

学修番号 17890539

## 修士論文

# 敵対的生成ネットワークを用いたニューラル機械翻訳と その評価

松村 雪桜

2019年2月22日

首都大学東京大学院  
システムデザイン研究科 情報通信システム学域

松村 雪桜

審査委員：

小町 守 准教授 (主指導教員)

山口 亨 教授 (副指導教員)

高間 康史 教授 (副指導教員)



# 敵対的生成ネットワークを用いたニューラル機械翻訳と その評価\*

松村 雪桜

## 修論要旨

近年、ニューラル機械翻訳 (Neural Machine Translation : NMT) の登場により、機械翻訳が盛んに研究されている。原言語文を中間表現に変換する Encoder, 及び、その中間表現から目的言語文を 1 単語ずつ生成する Decoder と呼ばれる 2 つの Recurrent Neural Network (RNN) を組み合わせた初期型の Encoder-Decoder が提案されて以来、原言語文の各単語を重み付けして考慮することができる Attention の導入等により、機械翻訳の精度は大きく向上してきた。しかしながら、翻訳時にいくつかの単語が翻訳されず消失してしまう、あるいは不必要な単語が出現してしまう、繰り返されてしまうといった現象がたびたび起きるという問題がある。

また、機械翻訳の評価には一般的に BLEU が用いられている。しかしながら、BLEU は単語 n-gram 適合率に基づき精度を評価する手法であり、単語の表層が異なる場合全く別の単語として評価されるため、文の意味を正しく考慮した評価はできていない。加えて、評価のために翻訳の正解である参照訳を必要とする。

画像生成の分野で注目を集めている敵対的生成ネットワーク (Generative Adversarial Network : GAN) は、Generator と Discriminator の 2 つのネットワークからなり、Discriminator はあるデータが正解データであるか Generator の出力であるかを識別する一方で、Generator は Discriminator が識別できないようなデータを生成するように敵対的な学習を行うことで、Generator が正解に近いデータを生成することを可能にしている。敵対的生成ネットワークは自然言語処理、とりわけ機械翻訳の分野でも使用が試みられており、これらの研究では Generator をニューラル機械翻訳モデル、Discriminator を入力された原言語文と目的言語文から目的言語文が参照訳であるかシステム出力文であるか予測する分類器として敵対

---

\*首都大学東京大学院 システムデザイン研究科 情報通信システム学域 修士論文, 学修番号 17890539, 2019 年 2 月 22 日.

的に学習を行うことで、Generator であるニューラル機械翻訳モデルの精度の向上を図っている。また、従来のニューラル機械翻訳では単語単位での最適化を行っているが、敵対的生成ネットワークを用いることで文単位の情報をも新たに用いて最適化することができ、文としてより自然なものを生成できることが期待される。

本研究では、日英および英日翻訳における敵対的生成ネットワークがニューラル機械翻訳に及ぼす影響を調査するとともに、目的言語文の分類器である Discriminator に注目し、Discriminator が予測する正解データらしさを機械翻訳の評価手法として用いることを提案する。敵対的生成ネットワークの設定では正解データは人手による参照訳なので、原言語文と目的言語文のペアを見て正解データらしいということは人手による翻訳である可能性が高いということであり、翻訳文の評価に転用できると考えられる。また、提案手法では評価時には原言語文と翻訳文のペアから評価を行うことから正解の参照訳を必要としないため、単言語コーパスなどの参照訳がない文に対する翻訳の評価への使用も期待できる。

敵対的生成ネットワークの構造としては Conditional Sequence GAN (CSGAN) を改良したものを実装し、学習における目的関数には 3 つの手法を用いた。1 つ目の手法は Goodfellow らによって最初に提案された GAN であり、Discriminator において目的言語文に対して正解データであるか生成データであるかの 2 クラス分類を行い、クロスエントロピーを用いて学習する。2 つ目の手法は Least Squares GAN (LSGAN) であり、従来の GAN とは異なり活性化関数を用いずに目的言語文の正解らしさを直接予測し、平均二乗誤差を用いて学習を行う。3 つ目の手法は Wasserstein GAN (WGAN) であり、Discriminator が正解データ（正例）に対して予測したスコアと生成データ（負例）に対して予測したスコアの差を用いて学習を行う。なお、敵対的生成ネットワークの学習は非常に不安定であり、どの手法においても Generator の事前学習を行った後にその出力を用いて Discriminator を事前学習し、最後に敵対的生成ネットワーク全体を学習している。

これら 3 つの手法について、Asian Scientific Paper Excerpt Corpus (ASPEC) を用いてそれぞれ実験を行い、BLEU 等の評価指標を用いてニューラル機械翻訳の精度を示し、敵対的生成ネットワークがニューラル機械翻訳に与える影響について考察を述べる。また、学習された Discriminator に原言語文と Workshop on Asian Translation (WAT) における実際の翻訳システムの出力文を入力することで、文

単位の翻訳精度を評価する。評価したスコアと人手評価スコアの相関をケンドールの順位相関を用いて示し，考察を述べる。

本論文の構成は以下のようになっている。第 1 章では本研究全体の提案，貢献，概要について述べる。第 2 章では先行研究とともにニューラル機械翻訳の基本的な構造について述べる。第 3 章では敵対的生成ネットワークに関する先行研究について述べる。第 4 章では敵対的生成ネットワークを用いた機械翻訳手法について述べる。第 5 章では敵対的生成ネットワークを用いた機械翻訳評価手法について述べる。第 6 章では，第 4, 5 章で述べた手法を用いた実験結果と考察について述べる。最後に，第 7 章で本研究のまとめについて述べる。

# Neural Machine Translation and its Evaluation using Generative Adversarial Network\*

Yukio Matsumura

## Abstract

In recent years, neural machine translation (NMT) has been researched all over the world. Once an encoder-decoder NMT which combines two recurrent neural networks (RNN) was proposed, NMT had gained great popularity in the machine translation community. However, the conventional encoder-decoder NMT works poorly on long sequences. Attention-based NMT can perform better prediction of output words by using the weights of each hidden state of the encoder as the context vector. It contributed to improvement of translation quality, especially in a long sentence. Nevertheless, NMT has many problems. For example, there are over-translation: some words are repeatedly translated or unnecessary words are generated and under-translation: some words are mistakenly untranslated.

In general, BLEU is used for the evaluation of machine translation. However, BLEU is an evaluation metric based on n-gram precision. Therefore, even if the system correctly translates a source sentence but the surface of word is different from the reference, BLEU evaluates the target sentence mistakenly. Furthermore, BLEU needs a reference.

Incidentally, Generative adversarial network (GAN) which is popular in the field of image generation consists of two networks; generator which generates a data and discriminator which distinguishes a generated data from true data. Discriminator is expected to distinguish the true or generated data, but generator aims to generate a data close to true data which discriminator cannot

---

\*Master's Thesis, Department of Information and Communication Systems, Graduate School of System Design, Tokyo Metropolitan University, Student ID 17890539, February 22, 2019.

distinguish. Generator can generate the data close to true data by adversarial training of these two networks. GAN is attempted to be used in the field of neural machine translation. Several previous studies regard NMT as generator, and the classifier as discriminator which distinguishes true sentence from generated sentence by the source sentence and target sentence. From this adversarial training, the quality of NMT is improved. Furthermore, an objective function of conventional NMT is optimized by word unit, so it is not guaranteed that an output of NMT is optimized as a sentence. Moreover, GAN NMT is optimized using sentence-level information, so GAN NMT is expected to generate natural sentence.

In this research, we examined the effect of GAN in Japanese-English and English-Japanese translation. Furthermore, we focus on discriminator which predicts the correctness of a sentence and proposed an evaluation method for machine translation using GAN. In this scenario, true data is the human translation. Therefore, the sentence which is predicted to be a correct sentence by discriminator is likely to be the translation which is close to the human translation, so it can be used for the evaluation of machine translation. This method does not need reference on evaluation because it only uses the pair of the source sentence and system output, so it is expected that the translation of the source sentence which does not have the target sentence, for example monolingual corpus, is evaluated.

