2024年度AAMT/Japio特許翻訳研究会
報 告 書

機械翻訳及び機械翻訳評価に関する研究
並びに
第8回特許情報シンポジウム開催報告

2025 年 3 月

一般財団法人　日本特許情報機構

目 次

AAMT/Japio 特許翻訳研究会委員名簿

（敬称略・五十音順）

委 員 長	辻井 潤一	国立研究開発法人産業技術総合研究所 フェロー
		東京大学大学院 名誉教授
		マンチェスター大学 教授
副委員長	須藤 克仁	奈良女子大学 教授
	綱川 隆司	静岡大学学術院 情報学領域 准教授
委 員	荒瀬 由紀	東京科学大学 教授
	今村 賢治	国立研究開発法人 情報通信研究機構
		ユニバーサルコミュニケーション研究所
		先進的音声翻訳研究開発推進センター 先進的翻訳技術研究室
	越前谷 博	北海学園大学 教授
	岡崎 直観	東京科学大学 教授
	菊井玄一郎	国立研究開発法人科学技術振興機構(JST)
	黒橋 禎夫	国立情報学研究所 所長
	後藤 功雄	愛媛大学 教授
	小町 守	一橋大学大学院
		ソーシャル・データサイエンス研究科 教授
	鈴木 潤	東北大学 言語 AI 研究センター センター長／教授
	田村 晃裕	同志社大学 大学院理工学研究科 准教授
	中澤 敏明	東京大学大学院 特任研究員
	二宮 崇	愛媛大学 教授
	渡辺 太郎	奈良先端科学技術大学院 教授
オブザーバー	江原 暉将	元・山梨英和大学 教授
	高 京徹	株式会社高電社
	園尾 聡	東芝デジタルソリューションズ株式会社
	王 向莉	株式会社ディープランゲージ

オブザーバー（(一般）日本特許情報機構）：

大塩 只明	特許情報研究所 調査研究部 研究企画課
笠田 和宏	特許情報研究所 調査研究部 研究企画課 課長
木下 聡	特許情報研究所 調査研究部 研究企画課
小林 明	専務理事/特許情報研究所 所長
西出 隆二	特許情報研究所 調査研究部 部長
塙 金治	特許情報研究所 研究管理部 研究管理課
舩戸さやか	特許情報研究所 調査研究部 研究企画課 係長
三橋 朋晴	特許情報研究所 研究管理部 研究管理課 課長

事 務 局　　（一般）日本特許情報機構

2024 年度 AAMT/Japio 特許翻訳研究会・活動履歴

2024 年 4 月 22 日
第 1 回 AAMT/Japio 特許翻訳研究会
（於オンライン開催）

2024 年 6 月 17 日
第 2 回 AAMT/Japio 特許翻訳研究会
（於オンライン開催）

2024 年 8 月 2 日
第 3 回 AAMT/Japio 特許翻訳研究会
（於オンライン開催）

2024 年 10 月 10 日
第 4 回 AAMT/Japio 特許翻訳研究会
（於オンライン開催）

2024 年 11 月 7 日
第 8 回特許情報シンポジウム
（於オンライン開催）

2024 年 12 月 4 日
第 5 回 AAMT/Japio 特許翻訳研究会
（於ハイブリット開催）

2025 年 2 月 27 日
第 6 回 AAMT/Japio 特許翻訳研究会
（於オンライン開催）

１．はじめに

AAMT/Japio 特許翻訳研究会委員長
産業技術総合研究所　フェロー
辻井　潤一

ChatGPT に代表される生成 AI が翻訳という作業を大きく変えつつある。

私が英語での論文や手紙を書く場合には、まず自分なりに英語で書いてみて、それを生成 AI に校正してもらう。あるいは、思いつくことを日本語で書いて、それを生成 AI で英語のテキストにし、それに文の追加・削除・修正、文の順序入れ替えなどをして英語テキストを作り、それを再び生成 AI に校正してもらう。日本語テキストを作って翻訳システムを使うというより、日本語・英語の両方を使いながら、英語テキストを生成 AI と共同して作り上げる、という過程になっている。

ただ、こういうテキストの作り方をしているときに困るのは、生成 AI が作る英語テキストが自然で流暢なために、私の意図したメッセージとは違ったメッセージを伝える可能性を見逃してしまうことである。これを避けるためには、出来上がった英語テキストを慎重に読んでみて、意図通りになっているかを確認する必要がある。

以前に日本語の特許文書をもとに英語での特許申請を作る専門家の話を聞いたことがある。この専門家は、日本語テキストから英語テキストを作る過程で、正しい英語を作るために技術者と頻繁にやり取りをする、と言っていた。簡単な例では、「A の上の B」を英語にするためには、A と B が接触しているかいないかを知り、”B on A”か“B above A”のいずれかを選択しなければならない。日本語文だけを見ていると接触しているかいないかの情報はない。正しく英語を作るための情報が欠けているのである。本来”B above A”と訳すべきところを翻訳システムが”B on A”と誤訳しても、記述されている内容がないとこの誤訳には気が付かない。結局、A と B がどういう位置関係にあるかを日本語特許テキストを書いた技術者に訊く必要がある。

機械翻訳の出力した訳文を修正するのは思ったよりもむつかしい。上の例では、翻訳の専門家だけでなく、英語に習熟していない技術者も英語を校正することはできない。技術者は、日本語の「上」が実際の位置関係で”on”と“above”に訳仕分けすることができないから。結局は、（１）両者が協働作業をする、（２）専門分野の知識をもった、その分野を専門とする翻訳家が行う、（３）英語に習熟した技術者、が行うことになる。

ちなみに、よく使われている機械翻訳システムに、「天井からいくつかの電灯がつるされています。テーブルの上の電灯だけを切ってください」を翻訳させると、「上」を”on”と翻訳してしまいます。あきらに、翻訳システムは、テキストで描かれている状況を理解していない。これに対して、同じテキストを私が使っている生成 AI に翻訳させると”above”と翻訳する。生成 AI は記述される状況を「理解」し、それを”on”と“above”の使いわけに反映している。

実際には、人間であれば言語文脈から正しい使いわけできるケースでも、生成 AI が正しく判断できない場合も多い。現在の生成 AI が人間と同等な理解能力を持っているわけではない。ただ、生成 AI が、理解を翻訳の過程に反映する能力を持ちだしていることは確かで、従来の機械翻訳の限界を突破し始めていることは明らかであろう。生成 AI は、従来の機械翻訳の直訳の世界から、理解を経て翻訳を行う意訳の世界へと進み始めている。今研究が活発化しているマルチ・モーダル生成ＡＩは、言語文脈には出てこない状況をも理解し、それを翻訳に反映できるようになるのだろうか？

