元のサンプルの類似度を測る新たな尺度を模索した．

5.1　評価尺度

5.1.1　攻撃成功割合（Attack rate）

Attack rate は対象のモデル（分類器）が正解したサンプルに対して敵対的サンプルを生成出来た（攻撃が成功した）割合である．Attack rate が高いほど，その生成手法は多くの入力に対して敵対的サンプルの生成が行えたということであり，生成手法の汎用性を確かめることが出来る．敵対的サンプルの生成モデルはテストデータ中の多くのサンプルに対して，敵対的サンプルの生成を行える方が良いため，この評価を行う．このスコアが低いということは，限定的なサンプルに対してしか攻撃を行うことが出来ないということである．

5.1.2 Perplexity

Perplexity を用いて，生成した敵対的サンプルの文としての自然さを評価する．Perplexity とは言語モデルの評価で使われる指標であり，文の不確かさを表す．言語モデルは単語の系列を与えた時にその次にくる単語を予測するモデルである．例えば，“This is a” を与えると，“a” の次に来るべき単語の確率分布を出力する（e.g. $P(\text{pen}|s) = 0.2$）．この時，各単語の確率の乗数の逆数が Perplexity である（$\prod_i^N P(w_i)^{-1/N}$）．Perplexity は確率が高いほど低くなり，自然文を入力として入れた時に低い Perplexity を出力する言語モデルほど良い．逆に不自然な文を入力として与えると Perplexity が高くなる．そのため，言語モデルを固定した時，Perplexity によって文の自然さを数値化することが出来るので，これを使って生成された敵対的サンプルの自然さの評価を行う．

5.1.3 敵対的学習

敵対的サンプルのラベル不変性を確認するために，訓練データに敵対的サンプルを追加して学習を行った．敵対的サンプルの人間からみたラベルが変化してしまっていた場合，敵対的サンプルを加えて学習を行なわれることによって，加えない時と比べてモデルの精度が低下するはずである．また，ラベルが変わっていない敵対的サンプルが生成出来ていた場合は，学習データが増えたと考えることが出来るので，モデルの汎化性能が向上するはずである [12]．

5.1.4 共通単語数

単語の入替えしか行わない手法と比べて生成モデルを使った手法は生成される敵対的サンプルの自由度が高いため，元の文と似た文となっている保証がない．そこで，共通する単語の数を数え，生成した敵対サンプルと元の文の類似度を測定する．この時文の意味に関わる単語（内容語）のみを対象とし冠詞など意味に直接かかわらない単語（機能語）を取り除いた．元の文及びその敵対的サンプルの各単語を品

詞タグ付けを行い，名詞・形容詞・副詞・動詞・数詞・代名詞に分類した．品詞タグ付けにはStanfordCoreNLP [34] を使用し，以下のタグは纏めて1つのタグとしたし，他のタグの単語は無視した．この時，生成モデルの出力はすべて小文字になっているので，StanfordCoreNLP の TrueCaseAnnotator を使い補正した[1]．

名詞　普通名詞（NN, NNS），固有名詞（NN, NNP）

形容詞　形容詞（JJ），比較級形容詞（JJR），最上級形容詞（JJS）

副詞　副詞（RB），比較級副詞（RBR），最上級副詞（RBS）

動詞　一般動詞（VB, VBD, VBG, VBN, VBP, VBZ）

数詞　数詞（CD）

代名詞　代名詞（PRP），指示代名詞（PRP$），Wh-代名詞（WP），Wh-所有格代名詞（WP$）

次に，単語の時制等による変化を吸収するために見出し語化を行った．また，1文に複数回同じ単語が出現する場合も1つして数え，単語の集合として扱った．一連の作業により，元の文と敵対的サンプルそれぞれの内容語の見出しの集合が手に入ったので，共通する単語数を数え，適合率と再現率の計算を行った．

$$\text{適合率} = \frac{\{\,\text{共通する内容語}\,\}}{(\{\,\text{共通する内容語}\,\} + \{\,\text{敵対的サンプルにのみ出現した単語}\,\})}$$

$$\text{再現率} = \frac{\{\,\text{共通する内容語}\,\}}{(\{\,\text{共通する内容語}\,\} + \{\,\text{元の文にのみ出現した単語}\,\})}$$

敵対的サンプルを生成するために変更を加えなければならないので，適合率や再現率が100 %となることはないが少ない変更によって作られる方がよい共通する単

[1]Caseless models: https://stanfordnlp.github.io/CoreNLP/caseless.html

語数は元の文と敵対的サンプルの距離として考えることができ，高い値で敵対的サンプルの生成が出来た方が良いといえる．

5.1.5 主観的評価

今までの自動評価指標で本当に元とラベルが変わっていないのか保証できない．また，元の文と意味的に似ているのか，文として自然なのかについても人間が判断した場合と本当に相関があるのか分からない．そのため，人手で敵対的サンプルの評価を行った．関連研究では，文法的に正しいかを 2 値で分類させたり，2 つの敵対的サンプルを与えてどちらがより元の文に近いかアノテートしたもの [1] と，敵対的サンプルのラベルをアノテートさせ真のラベルと比較し，元の文との類似度を 4 段階のスコアを付けさせるもの [16] がある．

本研究では，次の 3 つについて人による評価を行った．

Fluency Fluency は文の流暢度を表すもので，機械翻訳の人手評価で使われている指標である．一般的に 5 段階で評価するが，今回は文法的に破綻していないかを確かめることが目的であるため，「理解不能（0)」，「流暢ではない（1)」，「流暢（2)」の 3 段階に分類した．アノテータには，文法的に破綻していて文の意味が理解できないものを「0」，文意は分かるものの不自然な所があるものを「1」，人が書いたと思えるものは「2」に，とするように説明した．Zhao らは敵対的サンプルが「文法的か」どうかで 2 値分類したが，元の文が口語的であるためもともと文法的に正しくない可能性を考慮して 3 値分類で行った．