In our experiments, we implemented the model architecture based on conditional sequence GAN (CSGAN), and trained it by three types of objective functions. The first is conventional GAN. It is trained using cross entropy. The second is least squares GAN (LSGAN). This method predicts the correctness of target sentence directly without using the activation function and it is trained using mean squared error. The last is Wasserstein GAN (WGAN). This method considers the difference of scores between true data and generated data on training. Note that we applied pre-training to both generator and discriminator using baseline in all methods because the training of GAN is very unstable. We ex-

perimented GAN NMT on Japanese-English and English-Japanese translation using Asian scientific paper excerpt corpus (ASPEC). Furthermore, we evaluate the output of translation system in the Workshop on Asian Translation (WAT) using GAN NMT. We confirmed the results in terms of both translation and evaluation quality.

This paper comprises as follows. In Section 1, we introduce the overview of research on neural machine translation. In Section 2, we describe the architecture of neural machine translation. In Section 3, we introduce the previous studies on generative adversarial network. In Section 4, we describe the architecture of generative adversarial network in neural machine translation. In Section 5, we propose the evaluation method using generative adversarial network for machine translation. In Section 6, we show the experimental results on the translation and evaluation using generative adversarial network. Finally, in Section 7, we describe the summary of this paper.

# 目次

図目次	ix
第 1 章 はじめに	1
第 2 章 ニューラル機械翻訳	4
2.1 Encoder . . . . .	4
2.2 Decoder . . . . .	5
2.3 学習 . . . . .	6
2.4 翻訳 . . . . .	6
2.4.1 ビームサーチ . . . . .	7
2.4.2 アンサンブル . . . . .	7
第 3 章 敵対的生成ネットワーク	8
第 4 章 敵対的生成ネットワークを用いたニューラル機械翻訳	9
4.1 Generator . . . . .	9
4.2 Discriminator . . . . .	9
4.3 学習 . . . . .	11
4.3.1 GAN . . . . .	12
4.3.2 Least Squares GAN . . . . .	12
4.3.3 Wasserstein GAN . . . . .	13
第 5 章 敵対的生成ネットワークを用いた機械翻訳評価手法	14
第 6 章 実験	15

6.1	翻訳実験 . . . . .	15
	6.1.1 データ . . . . .	15
	6.1.2 翻訳モデル . . . . .	15
	6.1.3 結果 . . . . .	16
6.2	評価実験 . . . . .	17
	6.2.1 データ . . . . .	17
	6.2.2 評価手法 . . . . .	18
	6.2.3 結果 . . . . .	18
第7章	おわりに	21
	発表リスト	22
	謝辞	25
	参考文献	26

# 目次

2.1	Encoder-Decoder. . . . .	5
4.1	敵対的生成ネットワークを用いたニューラル機械翻訳モデル. . . . .	10
4.2	Discriminator の構造. . . . .	11
5.1	敵対的生成ネットワークを用いた機械翻訳の評価. . . . .	14
6.1	人手評価スコアと GAN 評価スコアの散布図. . . . .	19
6.2	被評価文の文長と平均評価誤差の相関. . . . .	19

## 第 1 章 はじめに

近年、ニューラル機械翻訳 (Neural Machine Translation : NMT) の登場により、機械翻訳が盛んに研究されている。原言語文を中間表現に変換する Encoder, 及び、その中間表現から目的言語文を 1 単語ずつ生成する Decoder と呼ばれる 2 つの Recurrent Neural Network (RNN) を組み合わせた初期型の Encoder-Decoder が Sutskever ら [1], Cho ら [2] によって提案されて以来、ニューラル機械翻訳は機械翻訳モデルの主流になった。初期型の Encoder-Decoder には長文の翻訳に弱いという問題があったが、Decoder における各単語の予測時に原言語文の各単語に対応する Encoder の隠れ層を重み付けして考慮する Attention が Bahdanau ら [3], Luong ら [4] によって導入され、機械翻訳の精度は向上し、特に長文の翻訳精度に関しては大きく改善した。最近では、Vaswani ら [5] によって Transformer と呼ばれる新しいニューラル機械翻訳モデルが提案された。Transformer では、Attention を一般化した Self-attention や、単語分散表現に単語の位置情報を付加する Positional Encoding といった技術により、RNN や Convolutional Neural Network (CNN) を用いずにより高い翻訳精度を達成している。しかしながら、依然として翻訳時にいくつかの単語が翻訳されず消失してしまう、あるいは unnecessary 単語が出現してしまう、繰り返されてしまうといった現象がたびたび起きるという問題がある。このような現象は Over-translation, Under-translation と呼ばれている [6]。加えて、ニューラル機械翻訳の目的関数は単語単位で最適化されており、必ずしも文単位で最適な出力が保証されているとは言えない。これは、Over-translation や Under-translation を引き起こす原因となりうる。

また、機械翻訳の評価には一般的に BLEU [7] が用いられている。しかしながら、BLEU は単語 n-gram 適合率に基づき精度を評価する手法であり、単語の表層が異なる場合全く別の単語として評価されるため、たとえ正しく翻訳できていても使用する単語等が異なれば不当に低く評価してしまう可能性があり、文の意味を正しく考慮した評価はできていない [8]。加えて、評価のために翻訳の正解である参照訳を必要とする。

画像生成の分野で注目を集めている敵対的生成ネットワーク (Generative Adversarial Network : GAN) [9] は、Generator と Discriminator の 2 つのネットワー

クからなり，Discriminator はあるデータが正解データであるか Generator の出力であるかを識別する一方で，Generator は Discriminator が識別できないようなデータを生成するように敵対的な学習を行うことで，Generator が正解に近いデータを生成することを可能にしている．敵対的生成ネットワークは自然言語処理，とりわけ機械翻訳の分野でも使用が試みられており，これらの研究では Generator をニューラル機械翻訳モデル，Discriminator を入力された原言語文と目的言語文から目的言語文が参照訳であるかシステム出力文であるか予測する分類器として敵対的に学習を行うことで，Generator であるニューラル機械翻訳モデルの精度の向上を図っている．また，従来のニューラル機械翻訳では単語単位での最適化を行っているが，敵対的生成ネットワークを用いることで文単位の情報をも新たに用いて最適化することができ，文としてより自然なものを生成できることが期待される．

本研究では，日英および英日翻訳における敵対的生成ネットワークがニューラル機械翻訳に及ぼす影響を調査するとともに，目的言語文の分類器である Discriminator に注目し，Discriminator が予測する正解データらしさを機械翻訳の評価手法として用いることを提案する．敵対的生成ネットワークの設定では正解データは人手による参照訳なので，原言語文と目的言語文のペアを見て正解データらしいということは人手による翻訳である可能性が高いということであり，翻訳文の評価に転用できると考えられる．また，提案手法では評価時には原言語文と翻訳文のペアから評価を行うことから正解の参照訳を必要としないため，単言語コーパスなどの参照訳がない文に対する翻訳の評価への使用も期待できる．

敵対的生成ネットワークの構造としては Conditional Sequence GAN (CSGAN) [10] を改良したものを実装し，学習における目的関数には 3 つの手法を用いた．1 つ目の手法は Goodfellow らによって最初に提案された GAN [9] であり，Discriminator において目的言語文に対して正解データであるか生成データであるかの 2 クラス分類を行い，クロスエントロピーを用いて学習する．2 つ目の手法は Least Squares GAN (LSGAN) [11] であり，従来の GAN とは異なり活性化関数を用いずに目的言語文の正解らしさを直接予測し，平均二乗誤差を用いて学習を行う．3 つ目の手法は Wasserstein GAN (WGAN) [12] であり，Discriminator が正解データ（正例）に対して予測したスコアと生成データ（負例）に対して予測したスコアの差を用いて学習を行う．なお，敵対的生成ネットワークの学習は非常に不