優秀な自然言語研究者が集積しているこの委員会が解くべき挑戦的な課題は、山積している。今後の委員会の活躍を期待している。

2. 年間報告書

2.1 低資源言語を対象とした訳文情報を含む単語分散表現を用いたニューラル機械翻訳

北海学園大学　越前谷　博
北海学園大学　田中　蒼大郎

2.1.1 はじめに

近年、ニューラル機械翻訳の進展により機械翻訳システムの性能向上が著しい。しかし、翻訳精度の向上には大規模な対訳コーパスが不可欠となるため、言語資源が乏しい低資源言語においては必ずしも十分な精度が得られるとは限らない。そこで低資源言語を対象としたニューラル翻訳の研究が盛んに行われている。

低資源言語における機械翻訳の問題を解決するためのアプローチとして、データ拡張の観点より擬似対訳コーパスを自動生成する手法が提案されている。例えば、Zhang ら[1]は原言語の単言語コーパスからニューラル機械翻訳を用いて目的言語文を得ることで擬似対訳コーパスを自動生成した。一方、Sennrich ら[2]は目的言語の単言語コーパスからニューラル機械翻訳を用いて原言語文を得ることで擬似対訳コーパスを自動生成し、それらを加えた対訳コーパスをニューラル機械翻訳の学習データとした。さらに Oh ら[3]は ChatGPT を活用することでプロンプトベースのデータ拡張を行う手法を提案した。その際には、原言語と目的言語の文それぞれに対して言い換えを行う方法（Paraphrase)、1 つの原言語の文を複数の目的言語の文に翻訳する方法（Mult-Target)、そして、原言語の文から 3 つの文を生成し、それらを翻訳する方法（Storytelling）により対訳コーパスを得た。しかし、このようなデータ拡張により生成される擬似対訳コーパスの中には質の低い文が含まれることがあるため、ニューラル機械翻訳の学習の妨げになることが問題となる。

また、データ拡張のアプローチとは異なるアプローチとして、ニューラル機械翻訳のモデル自体を低資源言語の翻訳に適応させる手法も提案されている。Zoph ら[4]は大規模な対訳コーパスを用いて親モデルを生成し、さらに低資源言語による小規模な対訳コーパスを用いて子モデルを生成した。そして、親モデルのパラメータを用いて子モデルを初期化することで、親モデルを用いないニューラル機械翻訳に比べて翻訳精度が向上することを確認した。しかし、この手法では大規模な対訳コーパスを用いて事前に親モデルを構築する必要がある。また、子モデルの構築の際に用いる対訳コーパスの言語は親モデルの構築時に用いた対訳コーパスの言語と類似した言語であることが前提となる。したがって、翻訳対象となる低資源言語は親モデルの制約を受けることが問題となる。

本報告では、上述した 2 つのアプローチにおいて後者の低資源言語の翻訳が可能なニューラル機械翻訳のモデルを構築する手法として新たな手法を提案する。提案手法では、BERT-fused NMT[5]のエンコーダ側の単語埋め込みに使用する単語分散表現を目的言語の情報を有するベクトルとすることで翻訳精度の向上を図る。また、そのような単語分散表現は多層パーセプトロンによるニューラルネットワークモデルを構築することで生成する。このようなニューラルネットワークモデルを本報告では“ベクトル変換モデル”と記す。また、このベクトル変換モデルを用

いた BERT-fused NMT を本報告では“Vec-trans NMT”と記す。本研究でもベクトル変換モデルを事前に構築する必要があるが、Vec-trans NMT の学習時に用いる対訳コーパスと同様の小規模な対訳コーパスを学習データとして使用するため、低資源言語に対応可能な汎用性の高いニューラル機械翻訳を実現できると考えられる。

2.1.2 提案手法

2.1.2.1 概要

図 1 に提案手法である Vec-trans NMT の概要図を示す。Vec-trans NMT における翻訳処理のアーキテクチャには BERT-fused NMT を用いる。BERT-fused NMT は BERT が出力する単語分散表現を Transformer のエンコーダとデコーダそれぞれに組み込んだ BERT-Enc Attention 機構と BERT-Dec Attention 機構で利用することで、より高い精度で文脈処理を行うことができる。また、提案手法では BERT ではなく多言語に対応した mBERT を使用している。

さらにエンコーダの単語埋め込みにおいて原言語文に対応する目的言語文の情報を含んだ文ベクトルをベクトル変換モデルにより生成し、それを単語分散表現に変換したベクトルを使用する。ここでベクトル変換モデルはフルスクラッチ実装の多層パーセプトロンによるニューラルネットワークモデルとなっている。