Label 元の文と敵対的サンプルのラベルが変化していないか確かめるために，ラベルのアノテートを行った．バイアスが入らないように，敵対的サンプルのみを与え元の文は与えずにアノテートを行った．実際に行った極性分類タスクでは 2 値分類だが，生成の結果極性がなくなる可能性があるため，「中立」を追加した．このアノテーションは Fluency が 1（「流暢でない」）以上になったものを対象に行った．

表 5.1: 客観的評価と主観的評価の比較

評価項目	客観的評価	主観的評価
敵対性	攻撃成功率	-
言語としての自然さ	Perplexity	Fluency
ラベル不変性	敵対的学習	Label
原文との類似度	共通単語数	Similarity

Similarity 元の文との類似度を測るために，既存手法並びに提案手法によって生成された敵対的サンプルを与え，元の文に意味的により近い敵対的サンプルを選択させた．元の文とラベルが変化している場合は，違いが大きいとした．

アノテーションは Fluency，Label，Similarity の順で行った．

客観的評価と主観的評価の対応関係は表 5.1．“言語との自然さ” は敵対的サンプルの要件ではないが，モデルの理解を行う上で入力空間内で精度が悪化する問題点を把握したいため，入力空間から外れていないという特性が必要となる場合がある．これらの評価項目が客観的評価によって測定可能になれば，コストのかかる主観評価を行わずに敵対的サンプル生成手法の評価を行うことが出来るようになる．

5.2 評価タスク

敵対的サンプルを生成する対象となるタスクとして極性分類タスクを採用した．

極性分類タスクとはあるテキストを与えらた時にそのテキストの極性（ネガティブ，ポジテイブ）を分類するタスクである．このタスクに対する敵対的サンプルは，元の文と逆の極性（ネガティブならポジティブ）を対象の分類器が出力してしまうものである．このタスクを選んだ理由は 2 値分類問題でありながら，トピック分類と比べて高度な意味理解を必要とする複雑な問題であるからである．

攻撃の成功は，騙す対象のモデルが元のサンプルとは逆のラベルを出力した時（ネガティブ⇔ポジティブ）であり，出力の信頼度などは考慮しない．つまり 2 値分類

なので softmax 層の偽のラベルの次元の値が 0.5 を超えれば良いとした．攻撃成功をある閾値を超えるまでとする場合もあるが，恣意的な判断が入ると思われるので今回は採用しなかった．

第6章　評価実験

6.1　実験設定

6.1.1　極性分類タスク

本実験ではIMDB映画レビューデータ [35] とSemEval2016のTask5のレビューデータ [36] の2つの極性の注釈が付けられたデータセットを使用した．SemEval2016のTask5にはいくつかのトピックに分かれたデータが存在し各サンプルはnegative，positive，neutralの3値分類されているが，今回はレストランレビューデータからnegativeもしくはpositivieなものだけ抜き出し使用した．各データセットの文章数,1文章あたりの文数，1文あたりの単語数を表6.1に示す．SemEvalはすべて単文であり，1文あたりの単語数も少ないことが分かる．また，SemEvalは訓練データが存在しないため，レストランレビューを分類するモデルの訓練には同じレストランレビューデータであるYelpレビューデータ [37] を使用した．

2つのデータセットをもとに分類モデルの学習を行った．Yelpレビューデータは極性情報を持たず，5段階評価の数値のみ存在するので，1〜2をnegative，4〜5をpositiveとして3は使用しなかった．IMDBでは訓練データから15％を分け， 訓練データは21,249文章，開発データは3749文章とし，Yelpでは訓練データから10000文抜き出して開発データとした．前処理として訓練，開発，テストデータの全てのテキストを小文字化し，単語分割した．訓練データから語彙を構築し，出現頻度上位5万単語を使用した．TextCNN [38] のフィルターの幅を3,4,5とし，各フィルターのチャネル数は128，単語埋め込み表現の次元 d は300とし，fasttext [39] の公開さ

表 6.1: 極性分類タスクのデータセット

	IMDB	SemEval	Yelp
# of train samplings	25k	0	5.3M
# of test samplings	25k	1554	10k
avg. sentences / sampling	12.3	1.0	8.29
avg. words / sentence	21.5	13.8	22.50

表 6.2: 分類モデルの各データセットでの性能

	IMDB		SemEval	Yelp	
	dev	test	-	dev	test
CNNText	0.887	0.876	0.891	0.972	0.967

れている埋め込み表現[1]を初期値として使用した．fasttext に存在しなかった語彙は，一様分布 $\mathcal{U}(-1/\sqrt{d}, 1/\sqrt{d})$ で初期化した．最適化手法は AMSGrad [31] で，学習率は 0.001, β_1 及び β_2 はそれぞれ 0.9, 0.999 とした．最適化の結果を両方のデータにおいて F 値を約 0.9 出すことが出来た（表 6.2）．

6.1.2 敵対的サンプルの生成

テストデータのうち訓練した TextCNN が正しく分類出来たサンプルからランダムに 500 件抽出し，それぞれの手法で敵対的サンプルの生成を行った．極性分類タスクでは，モデルの出力が真逆の極性（元がネガティブならポジティブ）に分類されるように敵対的サンプルの生成を行った．提案手法及び，ARAE で敵対的サンプルの生成を行う時，各試行で加える摂動の数 $N = 200$，最大試行回数 $B = 15$ の制限をかけた．摂動を加える方法では摂動の大きさ σ を $0.2, 0.5, 1.0$ を試した．Alzantot らの手法は彼らの論文に従い，単語入替えの最大試行回数 $G = 20$，各試行における個体数 $S = 60$，入替え単語候補数 $N = 8$ として生成を行った．

[1] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

表 6.3: 各データセットでの 5-gram 言語モデルの精度

	IMDB	SemEval
Perplexity	214.06	60.15

6.1.3 N-gram 言語モデル

Perplexity を測定するために使用する言語モデルとして N-gram 言語モデルを使用した, yelp のレビューデータ及び wikipedia データ[2]から抽出したそれぞれ 68,719,847, 48,912,182 文を用いて学習した．実装は KenLM を使用し，N=5 で行った．データセット内の文の Perplexity を測定した結果を表 6.3 に示す．IMDB ではテストセットを，SemEval は全てのデータを使用して測定をした．同じレストランレビューのデータが含まれているため，SemEval の Perplexity が小さくなっていることが分かる．N-gram 言語モデルでは，特定の N-gram が訓練データに存在するかどうかにより大きく精度が変わってしまうことから，IMDB では人名や映画タイトルなど Yelp には含まれていない単語などが存在することが Perplexity の乖離につながったと考えられる．