安定であり，どの手法においても Generator の事前学習を行った後にその出力を用いて Discriminator を事前学習し，最後に敵対的生成ネットワーク全体を学習している．

これら 3 つの手法について，Asian Scientific Paper Excerpt Corpus (ASPEC) [13] を用いてそれぞれ実験を行い，BLEU を用いてニューラル機械翻訳の精度を示し，敵対的生成ネットワークがニューラル機械翻訳に与える影響について考察を述べる．また，学習された Discriminator に原言語文と Workshop on Asian Translation (WAT) における実際の翻訳システムの出力文を入力することで，文単位の翻訳精度を評価する．評価したスコアと人手評価スコアの相関をケンドールの順位相関を用いて示し，考察を述べる．

本研究の貢献は以下のとおりである．

- 敵対的生成ネットワークが日英および英日翻訳の品質に与える影響を調べ，より流暢な翻訳ができることを示した．
- 敵対的生成ネットワークを機械翻訳の評価に用いる手法を提案した．
- 敵対的生成ネットワークを用いて日英翻訳の評価を行い，統計的機械翻訳 (Statistical Machine Translation : SMT) の評価において人手評価との相関を示した．
- 実験に使用した実装を公開した.\*

本論文の構成は以下のようになっている．第 1 章では本研究全体の提案，貢献，概要について述べる．第 2 章では先行研究とともにニューラル機械翻訳の基本的な構造について述べる．第 3 章では敵対的生成ネットワークに関する先行研究について述べる．第 4 章では敵対的生成ネットワークを用いた機械翻訳手法について述べる．第 5 章では敵対的生成ネットワークを用いた機械翻訳評価手法について述べる．第 6 章では，第 4, 5 章で述べた手法を用いた実験結果と考察について述べる．最後に，第 7 章で本研究のまとめについて述べる．

---

\*<https://github.com/yukio326/GAN-NMT>

## 第 2 章 ニューラル機械翻訳

ここで、本研究に用いた Encoder-Decoder と呼ばれるニューラル機械翻訳モデルについて説明する。我々は一般的に使用されている Luong ら [4] の手法を基に実装を行った。図 2.1 に示すように Encoder-Decoder は Encoder と Decoder の 2 つのネットワークからなる。Encoder には双方向 Long Short-term Memory (LSTM) を、Decoder には単方向 LSTM を使用している。

### 2.1 Encoder

入力された原言語文は、one-hot ベクトル系列 ( $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_{|\mathbf{X}|}]$ ) に変換される。ここで、 $|\mathbf{X}|$  は原言語文の文長を表す。各ステップ  $i$  において、原言語側の単語の埋め込み表現  $\mathbf{e}_i^s$  は、

$$\mathbf{e}_i^s = \tanh(\mathbf{W}_x \mathbf{x}_i) \quad (2.11)$$

と計算される。ここで、 $\mathbf{W}_x \in \mathbb{R}^{q \times v_s}$  は重み行列であり、 $q$  は埋め込み層の次元数、 $v_s$  は原言語側の語彙サイズを表す。Encoder の隠れ層  $\bar{\mathbf{h}}_i^s$  は、

$$\bar{\mathbf{h}}_i = \vec{\mathbf{h}}_i^{(L)} + \overleftarrow{\mathbf{h}}_i^{(L)} \quad (2.12)$$

と表され、 $L$  は層の数を示す。順方向の隠れ層  $\vec{\mathbf{h}}_i^{(l)}$  と逆方向の隠れ層  $\overleftarrow{\mathbf{h}}_i^{(l)}$  はそれぞれ LSTM を用いて、

$$\vec{\mathbf{h}}_i^{(l)} = \text{LSTM}(\vec{\mathbf{h}}_i^{(l-1)}, \vec{\mathbf{h}}_{i-1}^{(l)}) \quad (2.13)$$

および

$$\overleftarrow{\mathbf{h}}_i^{(l)} = \text{LSTM}(\overleftarrow{\mathbf{h}}_i^{(l-1)}, \overleftarrow{\mathbf{h}}_{i+1}^{(l)}) \quad (2.14)$$

と計算され、 $l$  は層の番号を示す。ここで、1 層目の隠れ層の計算に用いる  $\vec{\mathbf{h}}_i^{(0)}$  および  $\overleftarrow{\mathbf{h}}_i^{(0)}$  には単語埋め込み表現  $\mathbf{e}_i^s$  を用いる。

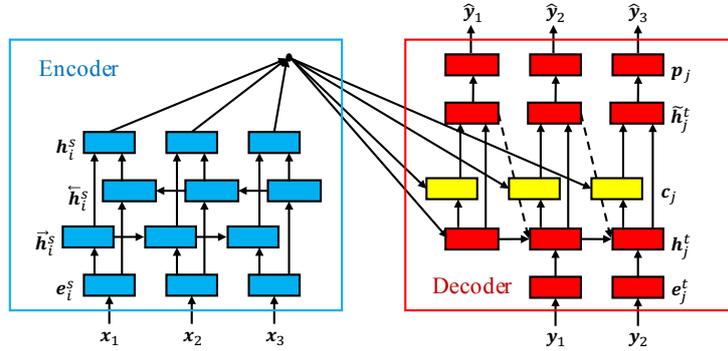


図 2.1: Encoder-Decoder.

## 2.2 Decoder

原言語文と同様に，入力された目的言語文は，one-hot ベクトル系列 ( $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_{|\mathbf{Y}|}]$ ) に変換される．ここで， $|\mathbf{Y}|$  は目的言語文の文長を表す．各ステップ  $j$  において，Decoder の隠れ層  $\mathbf{h}_j^{(l)}$  は，LSTM を用いて，

$$\mathbf{h}_j^{(l)} = \text{LSTM}(\mathbf{h}_j^{(l-1)}, \mathbf{h}_{j-1}^{(l)}) \quad (2.21)$$

と表される．

ここで，1 層目の隠れ層の計算に用いる  $\mathbf{h}_j^{(0)}$  には，1 ステップ前の単語埋め込み表現  $\mathbf{e}_{j-1}^t$  と文脈情報付き隠れ層  $\tilde{\mathbf{h}}_{j-1}$  を結合した  $[\mathbf{e}_{j-1}^t : \tilde{\mathbf{h}}_{j-1}]$  を用いる．各層の最初のステップの隠れ層  $\mathbf{h}_1^{(l)}$  は，

$$\mathbf{h}_1^{(l)} = \overrightarrow{\mathbf{h}}_{|\mathbf{X}|}^{(l)} + \overleftarrow{\mathbf{h}}_1^{(l)} \quad (2.22)$$

のように計算される．

目的言語側の単語埋め込み表現  $\mathbf{e}_j^t$  は，

$$\mathbf{e}_j^t = \tanh(\mathbf{W}_y \mathbf{y}_j) \quad (2.23)$$

と計算される．単語  $\mathbf{y}_j$  は，学習時には正解の単語の one-hot ベクトルを用い，評価時には予測した単語  $\hat{\mathbf{y}}_j$  の one-hot ベクトルを用いる．ここで， $\mathbf{W}_y \in \mathbb{R}^{q \times v_t}$  は重み行列であり， $v_t$  は原言語側の語彙サイズを表す．文脈情報付き隠れ層  $\tilde{\mathbf{h}}_j$  は，

$$\tilde{\mathbf{h}}_j = \tanh(\mathbf{W}_a [\mathbf{h}_j^{(L)} : \mathbf{c}_j] + \mathbf{b}_a) \quad (2.24)$$

