

# 原著論文

## 敵対生成ネットワークによる文書生成

---

岩田一樹

東北福祉大学

### Text Generation Based on a Japanese Data Set Using Generative Adversarial Networks

IWATA Kazuki

Tohoku Fukushi University

#### Abstract

Automatic text generation using generative adversarial networks (GANs) such as SeqGAN, TextGAN, and LeakGAN has attracted substantial attention. There are numerous studies regarding the automatic generation of English and Chinese texts using GANs or related algorithms. Moreover, methods to generate sentences automatically based on the speeches delivered by the former US president Barack Obama, captions of images, or Chinese poems have been realized. However, there are only a few reports regarding the generation of text from Japanese sentences.

In this study, to investigate the differences in generating text in different languages, we generate sentences from the Japanese novel “Botchan,” which was written by Soseki Natsume, using maximum likelihood estimation, SeqGAN, TextGAN, and LeakGAN. Furthermore, we evaluate the generated text using the metrics of negative log-likelihood loss, 2-gram BLEU, and embedding similarity. Consequently, it is concluded that LeakGAN can generate the most natural texts and that, for unknown reasons, TextGAN does not perform well with regard to automatic text generation.

Keywords : 自然言語処理、文書生成、敵対生成ネットワーク、青空文庫、坊ちゃん

#### 1. はじめに

ニューラル・ネットワークを使用した敵対生成ネットワーク (Generative Adversarial Network ; GAN) の提案から、機械による自動生成が注目を集めている [1-3]。Goodfellow らが初めて提案した GAN アルゴリズムは画像の生成を対象としており、画像を生成する生成器 (Generator)、および、生成器から生成された機械の画像か、人が描いた画像 (学習データ) かを識別する識別器 (Discriminator) の2つのオブジェクトから成ったモデルであった [1]。そして、これら2つのオブジェクトが敵対的に学習を進めるものである。すなわち、定性的には、生成器は識別器が機械の生成か人の生成かを区別できないように学習を進め、それに対して、識別器は生成器の生成か人の生成かを正確に識別できるように学習を進める。なお、画像の生成には生成器、および、識別器に畳み込みニューラル・ネットワーク (Convolutional Neural Network ; CNN) が用いられることが多い [4]。CNN とは、位置など局在的な変化に堅牢な局在不変性を仮定可能なニューラル・ネットワークのモデルで、最近の機械学習分野において中心的なニューラル・ネットワーク構造の1つである [5]。

最近では、日本語や英語、中国語といった自然言語や音声など系列を有するデータへの GAN の応用が提案されている。これら系列を取り扱う GAN 群と画像など系列を有さないデータを取り扱う GAN 群と

の違いとして、CNNも含めて、従来のニューラル・ネットワーク構造が系列データを扱うのに不向きであることから、その問題を解決するために生成器として再帰的ニューラル・ネットワーク（Recurrent Neural Network；RNN）を用いる点がある [6]。

RNNは  $0 \rightarrow 1 \rightarrow 2 \cdots \rightarrow t$  の順番を有する系列ベクトル  $X = (\vec{x}_0, \vec{x}_1, \vec{x}_2 \cdots \vec{x}_t)$  の入力データを、下式で与えられる出力として取り扱う。

$$h_t = A\vec{x}_t + Wh_{t-1} + b.$$

ここで、 $A$  はニューラル・ネットワーク、 $b$  はバイアス、 $H = (h_0, h_1, h_2 \cdots h_t)$  は入力  $\vec{x}_i$  ( $0 \leq i \leq t$ ) に対する出力値 ( $h_i$ )、 $W$  は系列間の重みであり、 $h_t$  に  $h_{t-1}$  の情報をどの程度反映されるかの項である。そして、 $t$  番目の出力  $h_t$  を、 $t$  番目の入力である  $\vec{x}_t$  による出力、 $t-1$  番目の出力に由来する  $h_{t-1}$ 、および  $b$  の和として与える。したがって、実際の計算では、 $t=0$  において、 $Ax_0 + b$  によって  $h_0$  を取得し、 $Ax_1 + Wh_0 + b$  によって  $h_1$  を取得することを逐次的に  $t$  まで算出して、系列情報を反映した出力を得る。

そして、学習には、最終的な出力に対する教師の値を用意するなどし、出力値と教師値の2つから得られる損失値を用いて、後方誤差伝搬法で行う。その際の留意点として、 $A$ 、 $W$ 、 $b$  はそれぞれ同じものを用いる点で、各入力時間に対して  $A$ 、 $W$ 、 $b$  があるのではなく、全ての時刻で共通の  $A$ 、 $W$ 、 $b$  を用いる。

ただし、このモデルには  $W$  の積とみなされる項があるため、 $W$  が1付近でない場合、系列情報が消失してしまったり ( $W \ll 1$ )、逆に、発散してしまったり ( $W \gg 1$ ) する問題がある。この問題を「忘却ゲート」「入力ゲート」「更新ゲート」「出力ゲート」と呼ばれるゲートを導入して改善したモデルが Long Short-Term Memory (LSTM) と呼ばれるものである [6, 7]。この LSTM は音声認識や機械翻訳など様々な分野で利用されている。

敵対性を使用した文生成の GAN としては、SeqGAN や TextGAN、LeakGAN などが提案されている [8-10]。これらの GAN においては、文が文字や単語の系列データとみなし、生成器として LSTM、識別機として CNN を用いて、漢詩やオバマ元米国大統領のスピーチ、俳句などの生成に成功している [8-11]。今後も、キャプション生成や新聞記事生成の自動化、音声合成など様々な系列を有する分野での応用が期待されている。