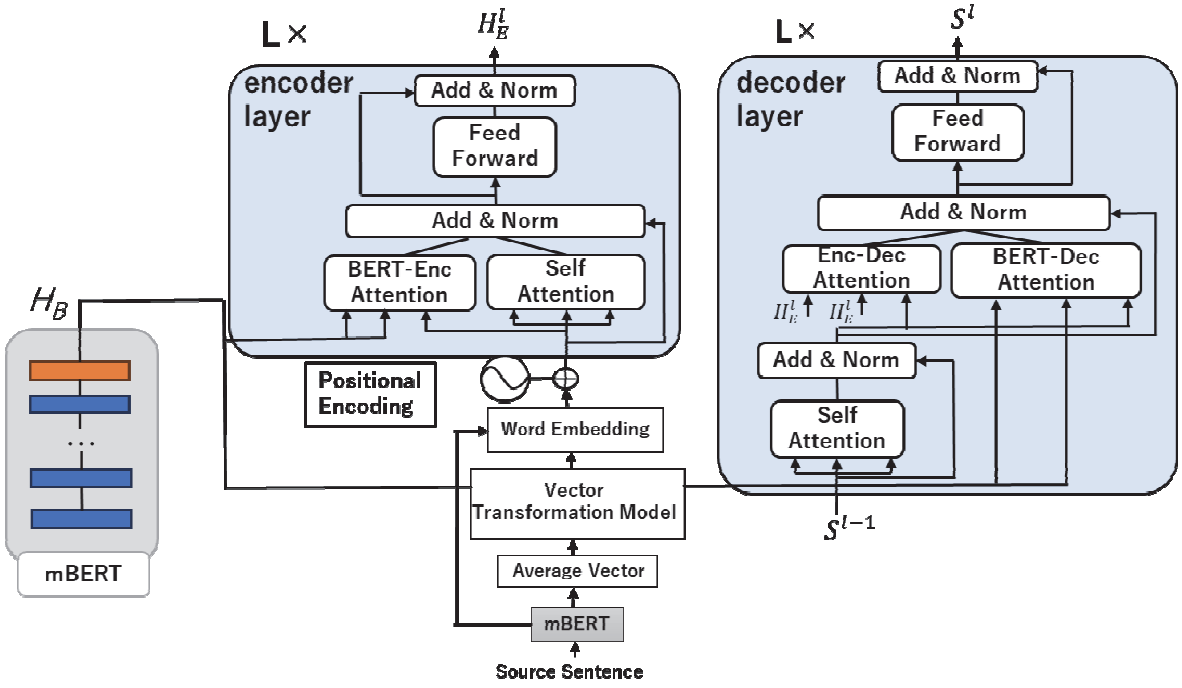

図 1　Vec-trans NMT の概要図

2.1.2.2 ベクトル変換モデルの概要

ベクトル変換モデルは原言語の文ベクトルに対して目的言語の文ベクトルの情報を反映したベクトルに変換することが目的である。図２にベクトル変換モデルの概要図を示す。

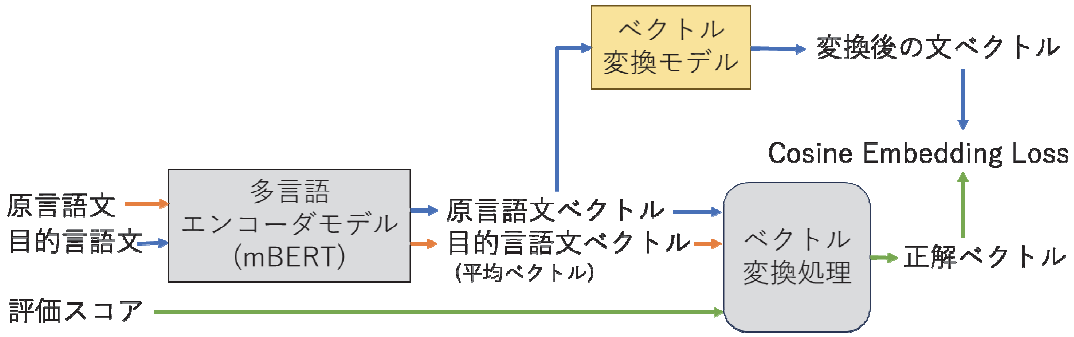

図２　ベクトル変換モデルの概要図

図 2 よりベクトル変換モデルの入力は原言語文に対して多言語のエンコーダモデルである mBERT を用いて得られる単語分散表現の平均ベクトルである。この原言語の文ベクトルはベクトル変換モデルにより原言語文に対応する目的言語文の情報を反映した文ベクトルに変換される。ベクトル変換モデルの理想的な出力は、目的言語の文ベクトルとの間のコサイン類似度が適切な評価スコアとなるような文ベクトルである。ここで直接 mBERT より得られる原言語文と目的言語文の出力ベクトルの平均ベクトル間のコサイン類似度を求めても、評価スコアとして適切な値は得られない。これは mBERT が文間の類似度を求めることを目的としたエンコーダモデルではないためである。また、文間の意味的な類似度を得るためのエンコーダモデルとして mSBERT[6] があるが、低資源言語においては、mSBERT の対象外の言語が存在した場合、その利用が困難となる。

したがって、本研究ではフルスクラッチ実装によるベクトル変換モデルを構築する。ここでベクトル変換モデルの学習時の正解には原言語の文ベクトルを目的言語の文ベクトルとの間のコサイン類似度が評価スコアと一致ように変換された文ベクトルを用いる。以降、これを正解ベクトルと記す。この正解ベクトルは mBERT より得られる原言語と目的言語の文ベクトル、そして、評価スコアに基づきベクトル変換処理にて生成される。ベクトル変換モデルより変換された文ベクトルと正解ベクトルに対しては、Cosine Embedding Loss により誤差を求めることで学習を行う。

2.1.2.3 ベクトル変換処理における評価スコアの生成

ベクトル変換処理に用いる評価スコアには人手による評価スコアを使用することが理想ではあるが、低資源言語においてそれは困難なタスクであると考えられる。そこで、本研究においては多言語のエンコーダモデルである mSBERT より得られる原言語と目的言語の文ベクトル間のコサイン類似度を評価スコアとして用いることとする。ただし、前述したとおり低資源言語には mSBERT に含まれない言語も存在することが想定されるため、その場合には、mSBERT に含ま

れている言語から言語的に近しい言語をいくつか選択し、それらの言語文を Google 翻訳により生成したうえで mSBERT の入力に用いる。mSBERT に含まれない言語を対象とした場合の評価スコア算出の概要を以下の図３に示す。

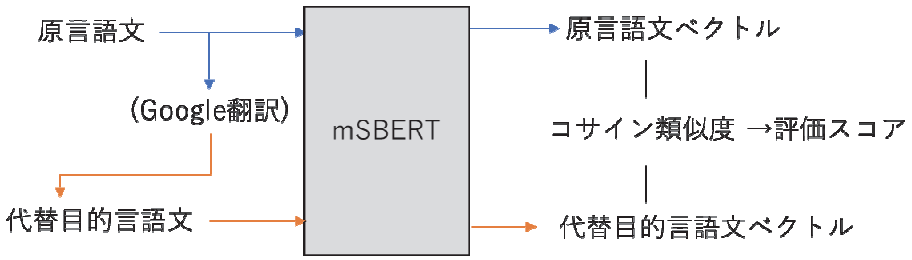

図３　低資源言語を対象とした mSBERT による評価スコアの生成

図３は目的言語が mSBERT には含まれていない言語とした場合の評価スコア生成の概要図である。その場合、原言語文を Google 翻訳を用いて目的言語と言語的に近しい言語の文を得る。これは対象となる目的言語文ではなく、あくまでも代替となる目的言語文となる。そして、原言語文と代替目的言語文を mSBERT により文ベクトルに変換し、コサイン類似度を求めることで評価スコアを得る。ここで代替目的言語文となる言語は、一つの言語だけでなく複数の言語とし、一つの原言語文に対して複数の代替目的言語文を生成する。その結果、異なる言語の複数のコサイン類似度が得られることになるため、それらのコサイン類似度の平均を評価スコアとして用いる。このような複数の言語による代替目的言語文を利用することでより客観的な評価スコアが得られると考えられる。

2.1.2.4 ベクトル変換処理における正解ベクトルの生成

図２に示したベクトル変換処理における正解ベクトルの生成について詳細を述べる。ベクトル変換処理への入力は mBERT より得られる原言語文と目的言語文の単語ベクトルをそれぞれ平均ベクトルにした文ベクトルと 2.1.2.3 で述べた評価スコアである。正解ベクトルの生成のためのアルゴリズムを以下に示す。 ここで、n 次元の原言語の文ベクトル を $S=(s_1,s_2,\ldots,s_n)$、n 次元の目的言語の文ベクトルを $T=(t_1,t_2,\ldots,t_n)$とする。

1. 原言語の文ベクトル S と目的言語の文ベクトル T の全要素について要素毎の比とそれらの平均値を m として求める。そして、要素毎の比から m を引いた、絶対値の並びを原文ベクトルの変換要素の箇所の決定の際に用いる。要素毎の比から平均 m を引いたものの絶対値の集合を R とすると以下のようになる。

$$R=\left\{\left|\frac{s_1}{t_1}-m\right|,\ \left|\frac{s_2}{t_2}-m\right|,\ \ldots,\left|\frac{s_n}{t_n}-m\right|\right\}$$