のように計算される。ここで、 $\mathbf{W}_a \in \mathbb{R}^{r \times 2r}$  は重み行列、 $\mathbf{b}_a \in \mathbb{R}^r$  はバイアスであり、 $r$  は隠れ層の次元数を表す。

文脈ベクトル  $\mathbf{c}_j$  は Encoder の各隠れ層  $\bar{\mathbf{h}}_i$  の重み付き和であり、

$$\mathbf{c}_j = \sum_{i=1}^{|\mathbf{X}|} \alpha_{ij} \bar{\mathbf{h}}_i \quad (2.25)$$

と表される。上式における重み  $\alpha_{ij}$  は、ソフトマックス関数を用いて全体の和が 1 となるよう正規化される確率分布であり、以下のように内積を用いて計算される。

$$\alpha_{ij} = \frac{\exp(\bar{\mathbf{h}}_i^T \mathbf{h}_j^{(L)})}{\sum_{k=1}^{|\mathbf{X}|} \exp(\bar{\mathbf{h}}_k^T \mathbf{h}_j^{(L)})}. \quad (2.26)$$

出力単語  $\hat{\mathbf{y}}_j$  の条件付き確率は、

$$p(\hat{\mathbf{y}}_j | \mathbf{Y}_{<j}, \mathbf{X}) = \text{softmax}(\mathbf{W}_p \tilde{\mathbf{h}}_j + \mathbf{b}_p) \quad (2.27)$$

と表される。ここで、 $\mathbf{W}_p \in \mathbb{R}^{v_t \times r}$  は重み行列、 $\mathbf{b}_p \in \mathbb{R}^{v_t}$  はバイアスである。

## 2.3 学習

学習時における目的関数は、

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{D} \sum_{d=1}^D \sum_{j=1}^{|\mathbf{Y}|} \log p(\mathbf{y}_j^{(d)} | \mathbf{Y}_{<j}^{(d)}, \mathbf{X}^{(d)}, \boldsymbol{\theta}) \quad (2.31)$$

のように定義する。ここで、 $\boldsymbol{\theta}$  はモデルにおける全てのパラメータとする。一般的に単語埋め込みに関するモデルパラメータは、事前に Word2vec [14, 15] や GloVe [16] を用いて学習されたものを初期値として用いる。その他のモデルパラメータはランダムに初期化される。

## 2.4 翻訳

ニューラル機械翻訳モデルでは各ステップでただ一つの出力単語が予測される。その単語が正しいという前提のもとで次の出力単語が予測されるため、一度誤った

単語が出力されてしまうとその後続く単語を正しく出力することが難しくなってしまう。より頑健な単語予測を行うために、実際の翻訳時には一般的にビームサーチ及びアンサンブルと呼ばれる2つの技術が用いられる。

#### 2.4.1 ビームサーチ

ビームサーチでは、各ステップにおいて最も確率の高い1単語だけではなく確率の高い $n$ 個の候補を保持しておき、次のステップではそれぞれの候補に対して単語予測の計算を行い前のステップと同様に $n$ 個の候補をそれぞれ求める。合計 $n^2$ 個の候補のうち確率の高い $n$ 個だけを残し次のステップに進む。これにより、誤った文を生成するリスクを軽減できる。

#### 2.4.2 アンサンブル

アンサンブルでは複数のモデルを用意する。出力単語 $\hat{y}_j$ の条件付き確率は各モデルの予測確率 $p^{(m)}$ の平均を用いて、

$$p(\hat{y}_j | \mathbf{Y}_{<j}, \mathbf{X}) = \frac{1}{M} \sum_{m=1}^M p^{(m)}(\hat{y}_j | \mathbf{Y}_{<j}, \mathbf{X}) \quad (2.41)$$

と表される。ここで、 $M$ はモデル数である。これにより、誤った単語を生成するリスクを軽減できる。

### 第 3 章 敵対的生成ネットワーク

敵対的生成ネットワークは、Goodfellow ら [9] によって提案された画像生成ネットワークである。敵対的生成ネットワークは、ノイズから画像を生成する Generator とある画像が真の画像であるか機械が生成した画像であるかを識別する Discriminator の 2 つのネットワークからなる。Discriminator は画像を正しく識別する一方で、Generator は Discriminator が識別できないような画像を生成するように敵対的な学習を行うことで、Generator が本物に近い画像を生成することを可能にしている。一般的に Discriminator は CNN を用いて構成される。

画像生成分野における敵対的生成ネットワークの研究は非常に盛んに行われており、多くの種類の敵対的生成ネットワークが新しく提案されている。Mao ら [11] は、従来の GAN の目的関数には勾配消失を引き起こす可能性を指摘し、Least Squares GAN (LSGAN) を提案した。LSGAN では、sigmoid 関数のような活性化関数は用いずにスコアを直接予測し、二乗誤差を用いて最適化を行う。Arjovsky ら [12] は、Discriminator が正解データに対して予測するスコアと生成データに対して予測するスコアの差を最大化する Wasserstein GAN を提案した。

最近では、自然言語処理、とりわけ機械翻訳の分野でも敵対的生成ネットワークの使用が試みられている [17, 10, 18]。画像生成では Generator はノイズから画像を生成していたが、機械翻訳では原言語文から目的言語を生成する、すなわちニューラル機械翻訳そのものとなっている。また、Discriminator は Generator が生成した目的言語文からだけでは翻訳として正しいかの判断ができないため、対応する原言語文も合わせて入力している。Yang ら [10] は CNN だけでなく RNN を用いた Discriminator についても提案し、敵対的生成ネットワークを使うことでニューラル機械翻訳の精度が向上することを示した。

また、従来の敵対的生成ネットワークで行っていた、Generator が生成するデータそのものを直接 Discriminator が識別するという仕組みではなく、ネットワーク内のベクトル表現等を Discriminator で識別してそれを騙すように敵対的な学習を行うことでより頑健な表現を作り出すような仕組みも提案されている [19, 20]。

## 第 4 章 敵対的生成ネットワークを用いたニューラル機械翻訳

ここでは、提案手法として用いた敵対的生成ネットワークを用いたニューラル機械翻訳モデルの構造について述べる。我々は Yang ら [10] における RNN モデルを改良した実装を行った。前章で述べたように、敵対的生成ネットワークは Generator と Discriminator の 2 つのネットワークから成る。図 4.1 に敵対的生成ネットワークを用いたニューラル機械翻訳モデルの概要を示す。

### 4.1 Generator

Generator は、入力された原言語文から Discriminator がシステムの出力文であると判断できないような正解データ（参照訳）に近い文の生成を図る。本研究では、2 章で述べたニューラル機械翻訳モデルをそのまま Generator として用いる。

### 4.2 Discriminator

Discriminator は、図 4.2 に示したように、入力された原言語文と目的言語文から、その目的言語文の正解らしさを予測する。すなわち、目的言語文として参照訳が入力された場合は高いスコアを、Generator の出力文が入力された場合は低いスコアを予測することが Discriminator の目的である。

入力された原言語文の各単語は、式 (2.11) に従ってそれぞれ単語埋め込み表現  $e_i^s$  へと変換される。各ステップ  $i$  において、原言語側の単語埋め込み表現  $e_i^s$  に対応する隠れ層  $f_i^s$  は、

$$f_i^s = \vec{f}_i^{s(L)} + \overleftarrow{f}_i^{s(L)}. \quad (4.21)$$

と表され、順方向の隠れ層  $\vec{f}_i^{s(l)}$  と逆方向の隠れ層  $\overleftarrow{f}_i^{s(l)}$  は、それぞれ LSTM を用いて、

$$\vec{f}_i^{s(l)} = \text{LSTM}(\vec{f}_i^{s(l-1)}, \overleftarrow{f}_{i-1}^{s(l)}) \quad (4.22)$$

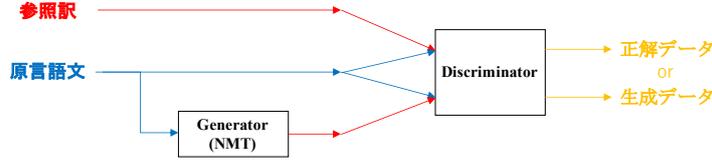


図 4.1: 敵対的生成ネットワークを用いたニューラル機械翻訳モデル.