しかし、俳句などの一部の研究はあるが、文生成に関する研究は英語と中国語のものが多く、日本語に関するものは少ない。単語の順に自由度の高い日本語と単語の順が重要でその自由度が低い英語とでは、文生成に違いが生じる可能性がある。そこで、本研究では、以下の4つの系列データ生成アルゴリズムについて日本語のデータセットを学習データとして文の自動生成実験を行い、その結果を検討した。なお、下記において、SeqGAN、TextGAN、LeakGAN が敵対生成アルゴリズムである。

#### ・ Maximum Likelihood Estimation [6, 12]

Maximum Likelihood Estimation (MLE) は、文における、文字または単語の生成確率分布を  $\pi_\theta$  によって近似し ( $\theta$ : パラメータ)、それからのサンプリングが学習データと一致するようにパラメータを更新するものである。すなわち、学習データが  $s_t = [x_0, x_1, \dots, x_{t-1}, x_t]$  である際に、 $\pi_\theta(x_t | s_{1:t-1})$  または、 $\log \pi_\theta(x_t | s_{1:t-1})$  が最大となるようにパラメータを決定することに相当する。実際の学習では、これを全ての学習データ、つまり、全ての文と文字または単語の系列に対して行い、

$$\max_{\theta} \sum_x \sum_t \log \pi_{\theta}(x_t | s_{1:t-1}).$$

となるように  $\theta$  を定め、その上で、系列に対して逐次的に  $\pi_{\theta}(\cdot | Y_{1:t-1})$  が最大となる  $\cdot$  を連ねて文を生成する。なお、ニューラル・ネットワークを用いる場合は、 $\pi_{\theta}$  に LSTM などの RNN を用いる。

MLE は元の文章から生成確率を学習する教師あり学習であるため、以下で述べる敵対性を用いたアルゴリズムよりは、学習が安定しやすい。その一方で、教師となるデータと類似した文が生成されやすいと考えられる。

・ SeqGAN [ 8 ]

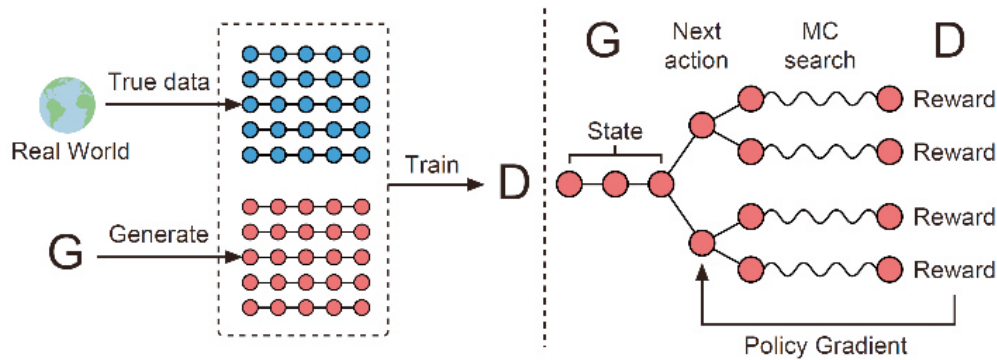


図1 SeqGAN の概念図 ([ 8 ] から引用)

SeqGAN とは、生成器と識別器を用いて生成器を学習させていく GAN の一種である。ただし、系列を有さないデータを対象とした GAN と異なり、系列モデル、すなわち、 $Y_{1:t-1}$  から  $y_t$  を予測するに際して、図 1 の概念図にあるように  $y_{t+1}$  以降をモンテカルロ法により生成し、その結果も含めて真偽を識別器に判定させ、それらを元に、 $Y_{1:t-1}$  から  $y_t$  を生成した結果の評価値を決定する。その際、生成器の誤差関数 ( $J_{\theta}$ ) は下式によって与えられる。

$$J_{\theta} = \sum G_{\theta}(y_t | Y_{1:t-1}) Q_{D_{\phi}}^{G_{\theta}}(Y_{1:t-1}, y_t).$$

$$Q_{D_{\phi}}^{G_{\theta}}(s = Y_{1:t-1}, a = y_t) = \begin{cases} \frac{1}{N} \sum_{n=1}^N D_{\phi}(Y_{1:T}^n), Y_{1:T}^n \in \text{MC}^{G_{\theta}}(Y_{1:t}; N) & \text{for } t < T. \\ D_{\phi}(Y_{1:t}) & \text{for } t = T. \end{cases}$$

ここで、 $G_{\theta}$  ( $G_{\beta}$ ) は  $\theta$  ( $\beta$ ) をパラメータとする LSTM を用いた生成器、 $D_{\phi}$  は  $\phi$  をパラメータとする CNN を用いた識別器、 $Q$  は報酬関数である。この誤差関数は、 $Y_{1:t-1}$  から  $y_t$  を生成する尤度を報酬関数で表し、それを生成器によって  $Y_{1:t-1}$  から  $y_t$  を生成される確率で期待値をとることに相当する。また、報酬関数は、 $Y_{1:t-1}$  から  $y_t$  を生成器によって生成した後、以降をパラメータ更新する前の生成器 ( $G_{\beta}$ ) からモンテカルロ法とマルコフ連鎖により複数生成し、その生成結果を識別器で評価、その上で、平均化したものである。この誤差関数によって、生成器は報酬関数が最大となるよう、パラメータ  $\theta$  を強化学習 (Q 学習) ベースで更新する。