IMDB のような複数の文からなるデータの敵対的サンプルの Perplexity を測る時は, 変更した文のみを対象とし, 元の文から変更されていないものは計算にいれなかった．生成ベースでは 1 サンプルあたりたかだか 3 文程度である．しかし，Alzantot らの手法では変更した文が多岐に渡るため，全ての文を計算対象にいれた．

6.1.4 主観的評価設定

5.1.5 節の主観評価を行うために，提案手法（perturbing, perturbing+sampling, denoising, denoising+sampling）と ARAE，単語置き換え（Genetic）全ての手法で敵対的サンプルの生成に成功したサンプルを抜き出した．抜き出したサンプルのうちランダムに 50 件選択し，各手法によって生成された敵対的サンプルと元の文計 300

[2]https://github.com/jack-and-rozz/wikipedia-scripts

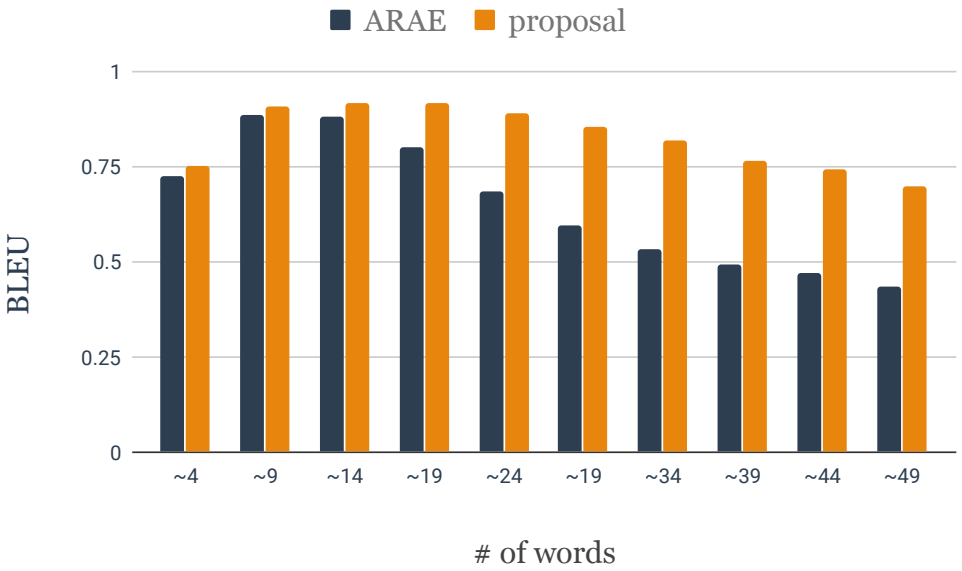

図 6.1: 入力文長と提案手法と ARAE の自己符号化器としての精度の関係

件アノテーションを行った．1 サンプルあたり 3 人アノテートを行った．Fluency 及び Label の評価を行う時は，アノテータは順番が混ぜられた敵対的サンプルのみを見て行うようにし，元が同じサンプルが連続しないようにした．

6.2 実験結果

6.2.1 長文の生成

ARAE の課題であった長文生成が改善されたかを確認するために，自己符号化器としての精度を測定した．図 6.1 から全ての文長において提案手法の方が高いスコアとなることが分かる．ARAE の精度が文長とともに大きく低下する理由は，潜在空間へ変換し元の中間表現への復元を行っているため，情報の保存が難しく復元時に大きく変わってしまうからだと考えられる．提案手法では，中間表現をそのままデコーダへ入力を行うので高い復元性能を誇ることが出来ている．

表 6.4: 各敵対的サンプル生成モデルの攻撃成功率と平均変更文数

Model	Attack Rate		Avg # of changes	
	SemEval	IMDB	SemEval	IMDB
perturbing ($\sigma = 0.2$)	0.348	0.952	1	1.66
perturbing ($\sigma = 0.5$)	0.934	1.000	1	1.29
perturbing ($\sigma = 0.2$) + sampling	0.765	0.991	1	1.33
perturbing ($\sigma = 0.5$) + sampling	0.982	1.000	1	1.23
denoising	0.835	0.998	1	1.70
denoising + sampling	0.991	0.997	1	1.05
ARAE [1]	0.998	0.988	1	1.57
Genetic [16]	0.683	0.850	–	–

6.2.2 Attack Rate

各手法で敵対的サンプルの生成を行った時の攻撃成功率を表6.4に示す．SemEvalはすべて単文のレビュー文に対する敵対的サンプルである．SemEvalの結果を見ると，摂動を加える手法ではσを大きくすることで，成功した数が増えていることが分かる．また，破壊的変更を加える手法でも高い割合で攻撃が成功している．どちらの方法でもデコード時に最大尤度のトークンを選択するのではなく確率的に選択を行うことで，攻撃成功率が高くなっていることが分かる．生成モデルを使っているARAEも高いスコアをだしている．一方で単語置き換えによる手法（Genetic）は他の手法と比べ，低いスコアとなっている．これは単文ではそもそも置き換えが可能な単語が少なく，候補となる文の生成が難しいためだと考えられる．

IMDBデータ訓練したモデルに対する攻撃は全てのモデルでSemEvalの時と比べ高くなっている．特に生成ベースの手法はほぼ全てのサンプルに対して敵対的サンプルの生成に成功している．

IMDBデータではSemEvalと違い文章の敵対的サンプルであり，文章中の複数の文に対して敵対的な文の生成を行うことが出来るため，変更可能な箇所が多いこと

表 6.5: N-gram 言語モデルにおける各敵対的サンプルの Perplexity

Model	Perplexity	
	SemEval	IMDB
perturbing ($\sigma = 0.2$)	64.56	185.58
perturbing ($\sigma = 0.5$)	76.96	178.02
perturbing ($\sigma = 0.2$) + sampling	74.29	312.64
perturbing ($\sigma = 0.5$) + sampling	62.14	489.63
denoising	97.64	213.12
denoising + sampling	177.33	376.17
ARAE	151.34	280.47
Genetic	814.13	658.63