2. 原言語の文ベクトル S と目的言語の文ベクトル T によるコサイン類似度と評価スコアを比較する。

(1)コサイン類似度の方が大きい場合はRの要素の中で値が最小である要素位置iを取得する。

(2) 評価スコアの方が大きい場合はRの要素の中で値が最大である要素位置iを取得する。

3. 処理2により変換箇所として決定された原言語の文ベクトルのi番目の要素に変数xを付与した原言語の文ベクトルをS’とする。

$$S' = (s_1, \ldots, s_i + x, \ldots, s_n)$$

4. ベクトルS’とベクトルTを以下の式(1)のコサイン類似度の計算式に当てはめる。その際、コサイン類似度の値として、2.1.2.3で述べた評価スコアを用いる。

$$\frac{\sum_{j=1}^{i-1}(s_j \times t_j) + (s_i + x) \times t_i + \sum_{j=i+1}^{n}(s_j \times t_j)}{\sqrt{\sum_{j-1}^{i-1} s_j^2 + (s_i + x)^2 + \sum_{j-i+1}^{n} s_j^2} \times \sqrt{\sum_{j-1}^{n} t_j^2}} = \text{評価スコア} \qquad \text{式(1)}$$

5. 式(1)は変数xを含んでおり、変数xの二次方程式となる。したがって、この二次方程式の解を求めることでベクトルS’を決定することができ、正解ベクトルが得られる。

(1) 解が2つの実数解の場合、絶対値が小さい方の解をxの値とする。また、解が1つの場合はその値をxの値とする。そして、xの値を付与したベクトルS’を正解ベクトルとする。

(2) 解が2つの複素数解の場合、実数部分のみをxの値とし、ベクトルS’とベクトルTの間のコサイン類似度を求める。その結果、評価スコアとの差が0.005以下となった時点のベクトルS’を正解ベクトルとする。0.005を超える場合には処理1以降を繰り返す。その場合、xの値を付与したベクトルS’を新たなベクトルSとする。

ここで、処理2においてコサイン類似度が評価スコアよりも大きい場合にはコサイン類似度を下げる必要がある。そのためには要素間の比が小さい箇所に着目し、その差を拡大するためにRの中で最小値が存在する位置を決定し、その位置に対応する原言語の文ベクトルの要素に変数xを加える。一方、評価スコアの方が大きい場合には逆に差を縮小させるために、Rの中で値が最大の位置を決定し、その位置に対応する原言語の文ベクトルの要素に変数xを加える。

また、複素数解が得られた場合には実部のみを用いて要素の更新を行っている。ここで虚数部を使用するために絶対値を用いるなど様々な方法が考えられるが、予備実験の結果より実部のみを用いた方法が最も評価スコアに近づけることができたため、今回は上述した方法を使用している。しかし、この点については今後も検討を行う予定である。このように複素数解が得られた場合、コサイン類似度を評価スコアと一致させることは困難となるため正解ベクトルには近似値を用いたベクトルを正解ベクトルとして用いることとなる。

2.1.2.5 ベクトル変換モデルによる文ベクトルの単語分散表現への変換

学習後のベクトル変換モデルを用いて得られる文ベクトルは単語分散表現に変換され、図1のニューラル機械翻訳のエンコーダの単語埋め込みに使用される。ここでは文ベクトルを単語分散表現にどのように変換しているのかについて述べる。本研究ではmBERTより得られる原言語の単語分散表現に基づき全単語ベクトルの要素毎の平均がベクトル変換モデルにより変換された文

ベクトルと一致するように任意の固定値を原言語の全単語ベクトルの同じ位置の要素に加える。例えば変換された文ベクトルを $T' = (t'_1, t'_2, \dots, t'_n)$ 、原言語文の構成単語数を m、mBERT によるそれぞれの単語ベクトルを $W_1 = (w_{11}, w_{12}, \dots, w_{1n})$, … , $W_m = (w_{m1}, w_{m2}, \dots, w_{mn})$ 、単語ベクトルの第 1 要素から第 n 要素に加える固定値を x_1, …, x_n とした場合、

$\frac{(w_{11} + x_1) + (w_{21} + x_1) + \dots + (w_{m1} + x_1)}{m} = t'_1$, … , $\frac{(w_{1n} + x_n) + (w_{2n} + x_n) + \dots + (w_{mn} + x_n)}{m} = t'_n$ が成り立つ x_1, …, x_n の値を求める。次いで、W_1, ..., W_m の各要素にそれぞれ x_1, …, x_n の値を加えることで変換された文ベクトルに対応する単語ベクトルを生成する。そして、生成された単語ベクトルをニューラル機械翻訳の単語埋め込みに用いる。

2.1.3 性能評価実験

2.1.3.1 実験データ

性能評価実験に使用した実験データについて述べる。提案手法におけるベクトル変換モデルの学習時に必要な評価スコアの算出にはアイヌ語日本語間ではニューラル機械翻訳の学習時に使用するアイヌ語日本語対訳文 91,168 対[7]~[9]に含まれている訓練データ 72,936 を用いた。その際、アイヌ語は Google 翻訳で扱うことが困難であるため、アイヌ語の代替目的言語としてモンゴル語、トルコ語、そして、韓国語を用いて日本語との間で翻訳を行い、mSBERT により原言語の文ベクトルと目的言語の文ベクトルを得た。アイヌ語は言語学的には独立言語であるが、日本語との共通点もあるため、日本語と文法構造が比較的類似している言語を用いた。そして、日本文との間のコサイン類似度の平均値を求め、それらが閾値 0.95 より高い場合、その日本文とアイヌ文の対をベクトル変換モデル学習時の評価スコアとして用いた。また、ベトナム語日本語間においてもニューラル機械翻訳の学習時の対訳文 20,101 対[10]に含まれている訓練データ 16,081 を評価スコアの算出に適用した。その際、ベトナム語は mSBERT に含まれているため Google 翻訳は用いていない。また、コサイン類似度に対する閾値には 0.9 を用いた。この閾値より高いコサイン類似を示したベトナム文と日本文の対をベクトル変換モデルの学習時の評価スコアとした。その結果、ベクトル変換モデルはアイヌ語日本語間の学習データ 4,787 文、検証データ 1,000 文の合計 5,787 の対訳文とそれらに対する評価スコアを用いて生成した。また、ベトナム語日本語間では学習データ 5,543 文、検証データ 1,000 文の合計 6,543 の対訳文とそれらに対する評価スコアを用いてベクトル変換モデルを生成した。

また、ニューラル機械翻訳のモデルにおいてはアイヌ語日本語間では 91,168 対、ベトナム語日本語では 20,101 対の対訳コーパスに対して 8:1:1 の割合で訓練データ、検証データ、そして、評価データ用に分割してモデルを構築した。

2.1.3.2 実験方法

本報告ではアイヌ語日本語間の翻訳実験とベトナム語日本語間の翻訳実験を通して、提案手法の有効性を検証する。そこでまずベクトル変換モデルの生成を行う。具体的にはアイヌ語から日本語、日本語からアイヌ語、ベトナム語から日本語、そして、日本語からベトナム語に対応した