一方、識別器は一般の GAN と同様に [ 1 ]、

$$-\mathbb{E}_{Y \sim \text{data}}[\log D_{\phi}(Y)] - \mathbb{E}_{Y \sim G_{\theta}}[\log(1 - D_{\phi}(Y))].$$

が最小になるよう  $\phi$  を更新する。なお、この式は識別器が全ての入力に対して真偽を正しく識別した際に最小値 0 となる。

このように、生成器の学習において敵対生成アルゴリズムと強化学習アルゴリズムを組み合わせることで、次に来るべき単語の確率的な妥当性のみを元にして学習し、文を生成する MLE とは異なり、生成した文全体の妥当性を識別器が評価し、それを元に生成器は学習をすることから、文法の妥当性を包含した生成を可能にするとされている [8]。また、識別機には、局在的な変化に堅牢な局在不変性を仮定可能な CNN を用いることから、真贋の判定への語順の影響は少ないと考えられ、語順の自由度の高い日本語の文においても英語などと同様に文が生成可能と期待される。

・ TextGAN [9]

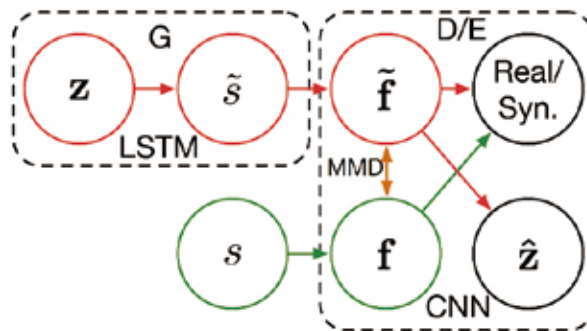


図2 TextGAN の概念図 ([9] から引用)

TextGAN も SeqGAN と同様に生成器として LSTM、識別器として CNN を用いた GAN モデルであるが、その特徴は Feature Matching と再構成を導入した点にある。

まず、Feature Matching とは、TextGAN の概念図 (図 2) において、MMD (Maximum Mean Discrepancy) で表現されている部分で、学習文と生成文を Gaussian カーネルによって再生核ヒルベルト空間に写像し、その平均差によって一致度 (MMD) を測定し、生成器はそれが最小になるように、識別器はそれが最大となるようにパラメータを更新することを指す。

また、再構成とは図 2 において、 $\tilde{f}$  から  $\hat{z}$ 、すなわち、生成時にシードとなったノイズ ( $z$ ) を推定することを意味する。したがって、識別器は上述のように MMD を最大化すると同時に、 $\|z - \hat{z}\|^2$  は最小化するようにパラメータを更新する。

TextGAN は SeqGAN と異なり  $y_{t+1}$  以降の生成は行わないため、学習にモンテカルロ法は用いない。そのため、識別器が判定する際の系列長が短くなるので、SeqGAN よりは学習が安定しやすいと期待される。このアルゴリズムによって生成される文は短文になる傾向があり、20 単語以下であれば、文法的にも比較的正しい文を生成することが知られている [9]。

・ LeakGAN [10]

LeakGAN は SeqGAN と似たアルゴリズムで、SeqGAN と同様に生成器として LSTM、識別器として CNN を用い、モンテカルロ法と報酬関数を用いる強化学習をベースとした GAN モデルである。しかし、SeqGAN との違いに、生成器は Worker と Manager の 2 種類の LSTM からできている点と、識別器が

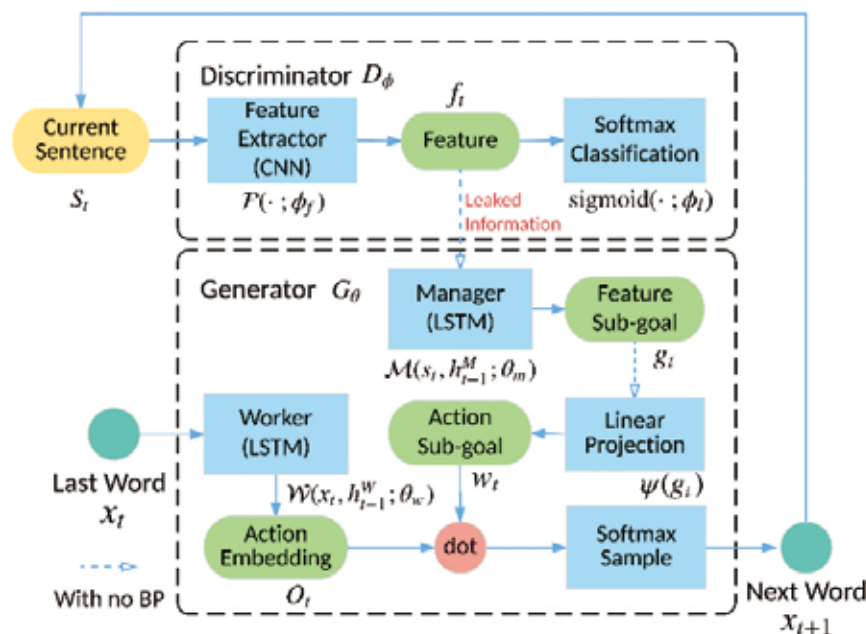


図3 LeakGAN の概念図 ([10] から引用)