が理由として考えられる．1 つの敵対的サンプルを生成するために変更した文数を見ると，SemEval では攻撃成功率が低かったものほど多くの変更を行っていることが分かる．攻撃成功率の高さは訓練したモデルのデータによる違いがあることも影響している．SemEval の実験では Yelp により訓練されたモデルを使用しているため，IMDB より多くのデータを使用して訓練がされている（表 6.1）．in-domain な評価では 0.967 という高い F 値をだし，SemEval を使った評価でも 0.89 と IMDB で訓練し IMDB で評価したモデルと遜色ない性能示している．IMDB で訓練されたモデルと比べ汎化性能が高くなっていると考えられる．Alzantot らの手法では，IMDB の方が置き換えの候補となる単語数が増え，またサンプルの単語数が増えることによって変更してもよい単語数が増えることが攻撃確率の向上につながっている．IMDB の各サンプルの平均文数が 12 あるが，たかだか 2 文変更を加えることで出力が変化してしまうので，深層学習モデルの感度の高さが確かめられた．

6.2.3 敵対的サンプルの Perplexity

次に，各敵対的サンプルの N-gram 言語モデルで測った Perplexity を表 6.5 に示す．表 6.3 で分かったように，元の文の PPL が違うため，データセット間の比較を行うことは難しいが，データセット内での比較を行うことは可能である．たたし，Genetic の IMDB だけ全文を使用しているため，比較は出来ない．どちらのデータセットでも摂動を使った手法が最も Perplexity が低くなった．また，摂動の大きさによる Perplexity の変化は攻撃成功率の上昇幅に比べて大きくない．ノイズを加える手法に関しては，Perplexity が摂動を加えた場合と比べて高くなっている．一方 sampling では SemEval での “perturbing + sampling” を除き，Perplexity が高くなっている．デコード時に出力単語を確率的に選択しているため，Perplexity が高くなることは予想がつくが，大きく高くなっていることが分かる．“perturbing + sampling” と “denoising + sampling” で生成された文を見比べてみると，後者の方が同じ入力文でも非文になっているものが多く存在した．これは “denoising + sampling” では破壊的変更が加えられた不完全な文が入力されるため，復元時に候補となりうる単語が多く存在するため，確率分布に従って選択をした時のばらつきが大きくなると予想される．Zhao らの手法（ARAE）も生成ベースなので，SemEval ＜ IMDB の傾向にあるが，SemEval での Perplexity が大きく離れている．これは，ARAE では生成した敵対的サンプルから最も良いものを選択する時に一番摂動が小さいものを選ぶからである．一方で，提案手法では Perplexity が低いものを優先的に選択するため，公平な比較にはなっていない．Alzantot らの手法（Genetic）が一番高い Perplexity となっている．類義語による単語の入替えなので，一番変動は少ない．これは，類義語の選択を外部リソース（Glove）を用いて選択しているため，データセットにない単語の選択が可能なためである．表 6.6 に生成された敵対的サンプル中で，対象のモデルでは未知語なる単語の数と割合に示す．Genetic によって生成されたものは元の文と比べて多くの未知語を含んでいることが分かる．Alzantot らの手法は対象のモデルで未知語になるように単語の入替えを行うことで，対象のモデルを騙す敵対的サンプルの生成を行っていると考えられる．

表 6.6: 各サンプル中の未知語の割合

	# of samplings	# of words	unknown / word ratio
perturbing ($\sigma = 0.5$)	442	6754	0.0022
ARAE	442	5831	0.0027
Genetic	442	3271	0.0706
original	1554	19287	0.0286

表 6.7: 敵対的学習によるモデル精度の変化．“No Adversarial” は敵対的サンプルを加え無かった場合の精度で，この精度との差を Δ とした．

Model	With original		Only adversarial	
	F-score	Δ	F-score	Δ
No Adversarial	0.799	-	0.799	-
perturbing	0.747	-0.052	0.574	-0.225
perturbing + sampling	0.763	-0.036	0.581	-0.218
denoising	0.740	-0.059	0.519	-0.28
denoising + sampling	0.749	-0.05	0.567	-0.232
ARAE	0.748	-0.051	0.525	-0.274
Genetic	0.769	-0.03	0.669	-0.13

6.2.4 敵対的学習

IMDB の訓練データからランダムに抽出した 2000 件に対して各手法で敵対的サンプルの生成し，生成した敵対的サンプルを訓練データに加えて学習を行い，テストセットでモデルの精度（F 値）を測った（表 6.7）．SemEval は訓練データが存在しないため，IMDB でのみ実験を行った．IMDB の訓練データは約 21k サンプル存在するので，2000 件加えたところで大きく影響しないと考えられるので，敵対的サンプルの元となった 2000 件と敵対的サンプルを加えた 4000 件と敵対的サンプルのみの 2000 件，それぞれで学習した．

表 6.8: SemEval データにおける敵対サンプルと元の文間で共通する内容語の割合

POS		perturbing	perturbing +sampling	denoising	denoising +sampling	ARAE [1]	Genetic [16]
NN	r	0.738	0.585	0.566	0.521	0.442	0.730
	p	0.710	0.506	0.775	0.566	0.456	0.701
ADJ	r	0.635	0.479	0.331	0.300	0.349	0.409
	p	0.559	0.405	0.549	0.315	0.388	0.404
RB	r	0.814	0.694	0.553	0.619	0.565	0.637
	p	0.536	0.443	0.789	0.632	0.532	0.645
V	r	0.833	0.739	0.609	0.591	0.645	0.778
	p	0.723	0.592	0.776	0.638	0.560	0.736
CD	r	0.720	0.620	0.580	0.460	0.479	0.794
	p	0.667	0.534	0.852	0.621	0.410	0.870
PRP	r	0.916	0.889	0.764	0.768	0.828	0.939
	p	0.771	0.639	0.885	0.848	0.689	0.965
ALL	r	0.765	0.641	0.546	0.527	0.520	0.688
	p	0.660	0.511	0.753	0.566	0.503	0.671