および

$$\overleftarrow{f}_i^s(l) = \text{LSTM}(\overleftarrow{f}_i^s(l-1), \overleftarrow{f}_{i+1}^s(l)) \quad (4.23)$$

と計算される．ここで，1層目の隠れ層の計算に用いる  $\overrightarrow{f}_i^s(0)$  および  $\overleftarrow{f}_i^s(0)$  には単語埋め込み表現  $e_i^s$  を用いる．得られた隠れ層から原言語文の文ベクトル  $\bar{f}^s$  を，

$$\bar{f}^s = \text{average} \left( \left[ f_1^s, f_2^s, \dots, f_{|X|}^s \right] \right). \quad (4.24)$$

のように求める．

同様に，入力された目的言語文の各単語は，式 (2.23) に従ってそれぞれ単語埋め込み表現  $e_j^t$  へと変換される．各ステップ  $j$  において，目的言語側の単語埋め込み表現  $e_j^t$  に対応する隠れ層  $f_j^t$  は，

$$f_j^t = \overrightarrow{f}_j^t(L) + \overleftarrow{f}_j^t(L). \quad (4.25)$$

と表され，順方向の隠れ層  $\overrightarrow{f}_j^t(l)$  と逆方向の隠れ層  $\overleftarrow{f}_j^t(l)$  は，それぞれ LSTM を用いて，

$$\overrightarrow{f}_j^t(l) = \text{LSTM}(\overrightarrow{f}_j^t(l-1), \overrightarrow{f}_{j-1}^t(l)) \quad (4.26)$$

および

$$\overleftarrow{f}_j^t(l) = \text{LSTM}(\overleftarrow{f}_j^t(l-1), \overleftarrow{f}_{j+1}^t(l)). \quad (4.27)$$

と計算される．ここで，1層目の隠れ層の計算に用いる  $\overrightarrow{f}_j^t(0)$  および  $\overleftarrow{f}_j^t(0)$  には単語埋め込み表現  $e_i^t$  を用いる．得られた隠れ層から目的言語文の文ベクトル  $\bar{f}^t$  を，

$$\bar{f}^t = \text{average} \left( \left[ f_1^t, f_2^t, \dots, f_{|Y|}^t \right] \right). \quad (4.28)$$

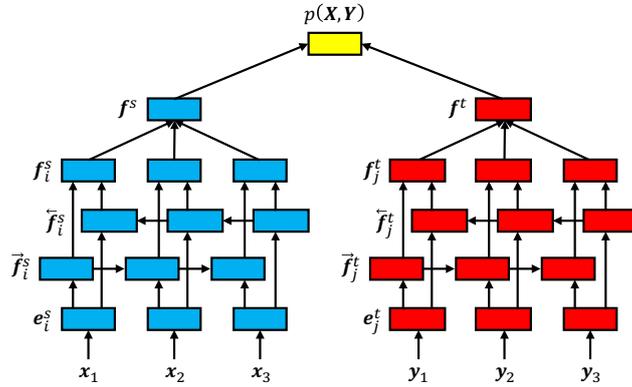


図 4.2: Discriminator の構造.

のように求める.

最後に, 求めた原言語文および目的言語文の文ベクトルの内積を用いて, 入力された原言語文に対する目的言語文の正解らしさを,

$$p(\mathbf{X}, \mathbf{Y}) = \text{sigmoid}(\bar{\mathbf{f}}^s \cdot \bar{\mathbf{f}}^t). \quad (4.29)$$

のように予測する.

### 4.3 学習

敵対的生成ネットワークでは, Generator は Discriminator を騙すように, Discriminator は Generator の出力文を区別できるように敵対的な学習を行わなければならない. したがって, 学習時における目的関数は Generator と Discriminator で異なる. 本研究では, 一般的な GAN, Least Squares GAN (LSGAN), Wasserstein GAN (WGAN) の 3 つの手法を参考に 3 種類の目的関数を実装した. なお, 予測系列を  $\hat{\mathbf{Y}} = [\hat{y}_1, \dots, \hat{y}_{|\hat{\mathbf{Y}}|}]$ ,  $\theta$  を Generator における全てのパラメータ,  $\gamma$  を Discriminator における全てのパラメータとする.

### 4.3.1 GAN

Goodfellow ら [9] によって最初に提案された一般的な敵対的生成ネットワークの目的関数について述べる．最適化にはクロスエントロピーを用いており，学習時の Generator の目的関数  $\mathcal{L}_G(\boldsymbol{\theta}, \gamma)$  および Discriminator の目的関数  $\mathcal{L}_D(\gamma)$  はそれぞれ，

$$\mathcal{L}_G(\boldsymbol{\theta}, \gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ \sum_{j=1}^{|\mathbf{Y}|} \log p(\mathbf{y}_j^{(d)} | \mathbf{Y}_{<j}^{(d)}, \mathbf{X}^{(d)}, \boldsymbol{\theta}) + \log p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma) \right\} \quad (4.31)$$

$$\mathcal{L}_D(\gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ \log p(\mathbf{X}^{(d)}, \mathbf{Y}^{(d)} | \gamma) + \log \left\{ 1 - p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma) \right\} \right\} \quad (4.32)$$

のように定義する．

### 4.3.2 Least Squares GAN

Mao ら [11] は，従来の GAN の目的関数には勾配消失を引き起こす可能性を指摘し，LSGAN を提案した．LSGAN では，式 4.29 における sigmoid 関数のような活性化関数はいずれにスコアを直接予測する．最適化には二乗誤差を用いており，学習時の Generator の目的関数  $\mathcal{L}_G(\boldsymbol{\theta}, \gamma)$  および Discriminator の目的関数  $\mathcal{L}_D(\gamma)$  はそれぞれ，

$$\mathcal{L}_G(\boldsymbol{\theta}, \gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ \sum_{j=1}^{|\mathbf{Y}|} \log p(\mathbf{y}_j^{(d)} | \mathbf{Y}_{<j}^{(d)}, \mathbf{X}^{(d)}, \boldsymbol{\theta}) - \frac{1}{2} (1 - p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma))^2 \right\} \quad (4.33)$$

$$\mathcal{L}_D(\gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ -\frac{1}{2} (1 - p(\mathbf{X}^{(d)}, \mathbf{Y}^{(d)} | \gamma))^2 - \frac{1}{2} p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma)^2 \right\} \quad (4.34)$$

のように定義する．

### 4.3.3 Wasserstein GAN

Arjovsky ら [12] は, Discriminator が正解データに対して予測するスコアと生成データに対して予測するスコアの差を最大化する WGAN を提案した. 学習時の Generator の目的関数  $\mathcal{L}_G(\boldsymbol{\theta}, \gamma)$  および Discriminator の目的関数  $\mathcal{L}_D(\gamma)$  はそれぞれ,

$$\mathcal{L}_G(\boldsymbol{\theta}, \gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ \sum_{j=1}^{|\mathbf{Y}|} \log p(\mathbf{y}_j^{(d)} | \mathbf{Y}_{<j}^{(d)}, \mathbf{X}^{(d)}, \boldsymbol{\theta}) + p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma) \right\} \quad (4.35)$$

$$\mathcal{L}_D(\gamma) = \frac{1}{D} \sum_{d=1}^D \left\{ p(\mathbf{X}^{(d)}, \mathbf{Y}^{(d)} | \gamma) - p(\mathbf{X}^{(d)}, \hat{\mathbf{Y}}^{(d)} | \gamma) \right\} \quad (4.36)$$

のように定義する.