ら Manager への情報の流れ (Leaked Information) の 2 点がある (図 3)。この識別器からの情報リークによって、それが無い場合よりも、生成器が多くの情報を利用でき、それによって、比較的長文の生成に強く、また、強化学習を用いた GAN の中では学習が比較的安定である特徴がある。その一方で、生成器へのリークは、学習データの情報を含んだ識別器の学習内容を生成器に反映させることから、SeqGAN よりも学習データに近い文を生成すると考えられる。

以上が、本研究で比較を実施するアルゴリズムである。MLE と GAN は文を生成するという点では同様であるが、前者が上述のように教師あり学習なのに対して、後者は敵対性による強化学習の一種である点に違いがある。この違いから、MLE は基本的には教師である学習データと同じ、または、非常に類似した文のみしか生成できないのに対し、生成器と識別器の競合の結果、それが人の判断に近いかどうかは別にして、GAN からは機械の判断した学習データの特徴にそってより多様な文が生成されることが期待される。この学習データに捉われない多様な生成が、文書のみでなく、GAN 全般の研究が行われるモチベーションであり、最終的には機械のみによる生成が可能となる可能性がある。そして、本研究の対象である文書においては、学習データの特徴に沿って、新しい文を生成可能な点に有用性がある。すなわち、小説、論文、詩などの特徴に沿った文の自動生成が可能になるということである。

## 2. 方法

### 2.1 評価指標

本研究では、文生成の性能比較に以下の 4 種の指標を用いた。

- ・ Negative Log-Likelihood loss [ 8 ]

Negative Log-Likelihood loss (NLL) とは、尤度ベースの評価指標の一つである。GAN による文書生成において、理想的な学習結果は、学習データ (人の書いた文) の単語生成分布と生成データ (機械が生成した文) の単語生成分布との間の交差エントロピー誤差が 0 になっている状態である。したがって、生成器の良さをそれら 2 つの交差エントロピー誤差から評価するのが NLL である。

本研究では、下式で与えられる NLL を用いた。

$$\text{NLL} = -\mathbb{E}_{Y_{1:T} \sim G_{\text{real}}} \left[ \sum_{t=1}^T \log(G_{\theta}(y_t | Y_{1:t-1})) \right]$$

ここで、 $\mathbb{E}$  は 1 単語目から T 単語目までの平均、 $G_{\text{real}}$  は学習データの生成分布、 $G_{\theta}$  は生成器の生成分布であり、 $\theta$  は分布関数のパラメータを示す（本稿では  $G_{\theta}$  を LSTM で近似しているため、LSTM のパラメータが相当）。また、 $G_{\theta}(y_t | Y_{1:t-1})$  は、1 から  $t-1$  までの系列（単語の並び）だった際に、 $t$  番目が  $y_t$  となる条件付確率を示す。この指標は 0 が最小値であり、小さいほど生成器の性能が良いと考えられている。

#### ・ BLEU score [13]

BLEU は文書間の単語の類似性の指標として広く用いられているもので、下式で与えられ、1 に近いほど性能が良いと考えられている。

$$\text{BLEU} = \min \left( 1, \exp \left( 1 - \frac{r}{c} \right) \right) \exp \left( \sum_{n=1}^N \frac{1}{N} \log P_n \right)$$

ここで、 $r$  は参照文の長さ、 $c$  は生成文の長さ、 $N$  は生成文内の  $n$ -gram 数、 $P_n$  は、

$$P_n = \frac{\sum_{\text{全生成文}} \sum_{\text{全生成文中の } n\text{-gram}} (\text{学習文と生成文が共有する } n\text{-gram 数})}{\sum_{\text{全生成文}} \sum_{\text{全生成文中の } n\text{-gram}} (n\text{-gram 数})}$$

で与えられる。本研究では、 $N=3$ （3 単語を 1 つのユニットとすることに相当）の BLEU 値で性能比較を行った。

#### ・ Embedding Similarity [12]

Embedding Similarity (EmbSim) とは、学習文と生成文との間における単語の分散表現、すなわち、登場する単語の意味の類似性を比較した指標である。

まず、学習文群において、word2vec [14] を用いて、各単語を分散表現して、各単語間のコサイン類似度を行列として、 $W_{i,j} = \cos(e_i, e_j)$  と表現する。また、生成器から生成された文についても同様なことを行い、 $W'_{i,j} = \cos(e'_i, e'_j)$  と表現し、下式によって評価する。

$$\text{EmbSim} = \log \left( \sum_{i=1}^N \frac{\cos(W'_i, W_i)}{N} \right)$$

EmbSim は学習文書と生成文書間での単語の意味合いが完全に一致する際に 0 となることから、値が 0 に近いほど性能が良いと考えられている。

## 2.2 データセット

学習データの前処理については、前処理として、底本、ルビ、注釈、カギ括弧、および、改行を削除した後、1 文ずつに分けて、単語および句読点で分かち書きを行った。

本研究では、青空文庫にアップロードされている『坊っちゃん』を学習セットとして用いた [15, 16]。そして、その基本量は単語の種類が 5,380、文数が 2,668、1 文の平均単語数が 19.0 (std : 10.3) である。