すべての手法で，敵対的サンプルを加えた場合精度が悪化していることが分かる．元のサンプルを加えた場合は大きくは悪化しないが，元のサンプルを加えなかった場合は Genetic 以外は大きくスコアを落としている．特に denoising により生成された敵対的サンプルはほぼ学習が出来ていないため，生成されたサンプルの真のラベルが変化していると考えられる．

6.2.5 共通単語数

元の文と敵対的サンプルの類似度を測るために，共通する内容語の数を測定した．その結果を SemEval データを使って測定した時の結果を表 6.8 に示す．摂動の σ は

0.5 とした．再現率（r）は元の文に含まれているある品詞の単語のうち，敵対的サンプルと共通して出現した割合で，適合率（p）は敵対的サンプルに出現したある品詞の単語のうち，元の文と共通している割合を示している．つまり，再現率が高いほど，元の文で現れる単語を網羅しており，適合率が高いほど元の文から追加の単語を増やしていないことになる．ALL は全ての品詞の単語を合わせた結果である．各手法で敵対的サンプルを生成する際にどのように変更を加えているのかが分かる．摂動を加える手法は再現率が適合率より高くなっており，逆にノイズを加える手法では適合率が再現率より高くなっている．このことから，摂動を加える手法は元の文に追加することにより敵対的サンプルの生成を行い，ノイズを加える方法は元の文から単語を削除することによって敵対的サンプルを行っていると考えられる．ノイズによって元の文から単語等の情報が削除され元の全情報がないため，ノイズを加える手法はこのようになっているのだと考えられる．また，どちらでも sampling を行うことで，共通する単語は減少している．これは共通単語数を見出し語化した単語の完全一致でしか測っていないので，sampling でたとえ似た単語を選んだとしても，共通すると考えられないからである．ARAE は提案手法と比べて全体的に低くなっている．これは ARAE が摂動を加えるときに，エンコーダとデコーダの間で 2 回多層パーセプトロンによる変換が入っているため，元の文の情報を完全に保つことが難しいためである．Genetic は適合率と再現率がほぼ同じ値になっている．単語の置き換えによって行っているため，1 単語消す操作と 1 単語を加える操作を行っていることに等しいと考えることが出来る．

品詞ごとにみると，変更量の多いものは形容詞（ADJ）で，変更が少ないのは代名詞であった（PRP）．極性分類タスクが対象であるため，形容詞の変更が大きく結果に影響するのだと考えられる．形容詞の変化量の大小が元の文と極性が変化している割合と関係あることが予想される．これはユーザー調査によって確かめる必要がある．

IMDB データで同様の実験を行った結果を表 6.9 に示す．SemEval の時と違い，文章の敵対的サンプルであるため，内容語の比較は変更を加えた文のみを対象に行った．提案手法，ARAE ともに傾向は変わらないも大きくスコアが落ちている．これ

表 6.9: IMDB データにおける敵対サンプルと元の文間で共通する内容語の割合

POS		perturbing	perturbing +sampling	denoising	denoising +sampling	ARAE [1]	Genetic [16]
NN	r	0.290	0.224	0.432	0.321	0.199	0.870
	p	0.262	0.167	0.631	0.378	0.240	0.867
ADJ	r	0.285	0.214	0.361	0.256	0.218	0.787
	p	0.212	0.159	0.481	0.277	0.239	0.751
RB	r	0.494	0.455	0.547	0.504	0.393	0.843
	p	0.336	0.299	0.731	0.570	0.463	0.859
V	r	0.503	0.476	0.568	0.520	0.451	0.916
	p	0.439	0.361	0.743	0.586	0.493	0.881
CD	r	0.354	0.372	0.429	0.312	0.348	0.895
	p	0.249	0.182	0.530	0.222	0.227	0.823
PRP	r	0.778	0.725	0.738	0.696	0.667	0.988
	p	0.549	0.476	0.847	0.726	0.709	0.984
ALL	r	0.402	0.349	0.486	0.405	0.330	0.868
	p	0.327	0.251	0.658	0.450	0.371	0.852

は敵対的サンプルを生成する時に 1 文ごとに生成を行い，尤度を最も下げるものを採用するが，他の文があることにより 1 文の変更によって出力の変更が出来ない場合が多い．そのため，摂動の縮小を行うことが出来ず，結果大きく変化した文が敵対的サンプルに含まれる．複数文変更を逐次的に行っているため起きる問題で，借りに各文について候補文を作成し，それぞれの組み合わせによって敵対的サンプルを生成する場合，より元の文に近い敵対的サンプルの生成が出来るかもしれないが，より計算コストがかかるという問題がある．

表 6.10: Inter-annotator agreement

Annotator	κ	
	Binnary distance	Identify fluent & disfluent
1,2,3	0.245	0.589
1,4,5	0.334	0.659

6.2.6 主観的評価結果

表 6.10 に Fluency におけるアノテータ間の合意関係を表す．指標として，一致度の尺度として良く使われる Fleiss の κ [40] を使用した．κ は，アノテータの合意関係にランダムに選択した時に偶然一致する確率で補正したものである．アノテータ 2 ～ 5 は 150 件ずつしかアノテートしていないため，κ を 150 件ごとに算出した．Fluency は $0, 1, 2$ の三段階評価であったため，合意決定はを完全一致した場合（Binnary distance）と disfluent と fluent は区別しない場合（Incomprehensible distance）の 2 つ用意した．これは，アノテータが非英語圏で英文の流暢度判定にバラツキが生じると考えたためである．Landis と Koch らによる [41] と，$\kappa > 0.2$ で “fair”，$\kappa > 0.4$ で “moderate”，$\kappa > 0.6$ で “substantial” であると述べているので，Fluency の一致度は Binnary distance で公平で，Fluent と Disfluent を区別しない場合は高い一致度であると言える．