ベクトル変換モデルを生成し、その後、それらを用いた Vec-trans NMT の学習を行った。したがって、ベクトル変換モデル、Vec-trans NMT のモデルそれぞれ 4 つずつを生成したうえで翻訳実験を行った。また、提案手法の有効性を確認するために、mBERT を用いた BERT-fused NMT のモデルをそれぞれの翻訳方向に伴い、4 つ生成した。そして、それらをベースラインとした。

本研究で生成するベクトル変換モデルのアーキテクチャには入力層（ニューロン数:768）， 2 つの中間層（ニューロン数:1536 と 768）， 出力層（ニューロン数:768）からなる多層パーセプトロンを用いた。入力層と出力層のニューロン数をそれぞれ 768 としているのは、mBERT より得られるベクトルが 768 次元となっているためである。

提案手法である Vec-trans NMT とベースラインである mBERT を用いた BERT-fused NMT に対する評価は SacreBLEU を用いた。

2.1.3.3 実験結果

アイヌ語日本語間の比較実験の結果を表 1 に、ベトナム語日本語間の比較実験の結果を表 2 に示す。

表 1　アイヌ語日本語間における BLEU スコア

	アイヌ語から日本語	日本語からアイヌ語	平均
提案手法	24.52	33.36	28.94
ベースライン	24.89	32.20	28.55

表 2　ベトナム語日本語間における BLEU スコア

	ベトナム語から日本語	日本語からベトナム語	平均
提案手法	9.42	7.79	8.61
ベースライン	8.69	8.60	8.65

2.1.3.4 考察

表 1 より提案手法はアイヌ語から日本語の翻訳においては BLEU スコアがわずかにベースラインに比べて低い値となったが、日本語からアイヌ語においては BLEU スコアが 1.16 ポイント向上した。その結果、平均では提案手法がベースラインを上回った。また、表 2 よりベトナム語から日本語においては BLEU スコアが 0.73 ポイント向上したが、日本語からベトナム語においては 0.81 ポイント低下した。その結果、平均では提案手法がベースラインをわずかに下回った。これらの結果より、提案手法はアイヌ語日本語間においては有効に働き、ベトナム語日本語間においては不十分であったと考えられる。

そこで、提案手法の特徴であるベクトル変換モデルがどの程度、目的言語文を反映した文ベクトルを生成できていたのかを調査した。調査の詳細を図 4 のベクトル変換モデルに対する調査方法の概要に基づき述べる。図 4 よりベクトル変換モデルにより変換された文ベクトルと目的言語の文ベクトル間のコサイン類似度、さらにベクトル変換モデルによる変換前の原言語の文ベクト

ルと目的言語の文ベクトル間のコサイン類似度を求める。この 2 つのコサイン類似度を比較した場合、ベクトル変換モデルにより変換された文ベクトルは目的言語の情報を反映したものとなっているため目的言語の文ベクトルとのコサイン類似度は高い値を示すことが予想される。そこで、ニューラル機械翻訳モデルの評価に使用した文から任意で 100 文を抽出し、ベクトル変換モデルによる文ベクトルと目的言語の文ベクトルとの間のコサイン類似度、そして、原言語と目的言語の文ベクトルとの間のコサイン類似度を求めた。その結果を表 3 に示す。

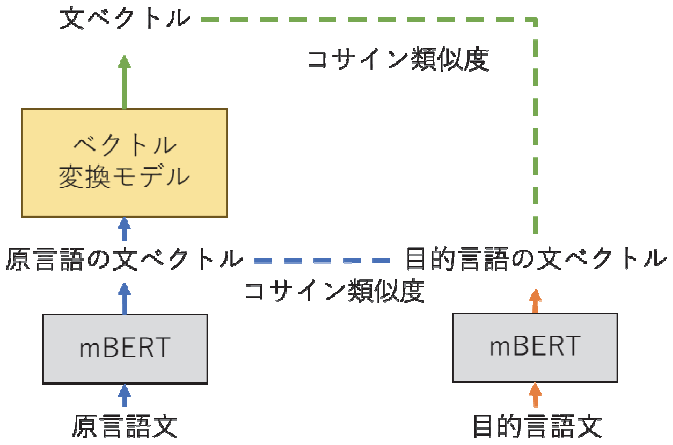

図 4　ベクトル変換モデルに対する調査方法の概要

表 3　ベクトル変換モデルの有無によるコサイン類似度と評価スコアの関係

原言語文	目的言語文	ベクトル変換モデル	コサイン類似度の平均値	評価スコア
アイヌ語	日本語	なし	0.3567	0.8387
		あり	0.7660	
日本語	アイヌ語	なし	0.3567	
		あり	0.7811	
ベトナム語	日本語	なし	0.6244	0.8743
		あり	0.8563	
日本語	ベトナム語	なし	0.6244	
		あり	0.8232	

表 3 の評価スコアは抽出された 100 文に対して **2.1.2.3** で述べた方法により求めた評価スコアの平均である。コサイン類似度の平均値を見ると、ベクトル変換モデルにより得られた文ベクトルを用いたコサイン類似度は原言語の文ベクトルを用いたコサイン類似度よりも高く、かつ、評価スコアにも近い値となっていることが確認できる。これはベクトル変換モデルが目的言語の情報を含んだ文ベクトルの生成に有効であったことを示している。

したがって、実験結果におけるベトナム語日本語間の翻訳精度が不十分であった原因としては、

ニューラル機械翻訳に対する学習データの不足により、ベクトル変換モデルの効果が翻訳精度に十分に反映されなかったためと考えられる。一方、アイヌ語日本語間の翻訳においては、学習データはベトナム語日本語の学習データに比べて約 4.5 倍であり、ベクトル変換モデルを有効利用できたため翻訳精度の向上をもたらしたと考えられる。

2.1.4 まとめ

本報告では、低資源言語を対象とした新たなニューラル機械翻訳の手法として Vec-trans NMT を提案した。提案手法では、様々な言語に対応可能なニューラル機械翻訳の構築を目的にフルスクラッチ実装の多層パーセプトロンによるニューラルネットワークモデルとしてベクトル変換モデルを生成し、それを使用した。ベクトル変換モデルは原言語の文ベクトルを目的言語の情報を反映した文ベクトルに変換する。そして、本研究では、変換された文ベクトルをニューラル機械翻訳の単語埋め込みに適用した。性能評価実験の結果、ベースラインに対してベトナム語日本語間の翻訳精度においてはほぼ同等であったが、アイヌ語日本語間においては翻訳精度の向上が確認された。

今後はベトナム語日本語間のように学習データが不十分な場合であっても翻訳精度に反映できるように、ベクトル変換モデルのニューラル機械翻訳への適用方法を検討する予定である。

謝辞

本研究の一部は一般社団法人アイヌ文化学術研究会との共同研究により行なわれた。

参考文献

[1] Jiajun Zhang and Chengqing Zong (2016) “Exploiting Source-side Monolingual Data in Neural Machine Translation,” Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP16), pp. 1535–1545.