## 2.3 学習設定

各モデルの生成器、ならびに、識別器のネットワーク構造は各論文に準じ [8-10]、MLE については GAN の生成器と同じ構造の LSTM を用いた。学習回数は、MLE については100 epochs、敵対生成アルゴリズム群については、生成器のプレトレーニングを80 epochs、次いで識別器のプレトレーニングを80 epochs、そして、敵対トレーニングを100 epochs を行った。ただし、LeaKGAN については、敵対トレーニングを5回の生成器のトレーニングと5回の識別器のトレーニングを1セットとして、それを10セット行うことで他の敵対生成アルゴリズムと学習回数を合わせた。

## 3. 結果

『坊ちゃん』を学習データとした際の学習曲線を図4~5に示す。図は横軸が学習回数 (epochs)、縦軸が、それぞれ、NLL (図4)、BLEU (図5)、EmbSim (図6) の値である。また、縦に引いた直線はプレトレーニングから敵対トレーニングへと切り替わる80 epochsを示している。なお、各図において、右図は左図を拡大したものである。

NLL についての学習曲線では (図4)、まず、TextGAN の成績が振るわないのが見て取れる。また、敵対生成アルゴリズム3種類において、SeqGAN と TextGAN が敵対生成トレーニング開始時 (80 epochs) から指標の値が悪くなっているのに対し、LeakGAN のみ、指標の値が改善していることがわかる。このことから、LeakGAN の学習が安定して進んでいることが予想される。なお、プレトレーニングにおいて、MLE と SeqGAN の学習曲線が良く似ているのは、本研究での MLE のネットワーク構造と SeqGAN の生成器に用いている MLE のネットワーク構造が同じであるからと推察される。

図5はBLEUについての学習曲線である。ここでもやはり、TextGAN が振るわないのが見て取れる。一方、SeqGAN と LeakGAN を比較すると、プレトレーニング時、および、敵対生成トレーニングが開始されてからすぐは LeakGAN の方が BLEU の値が高いが、敵対生成トレーニングが進むと、LeakGAN の値がほぼ変わらない、または、わずかに下降するのに対して、SeqGAN の値が上昇することによって、その値が逆転している。また、SeqGAN と MLE とを比較すると、敵対トレーニングが開始されてから後に値に明確な違いが生じることから、SeqGAN における BLEU の上昇は敵対トレーニングの結果だと考えられる。

EmbSim の結果 (図6) についても、上述の2つの指標と同様に TextGAN の値が最も悪い。また、SeqGAN と LeakGAN に比較においては、BLEU と同様の傾向が見て取れ、LeakGAN が学習開始当初から学習データとの高い類似性を示し、敵対トレーニング後もほぼ同じ値で安定しているのに対して、SeqGAN は、敵対トレーニング開始後に高い類似性を示すようになる。

本研究で得られたこれら3つの指標の振る舞いは、TextGAN を除いて、英語の学習データとして得られている結果と類似している [12]。このことから、TextGAN を除いて、日本語データにおいても先行研究と同程度の生成が可能であることを示唆していると考えられる。

次に、実際にどのような文が生成されているかを確認するために、表1~4に各アルゴリズムによって生成された文を示す。示した文は、各アルゴリズムにおいて10,000の文を機械によって自動生成させ、その中から一様乱数によって選択された8文である。なお、スペースが入っている箇所ではちががかれている。また、学習データ、および、各アルゴリズムにおいて生成された10,000の文群の基本量を表5に示す。

実際に生成された文においては、LeakGAN が生成している文に比較的自然で、文法的にも無理のない文が多いように感じられる。また、少し定量的に、学習データに含まれた文の基本量 (単語種類: 5,380、1文中の平均単語数が19.0 (std: 10.3)) と比較すると MLE および LeakGAN がそれに近い値になっているのに対して、SeqGAN は値が小さくなっている。すなわち、学習データより短い文を生成する傾向が

あることを意味している。

SeqGANの生成器がMLEと同じであること、プレトレーニング時においてこの2つに大きな差が見受けられないことから、この差異は敵対トレーニングの結果であると考えられ、そして、この基本量の変化がBLEUとEmbSimに反映されているものと考えられる。すなわち、BLEUは $\min\left(1, \exp\left(1 - \frac{r}{c}\right)\right)$ の項が学習文よりも短い文を生成した際のペナルティとなっている一方で、 $\exp\left(\sum_{n=1}^N \frac{1}{N} \log p_n\right)$ については、1文が短い方が高いスコアが出やすい。どちらの項がより効くかは自明ではないが、今回の実験結果からは後者の項の効果によって、スコアが上がったのではないかと推察される。一方、EmbSimについては、 $W'_{i,j} = \cos(e'_i, e'_j)$ の計算時に、発現していない単語のコサイン類似度の影響が効いていると考えられる。この様に生成される文が短文化される傾向は、マルコフ連鎖を用いることで短い系列の方が高く評価されやすいことに起因していると考えられ、既存の研究結果と整合している [8]。

一方で、MLEおよびLeakGANがほぼ、学習データと同じ基本量になっている。このことは、MLEにおいては、学習データに合うように学習を進めることから、妥当な結果といえる。一方、LeakGANについては、識別器から生成器へのリークによって、SeqGANよりも学習データの内容に似た文を生成したと解釈できる。