各アノテータにサンプルの流暢度を 3 段階で評価してもらい，過半数以上の合意が取れたものをそのサンプルの流暢度とした．この時，3 人のアノテータで意見が割れたもの（“Incomprehensible”，“Disfluent”，“Fluent” にそれぞれ 1 票ずつ入った時）は合意が取れていないものとして，別にした．その結果を集計したものを表 6.11 に示す．全体サンプル数に対する理解可能な文の割合を次式で計算した．

$$\text{Fluent ratio} = \frac{\text{“Fluent”} + \text{“Disfluent”}}{\text{“All samplings”} - \text{“Disagree”}}$$

変更を加えていないオリジナルの文の評価を見ると，約 8 割以上の文が「流暢」も

表 6.11: 各敵対的サンプルの Fluency アノテーション

	Incomprehensible	Disfluent	Fluent	Disagree	Fluent ratio
perturbing	11	19	16	4	0.761
perturbing + sampling	14	11	23	2	0.708
denoising	15	14	13	8	0.643
denoising + sampling	22	11	11	6	0.500
ARAE	20	18	10	2	0.583
Genetic	13	16	8	7	0.649
original	2	8	35	5	0.956

しくは「流暢でない」に分類されている．しかし，合意でないサンプルも多く，また 2 例「理解不能」に分類されている．合意が取れなかった例として次のような，複数の文がつながってしまっているものや，文の途中で切り出したようなものであった．

> the entertainment was great they have shows that go on through out the dinner.
> but $ 500 for a dinner for two that did't include wine?

「理解不能」に分類された 2 例は，文法的におかしいものと，英語以外の言語が混じっていた．

> a thai restaurant out of rice during dinner? la rosa waltzes in, and i think they are doing it the best.

生成された敵対的サンプルの中では Fluent ratio は摂動のみを加えたモデルが最も高く，次に入力にノイズを加えたものが高かった．どちらも sampling によるデコードを行うと，不可解な文が増えてしまっているが，入力にノイズを加えた時の方が顕著に増えている．むしろ，摂動+sampling の時だけ「流暢」なサンプルが増えている．表 6.4 から分かるようにノイズを加えるだけでは敵対的サンプルの生成が難しい．しかし，敵対的サンプルの生成にあたり，対象のモデルを騙せたサンプ

表 6.12: 各敵対的サンプルの Label アノテーション

	# of samplings	same	oppsite	neutral
perturbing ($\sigma = 0.5$)	39	0.49	0.28	0.18
perturbing ($\sigma = 0.5$) + sampling	36	0.33	0.44	0.17
denoising	35	0.20	0.40	0.31
denoising + sampling	28	0.11	0.68	0.18
ARAE	30	0.43	0.20	0.33
Genetic	31	0.84	0.07	0.03
original	48	0.92	0.04	0.04

ルを優先的に選択するため，sampling により生成された非文で騙せた時，その文が敵対的サンプルに採用されてしまっているのだと考えられる．そのため，表 6.5 で denoising+sampling だけ高い Perplexity になっている．一方で，sampling によりデコードされた文多様性が高いため，もともと攻撃成功率が高い摂動を加える方法では攻撃に成功したものの中から Perplexity が最も低いものを選択することができ，表 6.5 で Perplexity が下がっている．

既存手法である ARAE に関しては，「理解不能」に分類されているものが多く，また多くが「流暢でない」とされていることから，Perplexity の高さから理解出来る結果である．

次に，Fluency が「流暢でない」，「流暢」と合意が取れていないものについて，極性ラベルをアノテートした結果，元のラベルと変化しているかを集計した（表 6.12）．オリジナルの文での精度は 0.92 で，中立に分類されたものも含めると 0.96 と十分に高い数値になっている．一方で摂動のみを加えた敵対的サンプルがラベルが変化しない割合が最も高かったが，各手法で生成された敵対的サンプルは同じラベルになっているものは 5 割を切っていた．特にノイズを加える手法は逆のラベルに変化しているものが多かった．これは形容詞など極性を決定する上で重要な単語が欠落した状態で，逆の極性になるように復元可能なためである．摂動を加える方法はエンコード時は全ての入力情報を持っているため，抑制されてはいるが，ARAE より

表 6.13: 摂動による敵対的サンプルと原文の類似度

κ	similarity
0.6986	0.73

逆のラベルになっている割合が高い．しかし，これは摂動による手法が ARAE より劣っているというわけではなく，表 6.11 の結果から分かるように提案手法の方が自然な文の生成に成功しており，50 サンプル中同じラベルのサンプルを生成出来た数は ARAE より多い．また，sampling を行うことで，逆の極性になる確率が高くなっている．これは，sampling によって文脈的に適切な似た単語が選択される時，極性が変化するような単語が選択されている．

最後に，摂動による敵対的サンプルと ARAE による敵対的サンプルのどちらがより元の文に近いかの測定を行った．前述のラベルアノテートの結果，両手法とも真のラベルと同一もしくは中立となったサンプルでのみ評価に使用した．その結果，11 件のサンプルに関して 3 名にアノテートしてもらい，過半数を超えた方をより近いとして集計した結果を表 6.13 に示す．Fleiss の κ は高い値になっており，アノテータ間で合意が十分に取れていると考えられる．類似度は，全体数のうち摂動による敵対的サンプルがより原文に近いと判定された割合である．ARAE によって生成された敵対的サンプルと比べてより原文に近い文が生成出来ていることが分かる．これは，表 6.8 の結果からも分かるように共通単語が多く存在するためだと考えられる．

最後に，生成した結果について見比べてみる（表 6.14）．提案手法は perturbing の場合のみ載せている．元の文が長いと，ARAE では情報の欠落や後半部分で破綻していることが分かる．一方で，提案手法では概ね近い文が生成出来ている．単語置き換え手法は類義語による置き換えなので，元と近い文が生成されているが，置き換えた結果品詞が変わっていたりし，違和感のある文になっている．

表 6.14: 敵対的サンプル例

	1486041:2
original	Despite a slightly limited menu, everything prepared is done to perfection, ultra fresh and a work of food art.
perturbing	despite a slightly limited menu , everything is prepared to be done properly , and to do a mediocre care of art work .
ARAE	despite a limited menu slightly everything is done here , nothing to fresh and a chef of work .
Genetic	however a faintly scant menu entire prepared is done to faultless ultra fresh and a collaborating of food art