[2] Rico Sennrich and Barry Haddow and Alexandra Birch (2016) “Improving Neural Machine Translation Models with Monolingual Data,” Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 16), pp. 86–96.

[3] Seokjin Oh, Su Ah Lee, and Woohwan Jung (2023) “Data Augmentation for Neural Machine Translation using Generative Language Model,” https://arxiv.org/pdf/2307.16833

[4] Barret Zoph, Deniz Yuret, Jonathan May and Kevin Knight (2016) “Transfer Learning for Low-Resource Neural Machine Translation,” Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP16), pp. 1568–1575.

[5] Jinhua Zhu , Yingce Xia , Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li and Tie-Yan Liu (2020) “Incorporating BERT into Neural Machine Translation,” https://arxiv.org/pdf/2002.06823

[6] Nils Reimers and Iryna Gurevych (2020) “Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation,” Proceedings of the 2020 Conference on Empirical

Methods in Natural Language Processing, pp. 4512–4525.
[7] 田中蒼大郎, 越前谷博, 荒木健治 (2022) “Transformer を用いたアイヌ語-日本語間機械翻訳における精度と対訳コーパスの関係,” 令和 4 年度電気・情報関係学会北海道支部連合大会講演論文集, pp. 227-228.
[8] 田中蒼大郎, 越前谷博, 荒木健治 (2024) “翻訳対象言語の情報を利用した低資源言語のためのニューラル機械翻訳,” 令和 6 年度電気・情報関係学会北海道支部連合大会講演論文集, pp. 36-37.
[9] 田中蒼大郎, 越前谷博, 荒木健治 (2024) “ベクトル間類似度に基づくベクトル変換モデルを用いた低資源言語のためのニューラル機械翻訳, ” 第 262 回自然言語処理研究発表会, 講演番号 20.
[10] Hammam Riza, Michael Purwoadi, Gunarso, Teduh Uliniansyah, Aw Ai Ti, Sharifah Mahani Aljunied, Luong Chi Mai, Vu Tat Thang, Nguyen Phuong Thái, Vichet Chea, Rapid Sun, Sethserey Sam, Sopheap Seng, Khin Mar Soe, Khin Thandar Nwet and Masao Utiyama, Chenchen Ding (2016) “Introduction of the Asian Language Treebank,” Proceedings of 2016 Conference of the Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Technique (O-COCOSDA), pp.1-6.

2.2 多言語大規模言語モデルにおける英語指示文と対象言語指示文の公平な比較

東京都立大学大学院　システムデザイン研究科　情報科学域
榎本　大晟・金　輝燦
一橋大学大学院　ソーシャル・データサイエンス研究科
小町　守

2.2.1 はじめに

近年、大規模言語モデル（LLM）はさまざまな自然言語処理タスクにおいて優れた性能を示している。その能力を最大限に引き出すためには、 LLM に適切な指示を与えることが必要不可欠である [1, 2]。特に、**多言語大規模言語モデル（MLLM）**を用いて英語以外の言語（対象言語）のタスクを解く際、そのモデルへの指示文を英語で与えるべきか、それとも対象言語で与えるべきかについてはいくつかの研究で議論されてきた [3, 4, 5]。この背景には、 MLLM の学習データは多くの場合、英語を中心に構築されているという事実がある。このことから、たとえタスクが英語以外の言語であっても、英語で指示を与える方が MLLM の能力をより効果的に引き出せる可能性が指摘されている。実際に多くの先行研究が、対象言語よりも英語で指示文を与える方が高性能になる傾向を報告している [4, 5]。

しかしながら、これらの先行研究では、対象言語のテストデータセットや指示文として英語から翻訳されたものが使用されている、という問題がある。翻訳によって作成された文は、情報の欠落や不自然さ、母語話者が書いた文とは著しく異なる文体や構造を持つ可能性（**Translationese**）がある [6, 7, 8]。これにより、英語から翻訳された対象言語のデータセットでは、表現が英語の文体に近づいたり、内容が英語圏の文化や背景に影響を受けている場合がある。また、対象言語の指示文において翻訳の前後で含んでいる情報が異なる可能性がある。これらの要因から、先行研究では英語指示文が潜在的に有利な設定になっており、英語指示文と対象言語指示文の公平な比較ができていないと考えられる。

この問題を解決するために、本研究では、Translationese の影響を排除して、 MLLM において英語指示文と対象言語指示文の公平な比較を行う。具体的には、**翻訳に基づかない対象言語のデータセットや、言語的に自然で同じ内容を伝える公平な指示文（図 1）を使用し、 MLLM の性能の違いをさまざまなタスクで調査する**。特に分類タスクでは、複数のラベルセットを使用し、ラベルセットの違いによる結果の変化についても調べる。実験結果から、先行研究とは異なり、英語指示文と対象言語指示文のどちらがより優れているかは、タスクや分類ラベルによって異なる傾向があることを明らかにする。さらに、それぞれの指示文を用いた場合の MLLM が生成するテキストの特徴や指示追従度、活性化するニューロンの違いについての詳細な分析も行う。本研究は、MLLM における指示文言語の公平な比較を行うことで、 MLLM の能力を効果的に引き出すための新たな知見を提供する。

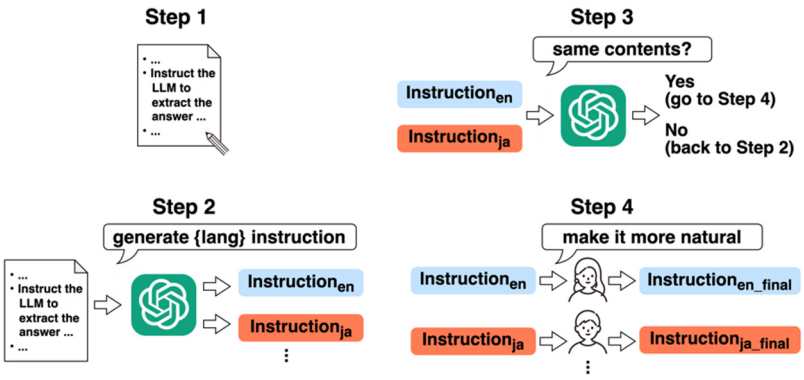

図 1: 公平な指示文の作成手順。

2.2.2 関連研究

指示チューニング済みモデルへのプロンプトはインスタンスと指示文の両方から構成されることが一般的である。MLLM へ入力するプロンプトが英語であるべきかそれとも対象言語であるべきかについての研究はインスタンスベースと指示文ベースに分類できる。

インスタンスベース インスタンスベースのアプローチではインスタンスを英語に翻訳することに焦点を当てている。Huang ら [9] と Etxaniz ら [10] は、タスクを処理する LLM 自体を用いてインスタンスを英語に翻訳することの有効性を報告した。一方で、Intrator ら [11] は PaLM2 ではインスタンスを英語に翻訳することがタスクの性能を低下させる傾向があることを報告した。