しかし、それらの一方で、TextGANは全く文が成立していない。基本量についても、既存研究においては比較的短い文を生成しやすいとされているが [9]、本研究においては長文になっており、平均でプレトレーニングに用いた文のそれの2倍以上の長さのものを生成してしまっている。このように性能が発揮されていない理由が、学習データが日本語によるものだからなのか、別に理由があるのかは未解明である。今後、さらにハイパーパラメータを変更した実験などを行い、その原因を検討する必要がある。

また、MLEはより学習データに近い文を生成し、GANは多様な文を生成することが期待されるが、表1～4からは、MLE、GAN問わず、学習データとはあまり似ていない。この点について、MLEは学習不足であると推察可能であり、GANについては、そのアルゴリズム上、妥当な結果といえる。

以上から、TextGANを除いては、本研究において、既存研究と4つの指標の変化が同傾向である点、および、抽出した生成文が文として成り立っていることが見て取れることから、日本語の文についても敵対性による文書生成は可能であると考えられる。

以下の図において、左図を拡大したものが右図であり、破線は、プレトレーニングから敵対トレーニングへと移行する80 epochsを示している。

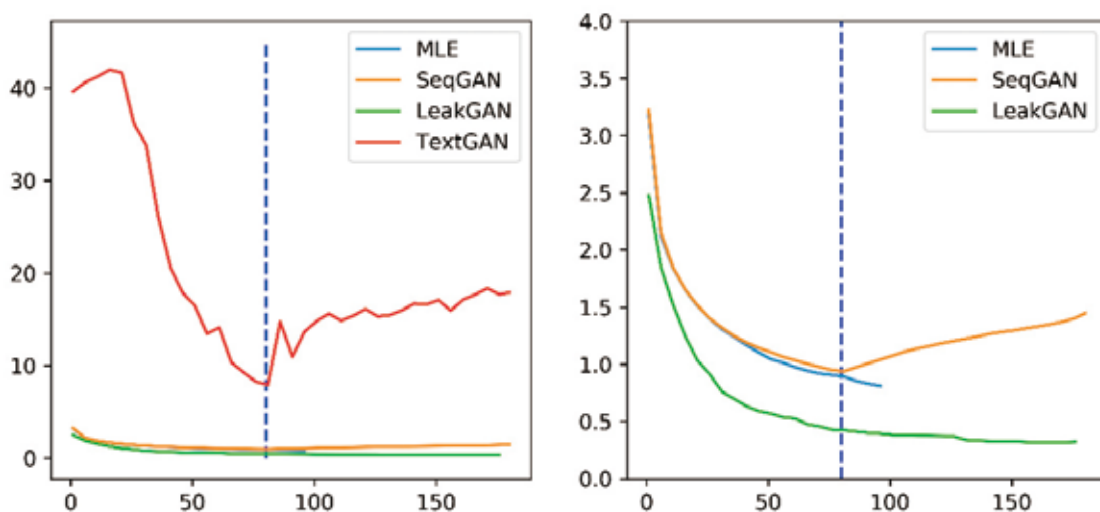


図4 NLLの学習曲線



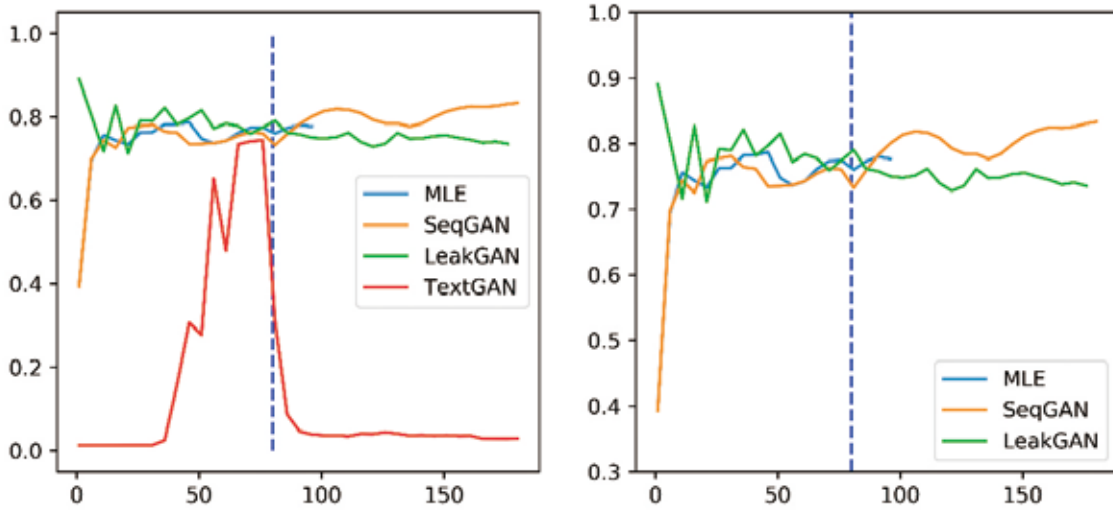


図5 BLEUの学習曲線

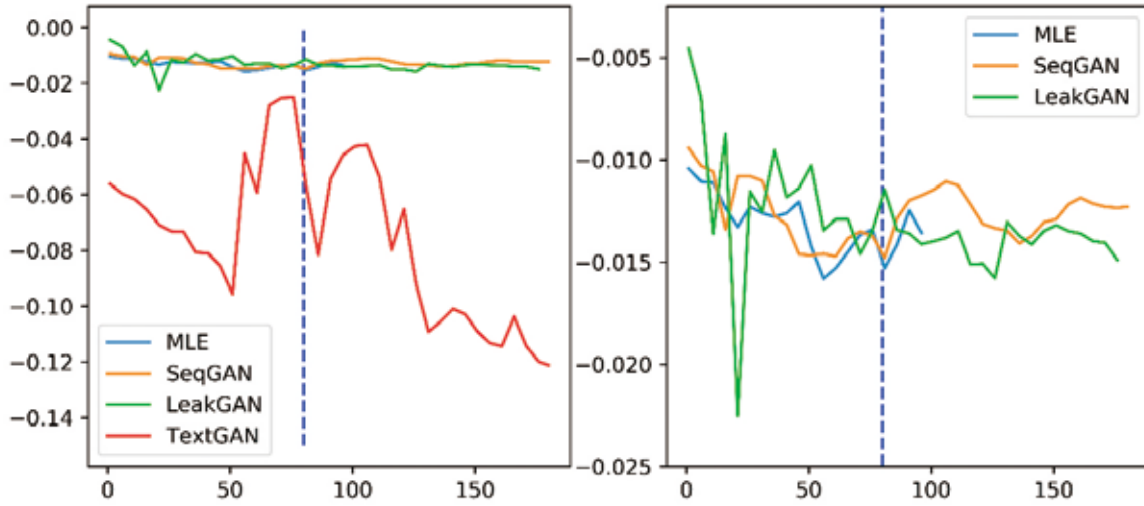


図6 EmbSimの学習曲線

表1 MLEによって生成された文（『坊ちゃん』）

1	温泉の月給が祝詞を相手に十五灯ほど、って出来た。
2	その外によっほど地に理窟が一本を据えた。
3	十の事じゃありませんと済むなもし。
4	妙な事がある。
5	天麩羅を捲いている。
6	この否や突いて飛んだった。
7	浮がなくっちゃ人間の癖に出来る。
8	おれは、すぐ講釈をした

表2 SeqGANによって生成された文（『坊ちゃん』）

1	じゃ、驚いたのはよく知ったかみさんが、気をつけてひどい目に遭わないような天気だ
2	害になら雇うぞなのです。
3	いよいようらなり面倒だろう。
4	これじゃおれや方を親不孝だか知らない。
5	会津っぼなんて生意気な者をして、述べるのか、釣れたのは起き抜けになって罰をやる。
6	足を覗き込んだ。
7	しかし一番寝ては驚いた。
8	と勧めても無理はない

表3 TextGANによって生成された文（『坊ちゃん』）