表 6.15: 各評価項目における各手法の比較

モデル	敵対性	自然さ		ラベル不変性		類似度	
		客観	主観	客観	主観	客観	主観
提案手法（摂動）	○	○	○	×	△	○	○
ARAE	○	△	×	×	×	×	△
Genetic	△	×	△	△	○	○	–

6.3 まとめ

本研究で評価項目とした敵対性，言語としての自然さ，ラベル不変性，原文との類似度について客観的評価及び主観的評価を行った結果を各手法で比較し表 6.15 にまとめた．Perplexity で測定した言語としての自然さは，生成モデルベースの手法である提案手応と ARAE においては主観的評価と似た傾向を示したが，単語入替えを行う Genetic では全く一致しなかった．これは未知語により，言語モデルが正しく評価出来なかったためであると考えられる．そのため，未知語に対処するためにより大規模なコーパスを使うか，単語埋め込み表現を扱えるニューラルネットを使ったモデルなどを使うことにより解決が可能だと予想される．ラベル不変性は，敵対的学習の実験結果から生成ベースの手法は単語置き換えの手法と比べて大きく精度が下が

ることが分かった．主観的評価により，提案手法で生成されたサンプルのラベルの変化率を見ると半分以上のサンプルが元と違うラベルになっており，一方で単語置き換えでは 8 割が元のラベルのままであった．敵対的学習の実験を行うことでラベル不変性の評価が可能であると考えられる．しかし，同じラベルのサンプルが約半分あった perturbing と，約 9 割のサンプルのラベルが変化している denoising+sampling で敵対的学習の性能差が無かったことから，ラベルノイズが多い時の差は明確ではないことが予想される．ラベルノイズの量とモデルの精度の関係を調べる必要がある．また，不自然な文を学習データに加えた時に敵対的学習に同様な影響が出るのか，今回考慮してしなかったため，不自然な文が取り除かれた主観的評価と完璧な比較は困難になっている．主観的評価では disfluent 以上のものしかアノテートしなかったため，denoising+sampling では約半分以上のサンプルが主観的評価から省かれしまっており，これの敵対的学習への影響を考える必要がある．また，これらの比較を正確に行うにあたって統計的検定を行う必要があるが，これは今後の課題とする．

第7章　おわりに

本研究では，自然言語分野における敵対的サンプルの生成に取り組んだ．単語の入替えによる手法ではなく，生成モデルを使用することにより，より多くの入力テキストに対して敵対的サンプルの生成が可能になると考え，また従来の生成ベースで出来なかった長文生成と文の意味を考慮するために，雑音除去事項符号化器に意味表現学習を加えたモデルによるた敵対的サンプル生成手法を提案した．また，敵対的サンプルの生成を行う際に 4 つの摂動付加の手法を考え，比較した．生成した敵対的サンプルの評価方法について，広く使われる方法がなかったため，評価尺度の整理を行った．客観的評価と主観的評価の両方を行い，客観的評価では実際に対象モデルを騙せた割合，文の自然さ，敵対的学習によるモデル精度の変化，共通単語数による原文との類似度を測定し，主観的評価では生成した敵対的サンプルの流暢さ，原文とラベルが変化していないかの評価を行った．

提案手法は原文と表層的な類似度が高く，流暢度の高い敵対的サンプルの生成が可能であることを確認したが，主観的評価によってラベルが元の文から変化している例が多く存在した．そのため，敵対的サンプルとして使えないものも多く，またそれを客観的評価によって明らかにすることが出来なかった．

今後の課題は，客観的評価と主観的評価の関係性が解析し，敵対的サンプルの評価を定量的に行えるようにする必要がある．そのためには，主観的評価によって得られた情報を精査し，ラベルの変化したかを自動的識別可能な手法を考える．騙す対象のタスクによって「ラベルの変化」は変動するが，今回実験で使用した極性分類問題に関しては単語分散表現など意味を考慮した単語情報を用いて文の類似度を測るという方法が考えられる．また，2 割の生成文のうち理解不能な文であったた

め，生成された文のラベルが変化せず，また摂動を加えた時に文法的に破綻しないように制約をかけるより良い手法を考える必要がある．これは生成に構文情報を明示的に学習するモデル [42] や生成時の内容をコントロールする生成モデル [43] を参考に検討を行いたい．

謝辞

弊研究室に配属されてからの２年間本当に多くの人にお世話になりました．

まず吉永直樹准教授に感謝申し上げます．喜連川研究室に所属しているにもかかわらず，言語処理分野の研究を行うことを決めたため，指導教員でないにもかかわらず沢山の指導や助言をして頂き大変助けられました．お忙しいなか論文資料，スライド，発表練習を丁寧に見て指導して頂きありがとうございます．また，研究以外でも美味しい店やビールを教えて頂いたり，ボードゲームや登山などを一緒にして頂きありがとうございました．おかげで２年間楽しく過ごすことが出来ました．

次に，豊田正史教授に感謝申し上げます．研究室にいる時に詰まっていたりすると見に来てくれて，助言や相談に乗って頂きありがとうございました．また，未熟ながらサーバの管理の一部を任せ貰ったことは心から感謝いたします．おかげで，サーバに関して詳しくなり，実験をストレスなく行うことが出来ました．

喜連川優教授には素晴らしい研究室環境を用意していただき感謝申し上げます．喜連川教授の正鵠を射た指摘は耳が痛いながら，深く考えることの重要性に気づかせられました．

次に研究室の先輩方である，佐藤文一さん・石渡祥之佑さん・金洪善さん・佐藤翔悦さん・赤崎智さん・澤田頌子さん・陳鍵さん・大原康平さんに感謝申し上げます．ミーティングなどで正確な指摘を頂き，諸先輩方の深い洞察力を見習いたいと思いました．また，生研で修士を生き残るすべを教えて頂きなんとかここまで来ることが出来ました．ありがとうございます．