指示文ベース 本研究が属する指示文ベースのアプローチは、指示文やプロンプトテンプレートの言語に焦点を当てており、インスタンスには変更を加えない。Lin ら [3] は MLLM に与えるプロンプトテンプレートの言語を比較し、英語のテンプレートがより高い性能になる傾向を報告した。Muennighoff ら [4] と Ahuja ら [5] は指示チューニング済みモデルにおいて、英語指示文と英語から翻訳された対象言語の指示文を比較し、英語指示文がより高い性能になる傾向を報告した。しかしながら、これらの研究は XNLI [12] のような英語から翻訳された多言語のデータセットをテストデータとして使用したり、英語から翻訳された対象言語の指示文を使用しており、Translationese の影響を考えていない。一方で、Barei ら [13] は英語に基づかない多言語データセットを用いているが、プロンプトテンプレートは機械翻訳に基づいている点と、自己回帰言語モデルでなくマスク言語モデルに焦点を当てている点において本研究と異なる。

2.2.3 公平な比較の実現

この節では、英語指示文と対象言語指示文の公平な比較を実現するために Translationese の影響を排除する方法と、実験設定について説明する。

2.2.3.1 公平な指示文の作成

英語指示文と対象言語指示文の公平な比較をするためには、両方の指示文が十分に流暢であることと、同じ内容を伝えることが不可欠である。我々はそのような指示文を以下の手順で作成す

る（図 1）：

1. 各タスクの指示文に含まれるべき内容を人手で定義する。
2. Step 1 の定義に基づき、 GPT-4（gpt-4o-2024-05-13）を用いて各言語の指示文を生成する。
3. 英語指示文と対象言語指示文が同じ内容を伝えているかを GPT-4 を用いて検証する。内容に違いがあると判断された場合は、Step~2 に戻る。
4. 各言語の指示文が自然な表現や言い回しになるように母語話者が修正を行う。

Step 1 の定義文から各言語の母語話者が指示文を作成する方法も検討したが、この方法では指示文間で内容や形式の差異が確認された。一方で我々の作成手順では、言語間で指示文が同じ内容を伝え、言語的に自然であることを保証する。

2.2.3.2 タスクとデータセット

本研究では 3 つのタスクで英語指示文と対象言語指示文の比較を行う。以降、それぞれのタスクの概要と、実験に使用する英語からの翻訳に基づかないテストデータセットについて説明する。表 1 に各タスクの日本語においてのインスタンスの例を示す。

語彙平易化タスク　語彙平易化タスクは、ある文中の対象となる単語を、より簡単で理解しやすい同義語に置き換えることで、文の意味を保ちながら平易にするタスクである。本研究では、対象単語ごとにより平易な同義語を 1 つ生成し、それがゴールドスタンダードの解答に含まれるかに基づいて Accuracy を測定する。語彙平易化タスクにおいての対象言語は、de、es、fr、ja、zh の 5 言語である。テストデータセットとしては、 MultiLS [14]（de、es、fr、ja）、Chinese-LS [15]（zh）を用いる。

機械読解タスク　機械読解タスクは、質問と参照テキストが与えられ、その質問に対する答えを参照テキストから抽出するタスクである。本研究では、質問に対する答えを参照テキストから抽出するように指示し、生成されたテキストがゴールドスタンダードの答えと完全に一致するかに基づいて Accuracy を測定する。機械読解タスクにおいての対象言語は、de、es、fr、id、ja、ko、zh の 7 言語である。テストデータセットとしては、GermanQuAD [16]（de）、SQAC [17]（es）、FQuAD [18]（fr）、TyDiQA-Gold [8]（id、ja、ko）、DRCD [19]（zh）を用いる。

レビュー分類タスク　本研究におけるレビュー分類タスクは、レビュー文の評価が肯定的か否定的かを分類するタスクである。分類ラベルセットの違いによる結果の変化を分析するために、英語のラベルセットを用いる設定と対象言語のラベルセットを用いる設定のそれぞれで英語指示文と対象言語指示文の macro-F1 を比較する。表 2 に各対象言語のラベルセットを示す。レビュー分類での対象言語は、de、es、fr、id、ja、ko、zh の 7 言語である。テストデータセットとしては、MARC [20]（de、es、fr、ja、zh）、NSMC [21]（ko）、PRDECT-ID [22]（id）を用いる。

表 1: 各タスクの日本語におけるインスタンスの例。

タスク	インスタンス	解答
語彙平易化	Sentence: 大病院は救急や重い症状の患者さんの治療を担う役割を持っています。 Target word: 担う	する，行う，引き受ける，...
機械読解	Reference: ダニエル・レイ・エインジ（Daniel Ray Ainge, 1959 年 3 月 17 日 - ）はオレゴン州ユージーン出身の元バスケットボール選手で元プロ野球選手（内野手）。現役時代は NBA でプレイし、1980 年代に黄金期を築いたボストン・セルティックスやフェニックス・サンズなどで活躍した。現役引退後は指導者に転向し、現在はセルティックスでジェネラル・マネージャーを務めている。野球選手としては MLB のトロント・ブルージェイズでプレイしており、NBA と MLB の二大リーグでプレイしたことがある稀有な存在である。甥のエリック・エインジは 2008 年の NFL ドラフトでニューヨーク・ジェッツに指名されて入団した。 Question: ダニエル・レイ・エインジはどこのプロ野球チームに所属した？	トロント・ブルージェイズ
レビュー分類	機械部分からの異臭が酷くて、とても不快な思いをしました。素敵なデザインだっただけに残念です。	bad

表 2: レビュー分類タスクの対象言語ラベルの設定で使用される各言語のラベルセット。英語のラベルセットは good-bad である。

対象言語	good	bad
ドイツ語	gut	schlecht
スペイン語	bueno	malo
フランス語	bon	mauvais
インドネシア語	baik	buruk
日本語	良い	悪い
朝鮮語	좋음	나쁨
中国語	好	差

2.2.3.3 MLLM

本研究は、指示文の言語による MLLM の性能の変化を分析することを目的としており、指示チューニング済みモデルに焦点を当てている。用いるモデルは Llama-3-suzume-multilingual 8B [23]、Qwen2-Instruct 7B [24]、Mistral-NeMo-Instruct 12B [25] である。これらはそれぞれ Llama 3、Qwen2、Mistral-NeMo の多言語指示チューニング済みモデルである。以降、それぞれを `llama3-i'、`qwen2-i'、`mistraln-i'と表記する。

2.2.4 実験結果

表 3 に zero-shot 設定での各タスクにおける全ての対象言語の平均の性能を示す。

語彙平易化タスク　実験結果から、対象言語指示文が英語指示文の性能を上回る傾向があることが確認された。さらに、対象言語が日本語である設定において、Translationese の悪い影響を観測した。以下に英語指示文（en）と英語から翻訳された対象言語指示文（tgt-mt）の一部を示す。

en　Please generate **a** simpler Japanese synonym for the word.
tgt-mt　より簡単な日本語の同義語を生成してください。