## 第5章 敵対的生成ネットワークを用いた機械翻訳評価手法

敵対的生成ネットワークを用いる一般的な目的は、Generator は Discriminator がシステムの出力文であると判断できないような正解データに近い文を生成するように、Discriminator は入力された原言語文と目的言語文からその目的言語文の正解らしさを予測し、Generator の出力文を区別できるように敵対的な学習を行うことで、Generator の出力文の質を向上させることにある。本研究では、Generator ではなく Discriminator に注目し、学習された Discriminator を用いて機械翻訳の評価を行うことを提案する。

提案手法では、はじめに Generator (NMT) の事前学習を行い、得られたモデルの出力を用いて Discriminator の事前学習を行う。その後、敵対的生成ネットワーク全体を学習し、図 5.1 に示すように、学習された Discriminator に原言語文とシステムの出力文を入力することで、翻訳精度を評価する。敵対的生成ネットワークにおいて、Discriminator は Generator の出力文がどれほど正解データに近いかを判断しており、翻訳精度を評価できることが期待される。

また、提案手法では評価に原言語文とシステムの出力文を用いており正解の参照訳を必要としないため、単言語コーパスなどの参照訳がない文に対する翻訳の評価への使用も期待できる。



図 5.1: 敵対的生成ネットワークを用いた機械翻訳の評価。

## 第 6 章 実験

本研究では，日英および英日翻訳における敵対的生成ネットワークがニューラル機械翻訳に及ぼす影響を調査するための翻訳実験と，敵対的生成ネットワークを機械翻訳の評価手法として用いる評価実験の 2 種類の実験を行う。

### 6.1 翻訳実験

#### 6.1.1 データ

実験に使用したコーパスは，Asian Scientific Paper Excerpt Corpus (ASPEC) [13] の日英コーパスである。学習用データに関しては，約 300 万文のうち文アライメントの信頼度上位 100 万文を用いた。日本語の単語分割には形態素解析器 MeCab\* (バージョン 0.996, IPADIC) を用い，英語の単語分割には Moses<sup>†</sup> の tokenizer.perl を用いた。原言語および目的言語の学習用データから 1 文あたり 60 単語を超える文対を削除したところ，コーパスの文数は表 6.1 のようになった。

#### 6.1.2 翻訳モデル

本実験では，ベースラインである一般的なニューラル機械翻訳モデルである Encoder-Decoder と敵対的生成ネットワークを用いた機械翻訳モデルを用いて実験を行った。なお，ネットワークは，原言語側語彙サイズ 100,000，目的言語側語彙サイズ 30,000，埋め込み層次元数 512，Encoder の層数 3，Decoder の層数 3，隠れ層次元数 512，バッチサイズ 128，ビーム幅 20 の設定で実験を行った。最適化手法については AdaGrad (初期学習率 : 0.01) で学習した後に，SGD を用いてチューニングを行った。

---

\*<https://github.com/taku910/mecab>

†<http://www.statmt.org/moses/>

学習用	974,198
検証用	1,790
評価用	1,812

表 6.1: ASPEC 日英コーパスの文数.

Model	BLEU	RIBES	AMFM
ベースライン	24.94	0.757955	0.596590
GAN NMT	25.17	0.757413	0.595850

表 6.2: 日英翻訳結果.

Model	BLEU	RIBES	AMFM
ベースライン	35.17	0.827386	0.749190
GAN NMT	35.09	0.827650	0.750350

表 6.3: 英日翻訳結果.

原言語文	Blood collection is indispensable for glucose level measurement for the diabetes mellitus diagnosis at present.
ベースライン	糖尿病診断のためには血糖値測定には採血が不可欠である。
GAN NMT	現在糖尿病診断のための血糖値測定には採血が必須である。
参照訳	糖尿病診断のための血糖値測定は、現在、採血が不可欠である。

表 6.4: 英日翻訳の出力例.

### 6.1.3 結果

表 6.2, 6.3 に, BLEU [7], RIBES [21], AMFM [22] の 3 つの評価指標を用いて, 日英翻訳と英日翻訳それぞれの翻訳精度を示す. 翻訳精度としては, ベースラインと GAN NMT でそこまで差がなかった.

表 6.4 に英日翻訳の出力例を示す. ベースラインでは, “for the diabetes mellitus diagnosis at present” が “糖尿病診断のためには” と翻訳されているが, この例のよ

	妥当性	流暢性
GAN NMT > ベースライン	16	23
GAN NMT = ベースライン	72	72
GAN NMT < ベースライン	12	5
total	100	100

表 6.5: ベースラインと GAN NMT のペアワイズ評価.

うに名詞句を修飾する場合本来は“糖尿病診断のための”と翻訳されるべきである。また，“at present”が翻訳されずに消失してしまっている。一方で，GAN NMT は正しく翻訳できていることがわかる。

敵対的生成ネットワークが翻訳品質に与える影響を調べるために，ペアワイズ評価を行った。翻訳の人手評価では，一般的に妥当性と流暢性の観点から評価が行われるため，この2つの観点から評価を行った。表 6.5 に，ペアワイズ評価結果を示す。合計 100 例のうち GAN NMT のほうが優れている，同等の質である，ベースラインのほうが優れているの 3 つの場合それぞれの事例数を示す。妥当性に関しては，両者に大きな差はなかったが，流暢性に関しては GAN NMT が優れていることがわかった。このように，敵対的生成ネットワークは Discriminator を用いて NMT の出力を正解らしい人間の翻訳に近づけることで，より流暢な翻訳ができることがわかった。

## 6.2 評価実験

### 6.2.1 データ

実験に使用したコーパスは翻訳実験と同様 ASPEC の日英コーパスであり，表 6.1 の学習用コーパスを用いて学習を行った。

翻訳の評価には，WAT 2015 [23] および WAT 2017 [24] の日英機械翻訳の人手評価データセットを用いた。このデータはそれぞれ，ASPEC の評価用データ中から抽出された 200 組の対訳文と，それぞれに対する 3 システム分の出力文・2 人分の人手評価スコア（1～5）からなる。人手評価スコアは，2 人分の評価スコアの平

評価手法	WAT 2015	WAT 2017
Sentence BLEU	0.234	0.305
CSGAN	0.195	0.105
LSGAN	-	0.112
WGAN	-	0.087

表 6.6: ケンドールの順位相関係数

均値を正規化して用いた。

### 6.2.2 評価手法

本実験では、200 文 × 3 システムの合計 600 文に対して、ベースライン (Sentence BLEU [25]) および提案手法 (GAN, LSGAN, WGAN) を用いて文レベルで評価を行った。なお、ネットワークは、原言語側語彙サイズ 30,000, 目的言語側語彙サイズ 30,000, 埋め込み層次元数 512, Generator の層数 2, Discriminator の層数 2, Generator の隠れ層次元数 512, Discriminator の隠れ層次元数 512, バッチサイズ 128, 最適化手法 AdaGrad (初期学習率 : 0.01) の設定で実験を行った。

### 6.2.3 結果

実験で得られたスコアについて、人手評価との相関を測った。相関を示すスコアとして、ケンドールの順位相関係数を使用した。結果は表 6.6 の通りである。また、人手評価スコアと GAN 評価スコアの散布図を図 6.1 に示す。実験の結果、WAT 2015 のデータセットでは、敵対的生成ネットワークを用いた評価は参照訳を用いていないものの、Sentence BLEU とほぼ同程度の手人評価との相関を示した。一方で、WAT 2017 のデータセットでは、ほぼ無相関であることがわかった。WAT 2015 におけるシステムは SMT であったが WAT 2017 では NMT が主流となっており、評価対象である機械翻訳モデルの種類によって評価できる場合とそうでない場合があることがわかる。

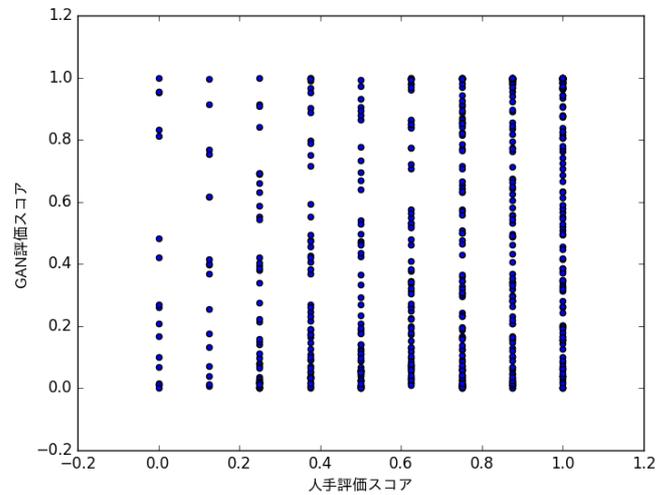


図 6.1: 人手評価スコアと GAN 評価スコアの散布図.