研究室の後輩方である，三條嵩明君・別所祐太朗君・福田展和君・大葉大輔君・土屋潤一郎君・蔦侑磨君・杉山普君・左天池君に感謝申し上げます．初めての後輩

ということもあり，先輩として至らないところが多々あったと思いますが，諸君らと研究の話をすることは大変楽しかったです．

そして，研究室同期である，羅博明君，遠田哲史君・根石将人君・佐久間仁君・清水洸希君・張翔君に感謝申し上げます．研究室に行くと常に誰かおり，話すことが出来るということが辛い時の心の支えになっていました．研究に関してもお互いに教え合うことができ，2 年間楽しく過ごせたことは素晴らしい思い出となりました．

また，締切の直前で研究の評価に手伝って頂いた，石渡祥之佑さん・日並遼太さん・赤崎智さん・三條嵩明君には特別感謝申し上げます．皆様の協力なくしてはこの論文は完成しなかったでしょう．

最後に，24 年間常に支えて頂いている家族に感謝いたします．気分が落ち込んでいる時に，気分転換になるように取り計らっていただいたことは大変助かりました．心より感謝申し上げます．

2018 年 1 月 31 日

参考文献

[1] Zhengli Zhao, Dheeru Dua, and Sameer Singh. Generating natural adversarial examples. In *International Conference on Learning Representations*, 2018.

[2] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[3] Danqi Chen, Jason Bolton, and Christopher D. Manning. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2358–2367. Association for Computational Linguistics, 2016.

[4] Masatoshi Tsuchiya. Performance Impact Caused by Hidden Bias of Training Data for Recognizing Textual Entailment. In *Language Resources and Evaluation Conference*, 2018.

[5] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pp. 1096–1103, New York, NY, USA, 2008. ACM.

[6] Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Unsupervised Machine Translation Using Monolingual Corpora Only. In *International Conference on Learning Representations*, 2018.

[7] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust Physical-World Attacks on Deep Learning Visual Classification. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 1625–1634, Salt Lake City, UT, USA, June 2018.

[8] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman. SoK: Security and Privacy in Machine Learning. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 399–414, April 2018.

[9] Nicolas Papernot, Patrick D. McDaniel, and Ian J. Goodfellow. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples. *CoRR*, Vol. abs/1605.07277, , 2016.

[10] Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S. Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian Goodfellow. Adversarial Spheres. *arXiv:1801.02774 [cs]*, 2018.

[11] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, May 2017.

[12] Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. In *International Conference on Learning Representations*, 2017.

[13] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-box Adversarial Examples for Text Classification. In *Proceedings of the 56th*

Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 31–36. Association for Computational Linguistics, 2018.

[14] Motoki Sato, Jun Suzuki, Hiroyuki Shindo, and Yuji Matsumoto. Interpretable adversarial perturbation in input embedding space for text. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 4323–4330. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[15] Georg Heigold, Stalin Varanasi, Günter Neumann, and Josef Genabith. How robust are character-based word embeddings in tagging and mt against wrod scramlbing or randdm nouse? In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pp. 68–80. Association for Machine Translation in the Americas, 2018.

[16] Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2890–2896. Association for Computational Linguistics, 2018.

[17] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Semantically equivalent adversarial rules for debugging nlp models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 856–865. Association for Computational Linguistics, 2018.

[18] G E Hinton and R R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, Vol. 313, No. 5786, pp. 504–507, July 2006.

[19] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, Vol. 45, No. 11, pp. 2673–2681, Nov 1997.

[20] Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. In *International Conference on Learning Representations*, 2018.

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc., 2013.

[22] Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1875–1885. Association for Computational Linguistics, 2018.

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[24] Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–148. Association for Computational Linguistics, 2016.

[25] Junbo (Jake) Zhao, Yoon Kim, Kelly Zhang, Alexander M. Rush, and Yann LeCun. Adversarially regularized autoencoders, 2018.

[26] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[27] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734. Association for Computational Linguistics, 2014.

[28] Felix A. Gers, Jürgen A. Schmidhuber, and Fred A. Cummins. Learning to forget: Continual prediction with lstm. *Neural Comput.*, Vol. 12, No. 10, pp. 2451–2471, October 2000.

[29] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Netw.*, Vol. 12, No. 1, pp. 145–151, January 1999.

[30] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, Vol. abs/1412.6980, , 2014.

[31] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

[32] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002.

[33] Kenneth Heafield. KenLM: faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pp. 187–197, Edinburgh, Scotland, United Kingdom, July 2011.

[34] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60, 2014.

[35] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pp. 142–150, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[36] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud Maria Jiménez-Zafra, and Gülşen Eryiğit. Semeval-2016 task 5 : aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 19–30. Association for Computational Linguistics, 2016.

[37] Yelp Dataset, 2014. https://www.yelp.com/dataset/challenge.

[38] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751. Association for Computational Linguistics, 2014.

[39] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, Vol. 5, pp. 135–146, 2017.

[40] JL Fleiss and M Davies. Jackknifing functions of multinomial frequencies, with an application to a measure of concordance. *Am J Epidemiol*, 1982.

[41] J. R. Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, Vol. 33, No. 1, pp. 159–174, 1 1977.

[42] Hanjun Dai, Yingtao Tian, Bo Dai, Steven Skiena, and Le Song. Syntax-directed variational autoencoder for structured data. In *International Conference on Learning Representations*, 2018.

[43] Zhiting Hu, Zichao Yang, Ruslan R Salakhutdinov, LIANHUI Qin, Xiaodan Liang, Haoye Dong, and Eric P Xing. Deep generative models with learnable knowledge constraints. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pp. 10522–10533. Curran Associates, Inc., 2018.

発表文献

査読なし国内会議

1. 保田和彦，吉永直樹，喜連川優，Denoising Autoencoder を用いた多様な敵対的サンプルの生成，第 11 回データ工学と情報マネジメントに関するフォーラム（DEIM 2019），長崎，2019.（発表予定）

研究会ポスター発表

1. 保田和彦，吉永直樹，喜連川優，学習データ拡張のための多様な敵対的サンプル生成モデル，NLP 若手の会 (YANS) 第 13 回シンポジウム，香川，2018.