1	敷きつめ 散歩 茶碗 われわれ 鯛 こないだ いっ 並べ かしこまり みろ 処決 二月 ぞくぐり 行こ 敷きつめ 丸め 異状 行く先 け 売りゃ 取り 鎮める 朝顔 ぼし 野 やおや 西洋 覚え 西洋 っ て 紫 舂 寄せ 水水 奇麗 表せ まだしも 売りゃ みんな 紅梅 鉄面皮 井戸端 売る 括っ 熱心 笑われる 何だか 恥ずかしい
2	あ だら 可哀想 目くばせ 大概 奥さん 大概 敷きつめ 大学 召 まだしも 反動 敷きつめ 落し しまっ 並べ 奇麗 飯 保っ 血気 たとい 敵味方 島 逢っ づれ 死ぬ 遠足 求めて 引っ立てる お前 どう 鯛 離れ 肥料 やおや 自 す 逢っ いっ 車夫 過ごせ 取り 汗 けれど 心当り 異状 何だか 引き取っ
3	東西 捜し当て 並び 半日 しまっ ぐるり 美しい 乱暴 しまっ 横風 名折れ ずたずた 蚊帳 あんた 車夫 尖 一室 別室 蚊帳 鼓吹 おき 一概に 尺 靡い 二つ 時 おや 奥さん 言語 薬味 売買 頬 約束 箒 浜 過 耽る 笑い草 よ 茶碗 よ われ われ 敷きつめ 散歩 処決 策 離れ がっ
4	言語 売る 門 ごとく 遠足 みろ やっぱり 一室 みろ 覗い やおや 笑う 敷きつめ 風儀 茶 心当り 黒焦 売買 やおや 逢っ 二月 大学 たまらなかつ 潮水 さよう 閑静 似合い たくっ っ て 敷きつめ 寝巻 略 熱心 陣羽織 水水 住み 逢っ ぼし みろ 並べ 返 蒞 蕪版 差し上げ 批評 ひらつかあ 敷きつめ 出入
5	ぬかし 下宿 茶碗 われわれ 蚊帳 やおや やおや 語 ひと通り 失言 廻す みろ ぐるり 美しい 呂律 二つ 光 沢庵 茶碗 兼 尺 はいろう 売りゃ やおや 抱い 若い衆 そら 取り 顔 云い 不順 井戸端 蝦蟇口 召 がいや 瀬戸物 中学生 瀬戸物 さよう 落ちつか 意気地 張り飛ばし はた 提げ 切っ だら 学資 あくる
6	漕ぎ 所為 印材 六日 目 中 逢っ 紫色 見物 やしまいし 薬味 教育者 苦しむ 内所 さ 責任者 廊下 新来 一つも だら 腹 案 だら だら 廊下 どんと 辞 田舎言葉 下し そのうち 乗ろ 辞 暮方 召 売買 練兵 重み 敷きつめ 留める 蝦蟇口 親類 印材 漕ぎ 着け 探偵 第 意く 四つ 一つも
7	色 しまっ ぐるり け ぼし 茶碗 われわれ 鯛 ハハハハ 笑い草 まほう 揉み 並び 顔 顔 下宿人 がいや 着く 新来 威張 れ 烟草 面白い がいや 茶碗 茶碗 われわれ 鯛 大き 黒々 寝巻 私語 恐る 遠足 求めて 野 方角 処決 金鎖 れれ 直す 耽る 浜 提げ 思っ 至っ 所為 立た 過
8	次第に 肥料 腹案 みろ 車夫 非 乗ぜ ひらつか 大学 奥さん 消え 売る 贅沢 坐っ 大学 大学 神経衰弱 約束 若い衆 三 割 乗せ こんなに 割っ 断わっ 早速 大概 ヶ条 みろ 事実 う 某 うて なじみ 擦り 黒焦 あ 抛り 出し やおや やおや やおや 逢っ ふかし 稽古 しまっ 茶 燈 周囲 みろ