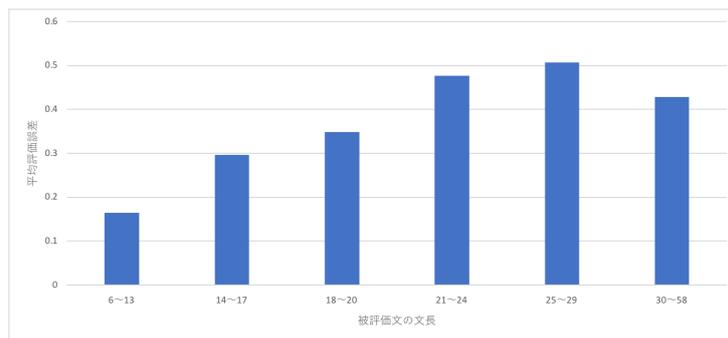


図 6.2: 被評価文の文長と平均評価誤差の相関.

表 6.7 に WAT 2015 のデータセットにおける実際の評価スコア例を示す．成功例において，参照訳は受動態の文となっているが，システム出力文は能動態になっている．また，原言語文の“施行した”に対応する動詞が参照訳では“applied”，システム出力文では“underwent”となっており表層が異なるため，Sentence BLEU スコアでは低く評価されているが，GAN 評価スコアでは人手評価と同様にほぼ正解と評価できている．

しかし失敗例においては，参照訳とシステム出力文において意味も表層もほぼ同じであり，人手評価スコアも Sentence BLEU スコアも高い評価ができているにも

成功例	
原言語文	両症例とも保存療法を施行した。
参照訳	The conservative treatment was applied to both cases .
システム出力文	Both cases underwent conservative treatment .
人手評価 : <b>1.000</b> , Sentence BLEU : <b>0.232</b> , GAN 評価 : <b>0.999</b>	
失敗例	
原言語文	次世代 ネットワーク 下での ネットワーク 管理 システム に対する 要求 を 論じ , NEC により 提供 される 幾つか の 解決 策 を 紹介 した 。
参照訳	Demands for the network management systems under the next generation networks are discussed , and some solutions offered by NEC are introduced .
システム出力文	The demand for the network management system under the next generation network is discussed, and some solutions offered by NEC are introduced .
人手評価 : <b>1.000</b> , Sentence BLEU : <b>0.644</b> , GAN 評価 : <b>0.087</b>	

表 6.7: 評価スコア例.

かかわらず, GAN 評価スコアでは非常に低い評価をしてしまっている. この例のほかに, 特に長文や複文において誤った評価をしている例が多数見受けられた. 図 6.2 に被評価文の文長と平均評価誤差の相関を示す. 文長の範囲はそれぞれ約 100 文になるように定めている. 図からも被評価文の文長が長いほど評価を誤る傾向があることがわかる. 原因としては, 文ベクトルの作成において各単語に対応する隠れ層の平均を用いているため, 文長が長い場合には文ベクトルを作成した段階で情報が多く欠落してしまうことが考えられる. 特に, 述語は文ベクトルを構成する上で重要度が高いと考えられ, 複文のように述語が複数ある場合には, 文ベクトルを作成した段階で非常に多くの情報が欠落してしまっている可能性が高いと推測される.

## 第 7 章 おわりに

本研究では，日英翻訳における敵対的生成ネットワークがニューラル機械翻訳に及ぼす影響を調査するとともに，目的言語文の分類器である Discriminator に注目し，Discriminator が予測する正解データらしさを機械翻訳の評価手法として用いることを提案した．翻訳実験の結果，日英および英日翻訳において，敵対的生成ネットワークを用いることで，ニューラル機械翻訳の流暢性を向上させることを確認した．また，評価実験の結果，敵対的生成ネットワークを用いることで統計的機械翻訳の評価では人手評価との相関を確認できたものの，ニューラル機械翻訳の評価には向いていないことがわかった．今後は，ニューラル機械翻訳の評価にも適した手法を検討したい．

# 発表リスト

## 国際会議

1. Yukio Matsumura, Takayuki Sato and Mamoru Komachi. **English-Japanese Neural Machine Translation with Encoder-Decoder-Reconstructor**. In arXiv e-prints, 1706.08198. June 27, 2017.
2. Yukio Matsumura and Mamoru Komachi. **Tokyo Metropolitan University Neural Machine Translation System for WAT 2017**. In Proceedings of the 4th Workshop on Asian Translation (WAT 2017). Taipei, Taiwan. November 27, 2017.
3. Michiki Kurosawa, Yukio Matsumura, Hayahide Yamagishi and Mamoru Komachi. **Japanese Predicate Conjugation for Neural Machine Translation**. In Proceedings of NAACL 2018 Student Research Workshop (NAACL-SRW 2018), pp.100-105. New Orleans, Louisiana, USA. June 2, 2018.
4. Satoru Katsumata, Yukio Matsumura, Hayahide Yamagishi and Mamoru Komachi. **Graph-based Filtering of Out-of-Vocabulary Words for Encoder-Decoder Model**. In Proceedings of ACL 2018 Student Research Workshop (ACL-SRW 2018), pp.112-119. Melbourne, Australia. July 17, 2018.
5. Yukio Matsumura, Satoru Katsumata, Mamoru Komachi. **TMU Japanese-English Neural Machine Translation System using Generative Adversarial Network for WAT 2018**. In Proceedings of the 5th Workshop on Asian Translation (WAT 2018). Hong Kong.

December 3, 2018.

#### 国内会議

1. 松村雪桜, 佐藤貴之, 叶内晨, 小町守. 文節へのアテンションを用いたニューラル日英翻訳. NLP 若手の会第 11 回シンポジウム (YANS 2016). 和歌山. August 28, 2016.
2. 松村雪桜, 佐藤貴之, 小町守. 逆翻訳によるニューラル機械翻訳の最適化. 言語処理学会第 23 回年次大会 (NLP 2017), pp.986-989. つくば. March 16, 2017.
3. 嶋中宏希, 山岸駿秀, 松村雪桜, 小町守. クロスリンガルな単語分散表現を用いた機械翻訳自動評価手法の検討. NLP 若手の会第 12 回シンポジウム (YANS 2017). 那覇. September 3, 2017.
4. 勝又智, 松村雪桜, 山岸駿秀, 小町守. ニューラル日英翻訳における RNN モデルと CNN モデルの出力分析. NLP 若手の会第 12 回シンポジウム (YANS 2017). 那覇. September 4, 2017.
5. 黒澤道希, 山岸駿秀, 松村雪桜, 小町守. 活用情報を用いた日英ニューラル機械翻訳. NLP 若手の会第 12 回シンポジウム (YANS 2017). 那覇. September 4, 2017.
6. 松村雪桜, 小町守. 敵対的生成ネットワークを用いた機械翻訳評価手法. 言語処理学会第 24 回年次大会 (NLP 2018), pp.568-571. 岡山. March 14, 2018.
7. 黒澤道希, 松村雪桜, 山岸駿秀, 小町守. 述語の活用情報を用いたニューラル日英翻訳. 言語処理学会第 24 回年次大会 (NLP 2018), pp.813-816. 岡山. March 14, 2018.
8. 勝又智, 松村雪桜, 山岸駿秀, 小町守. ニューラル機械翻訳における共起情報を考慮した語彙選択. 言語処理学会第 24 回年次大会 (NLP 2018), pp.1058-1061. 岡山. March 15, 2018.
9. 吉村綾馬, 松村雪桜, 山岸駿秀, 小町守. 機械翻訳の自動評価のための N-best を用いたマルチリファレンス作成手法の提案. NLP 若手の会第 13 回シンポジウム (YANS 2018). 高松. August 27, 2018.