表4 LeakGANによって生成された文（『坊ちゃん』）

1	おれが蜜柑の事を考えている。
2	赤シャツはおれに一銭五厘奮発させる気だから黙っていた。
3	おれが蜜柑の事を考えているところへ、偶然山嵐が話しにやって来た。
4	活気にみちて困るなら運動場へ出て相撲でも取るがいい。
5	おれは様子が分らない。
6	何だか憐れぼくってたまらない。
7	おれは肴を食ったら、こう答えた。
8	おれは一貫張の机の上にあった置き洋燈をふっと吹きつけた。

表5 自動生成された文の基本量

アルゴリズム	使用単語数	1文中の単語数の平均	1文中の単語数の標準偏差
学習データ	5,380	19.0	10.3
MLE	4,705	18.0	9.7
SeqGAN	2,723	14.0	6.7
TextGAN	3,365	48.6	3.2
LeakGAN	4,189	19.5	9.9

#### 4. まとめ

本研究では、MLE および3種類のGANを用いて、日本語のデータセットを学習データとして自動文章生成を行い、以下の知見を得た

- ・NLL、BLEU、EmbSimの学習曲線が、TextGANを除いて、既存研究の傾向と類似していた点と生成された文に文として成立していたものが見て取れたことから、学習データが日本語であってもGANによる文書生成は可能であることが示唆された
- ・MLE、LeakGANと比較して、SeqGANの生成する文は短いものが多かった
- ・TextGANは全く性能が発揮できなかった

本研究において、TextGANの性能が全くでなかったが、その原因の解明はできていない。したがって、これに関しては、更なる実験が必要である。

また、本研究で用いたデータセットの文数は一般的なGANで使用するデータ数と比較するとかなり少量なので、大きなデータセットにおいても学習可能なのかを実験する必要がある。

#### 謝辞

本稿を執筆するにあたり、東北福祉大学 総合マネジメント学部 情報福祉マネジメント学科の石渡さん、古屋さん、星さんにはデータの収集・整理や内容などご助言頂きました。この場で、感謝申し上げます。

本研究は、本研究はJSPS 科研費 18K13240の助成を受けて実施されたものです。

#### 参照

- 1) I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets", Advances in Neural Information Processing Systems, pp.2672-2680, 2014.
- 2) A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks", ICLR, 2016.
- 3) Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford and Xi Chen, "Improved Techniques for Training GANs", 2016, <https://arxiv.org/pdf/1606.03498.pdf>.
- 4) Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", Proc. of the IEEE, pages 2278-2324, 1998.
- 5) 岡谷貴之,『深層学習』. 講談社, 2015.
- 6) 斎藤康毅,『ゼロから作る Deep Learning 2』オライリー・ジャパン. 2018.

- 7) S. Hochreiter and Jürgen Schmidhuber, “Long Short-Term Memory”, Journal Neural Computation, vol 9, pages 1735-1780, 1997.
- 8) L. Yu, W. Zhang, J.Wang, and Y. Yu, “SeqGAN: Sequence generative adversarial nets with policy gradient”, AAAI, pp.2852-2858, 2017.
- 9) Y. Zhang, Z. Gan, K.i Fan, Z.i Chen, R. Henaio, D. Shen, and L. Carin, “Adversarial Feature Matching for Text Generation”, 2017, <https://arxiv.org/abs/1706.03850>.
- 10) J. Guo, S. Lu, H. Cai, W. Zhang, Y. Yu and J. Wang, “Long Text Generation via Adversarial Training with Leaked Information”, 2017, <https://arxiv.org/abs/1709.08624>.
- 11) 廣田敦士, 岡夏樹, 荒木雅弘, 田中一晶, “学習データセットを分けた seqGAN による俳句生成”, 言語処理学会 第24回年次大会 発表論文集, 1292-1295, 2018.
- 12) Y. Zhu, S. Lu, L. Zheng, J. Guo, W. Zhang, J. Wang, and Y. Yu, “Texygen: A Benchmarking Platform for Text Generation Models”, 2018, <https://arxiv.org/abs/1802.01886>.
- 13) Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. (2002) BLEU: a method for Automatic Evaluation of Machine Translation. ACL.
- 14) Tomas Mikolov, “Efficient Estimation of Word Representations in Vector Space” arXiv:1301.3781.
- 15) 青空文庫 <https://www.aozora.gr.jp/>
- 16) 夏目漱石, 『坊っちゃん』, 筑摩書房, 1987.