10. 甫立健悟, 松村雪桜, 勝又智, 金子正弘, 小町守. 敵対的生成ネットワークを用いた文法誤り訂正. NLP 若手の会第 13 回シンポジウム (YANS 2018). 高松. August 28, 2018.
11. 平澤寅庄, 松村雪桜, 山岸駿秀, 小町守. 画像から抽出した複数種の特徴量を組み込んだニューラル機械翻訳の検討. NLP 若手の会第 13 回シンポジウム (YANS 2018). 高松. August 29, 2018.
12. 松村雪桜, 小町守. 敵対的生成ネットワークを用いた機械翻訳の評価における分野依存性の分析. NLP 若手の会第 13 回シンポジウム (YANS 2018). 高松. August 29, 2018.

#### 招待講演

1. 松村雪桜. ニューラル機械翻訳における敵対的生成ネットワークの導入. AAMT/Japio 特許翻訳研究会. 東京. October 12, 2018.
2. 山岸駿秀, 松村雪桜. 小町研究室における機械翻訳研究の動向. グローバルコミュニケーション開発推進協議会研究開発部会. 東京. December 14, 2018.

# 謝辞

研究活動に際して丁寧に指導をして下さるとともに，研究する上で快適な環境を与えて下さった小町守准教授に深く感謝します．研究生活を通して，学会発表やメンターとしての後輩の指導等，大変多くの貴重な経験をする事ができました．また，研究室配属時にメンターとして指導して下さいました佐藤さん，叶内さんには基礎から細かく教えて頂き，本当に感謝しています．そして，研究生活をともに過ごし様々な相談に乗って頂いた研究室の学生の皆さん，ありがとうございます．最後に，副査を引き受けて下さり多くのアドバイスを頂きました山口亨教授と高間康史教授に感謝します．

## 参考文献

- [1] I. Sutskever, O. Vinyals, and Q.V. Le, “Sequence to Sequence Learning with Neural Networks,” *Advances in Neural Information Processing Systems 27 (NIPS2014)*, eds. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, pp.3104–3112, Curran Associates, Inc., 2014.
- [2] K. Cho, B. vanMerriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation,” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.1724–1734, Association for Computational Linguistics, Doha, Qatar, 2014.
- [3] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” *Proceedings of the 3rd International Conference on Learning Representations (ICLR2015)*, 2015.
- [4] T. Luong, H. Pham, and C.D. Manning, “Effective Approaches to Attention-based Neural Machine Translation,” *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp.1412–1421, Association for Computational Linguistics, Lisbon, Portugal, 2015.
- [5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.u. Kaiser, and I. Polosukhin, “Attention Is All You Need,” *Advances in Neural Information Processing Systems 30*, eds. by I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, pp.5998–6008, Curran Associates, Inc., 2017. <http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf>
- [6] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, “Modeling Coverage for Neural Machine Translation,” *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp.76–85, Association for Computational Linguistics, Berlin, Germany, 2016. <http://www.aclweb.org/anthology/P16-1008>
- [7] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a Method for Automatic Evaluation of Machine Translation,” *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pp.311–318, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002.
- [8] D. Kauchak and R. Barzilay, “Paraphrasing for Automatic Evaluation,” *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, Main Conference*, pp.455–462, 2006. <http://aclweb.org/anthology/N06-1058>

- [9] 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 27*, eds. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, pp.2672–2680, Curran Associates, Inc., 2014. <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>
- [10] Z. Yang, W. Chen, F. Wang, and B. Xu, “Improving Neural Machine Translation with Conditional Sequence Generative Adversarial Nets ,” *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp.1346–1355, Association for Computational Linguistics, New Orleans, Louisiana, USA, 2018. <http://aclweb.org/anthology/N18-1122>
- [11] X. Mao, Q. Li, H. Xie, R.Y.K. Lau, and Z. Wang, “Least Squares Generative Adversarial Networks,” , 2016. <http://arxiv.org/abs/1611.04076>
- [12] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” , 2017. <http://arxiv.org/abs/1701.07875>
- [13] T. Nakazawa, M. Yaguchi, K. Uchimoto, M. Utiyama, E. Sumita, S. Kurohashi, and H. Isahara, “ASPEC: Asian Scientific Paper Excerpt Corpus,” *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, eds. by N.C.C. Chair), K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, pp.2204–2208, European Language Resources Association (ELRA), Paris, France, May 2016.
- [14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” *Proceedings of the 1st International Conference on Learning Representations (ICLR2013)*, 2013.
- [15] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” *Advances in Neural Information Processing Systems 26 (NIPS2013)*, eds. by C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, pp.3111–3119, Curran Associates, Inc., 2013.
- [16] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.1532–1543, Association for Computational Linguistics, Doha, Qatar, Oct. 2014.
- [17] L. Yu, W. Zhang, J. Wang, and Y. Yu, “SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient,” *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, 2017.

- [18] L. Wu, Y. Xia, L. Zhao, F. Tian, T. Qin, J. Lai, and T.-y. Liu, “Adversarial Neural Machine Translation,” Proceedings of the 10th Asian Conference on Machine Learning, 2017.
- [19] Y. Cheng, Z. Tu, F. Meng, J. Zhai, and Y. Liu, “Towards Robust Neural Machine Translation,” Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp.1756–1766, Association for Computational Linguistics, 2018. <http://aclweb.org/anthology/P18-1163>
- [20] Z. Yang, W. Chen, F. Wang, and B. Xu, “Unsupervised Neural Machine Translation with Weight Sharing,” Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp.46–55, Association for Computational Linguistics, 2018. <http://aclweb.org/anthology/P18-1005>
- [21] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada, “Automatic Evaluation of Translation Quality for Distant Language Pairs,” Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp.944–952, Association for Computational Linguistics, Cambridge, Massachusetts, USA, 2010.
- [22] R.E. Banchs and H. Li, “AM-FM: A Semantic Framework for Translation Quality Assessment,” Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp.153–158, Association for Computational Linguistics, Portland, Oregon, USA, 2011.
- [23] T. Nakazawa, H. Mino, I. Goto, G. Neubig, S. Kurohashi, and E. Sumita, “Overview of the 2nd Workshop on Asian Translation,” Proceedings of WAT, 2015.
- [24] T. Nakazawa, S. Higashiyama, C. Ding, H. Mino, I. Goto, G. Neubig, H. Kazawa, Y. Oda, J. Harashima, and S. Kurohashi, “Overview of the 4th Workshop on Asian Translation,” Proceedings of the 4th Workshop on Asian Translation (WAT2017), Taipei, Taiwan, Nov. 2017.
- [25] C.-Y. Lin and F.J. Och, “ORANGE: a Method for Evaluating Automatic Evaluation Metrics for Machine Translation ,” Proceedings of the 20th International Conference on Computational Linguistics, 2004.