英語指示文では、生成するべき同義語の個数情報である `a' が含まれていることがわかる。一方で、英語から翻訳された日本語指示文は、翻訳の過程で個数情報が失われてしまい、同義語をいくつ生成するべきなのか不透明になっている。その結果、英語から翻訳された日本語指示文を用いると、MLLM は `パトカー' という単語に対して `交番車, 車両, 付近の警備車, 駆けつけ車, 警察車' のような複数の同義語を生成することがあり、性能が大幅に低下した。このことは、先行研究のような、英語指示文と英語から翻訳された対象言語指示文の比較は必ずしも公平でないことを示している。そのような偏った条件では英語指示文が効果的であると不当に評価されることになる。

機械読解タスク　実験結果から、英語指示文が対象言語指示文の性能を上回る傾向があることが確認された。この傾向は、LS の傾向とは対照的であり、英語指示文と対象言語指示文のどちらがより効果的なのかはタスクにより変化することを示している。

レビュー分類タスク　実験結果から、英語の分類ラベルを使用する設定では、英語指示文が対象言語指示文の性能を上回る傾向があることが確認された。一方で、対象言語の分類ラベルを使用する設定では、対象言語指示文が英語指示文の性能を上回る傾向がある。これらの結果は、分類タスクにおいて最適な指示文の言語は分類ラベルの言語に依存し、ラベルの言語と同じ言語の指示文がより高い性能になる傾向があることを示している。

表 3: en（英語指示文）、tgt（対象言語指示文）、tgt-mt（Bing Translator を用いて英語から翻訳された対象言語の指示文）の性能の比較。全対象言語間の平均スコアを示す。タスクごとに各モデルの最高性能を太字で強調する。

タスク	指示文	性能		
		llama3-i	qwen2-i	mistraln-i
語彙平易化	en	26.95	44.38	48.68
	tgt	**28.31**	**46.52**	**52.78**
	tgt-mt	23.33	40.64	46.12
機械読解タスク	en	**25.47**	**32.33**	**39.48**
	tgt	20.07	22.19	31.47
	tgt-mt	18.01	18.47	32.91
レビュー分類（en label）	en	**87.66**	**90.58**	**89.15**
	tgt	77.57	90.56	80.47
	tgt-mt	83.69	88.82	79.06
レビュー分類（tgt label）	en	66.72	86.49	65.34
	tgt	**70.14**	**89.46**	**65.47**
	tgt-mt	69.22	81.58	61.17

2.2.5 指示文による違い

2.2.5.1 生成テキストの特徴

機械読解タスクにおいて、英語指示文を用いる設定と対象言語指示文を用いる設定間で MLLM が生成するテキストが同じであるインスタンスの割合は llama3-i が約 30%、qwen2-i が約 37%、mistraln-i が約 48%である。これらの結果から、同じ内容を伝えるが異なる言語で書かれている 2 つの指示文に対して、MLLM は異なるテキストを生成することが多くあることがわかる。以下では、各指示文を用いたときに MLLM が生成するテキストの特徴を分析する。

英語指示文は非対象言語の生成が増加する ここでは MLLM によって生成されたテキストの言語を判別する。言語の判別には FastText [26] を用いる。先行研究 [27, 28] を参考に、FastText の言語判別の確信度が 50%以上の結果のみを用いる。表 4 に MLLM が非対象言語のテキストを生成したインスタンスの割合を示す。これらの結果は、英語指示文は非対象言語で生成することを増加させる傾向を示している。この観測は Marchisio ら [29] の報告に類似している。特に、英語指示文を用いると、MLLM は英語のテキストを生成することの増加が確認された。また、qwen2-i では英語指示文を用いると、中国語のテキストを生成することも増加する。

表 4: MLLM が対象言語以外の言語のテキストを生成するインスタンスの割合。全対象言語の平均の割合を示す。

タスク	指示文	llama3-i	qwen2-i	mistraln-i
語彙平易化	en	9.94	8.23	7.08
	tgt	7.13	6.43	6.22
機械読解タスク	en	4.33	4.36	2.98
	tgt	2.16	1.47	1.76

表 5: スペイン語と日本語の機械読解タスクにおいて、MLLM が末検出テキストを生成するインスタンスの数。

対象言語	指示文	llama3-i	qwen2-i	mistraln-i
スペイン語	en	0	1	0
	tgt	8	18	2
日本語	en	3	5	0
	tgt	28	15	3

対象言語指示文は未検出テキストの生成が増加する　機械読解タスクでは、参照テキスト中に質問に対する解答が必ず含まれる。しかしながら、``与えられた参照文には質問に対する情報がありません。" のような、情報が見つからなかったことを示すテキスト（未検出テキスト）を MLLM が生成する現象を確認した。我々はスペイン語と日本語において、MLLM がそのような未検出テキストを生成するインスタンスを人手で数えた。表 5 に MLLM が未検出テキストを生成するインスタンスの数を示す。これらの結果は、対象言語指示文を用いることは未検出テキストの生成を増加させることを示している。特に、対象言語指示文の場合は未検出テキストを生成する一方で、英語指示文の場合は正しい回答を生成するようなインスタンスがいくつか確認された。このことは、英語指示文を用いることは MLLM の読解能力を引き出すのにより効果的であることを示唆している。

2.2.5.2　指示追従度

本節では、英語指示文と対象言語指示文のそれぞれに対する MLLM の指示追従度の違いを分析する。

指示不追従の定義　MLLM がどのようなテキストを生成した場合に指示文に追従していないとするかを定義する。まず、語彙平易化タスクでは 1 つの同義語のみを生成するように MLLM に指示をする。そのため、1 つの単語やフレーズというよりも文に近いものが生成される場合に、MLLM が指示文に追従していないと定義する。生成されたテキストが文に近いかの判別には、ド

表 6: MLLM が指示文に追従するインスタンスの割合。全対象言語の平均の割合を示す。指示文間でより高い追従度を太字で強調する。

タスク	指示文	llama3-i	qwen2-i	mistraln-i
語彙平易化	en	**80.05**	**97.69**	**99.65**
	tgt	76.46	97.03	99.09
機械読解タスク	en	**54.43**	**62.51**	**72.66**
	tgt	38.86	41.67	53.10

イツ語、スペイン語、フランス語、中国語ではそのテキストが 6 単語以上 [30] かどうか、日本語では 8 単語以上 [31] かどうかを基準とする。単語分割には spaCy [32] を用いる。機械読解タスクでは参照文から答えのみを抜き出すように MLLM に指示をする。そのため、参照文に出現しない文字列が生成される場合に、MLLM が指示文に追従していないと定義する。

指示追従度の比較　表 6 に MLLM が指示文に追従するインスタンスの割合を示す。これらの結果は、全ての MLLM が、英語指示文に対してより追従することを示している。この観測は、複雑な指示を要求するタスクでは英語指示文を用いることがより効果的であることを示唆している。

2.2.5.3　活性化ニューロン

これまでの結果から、同じ内容を伝える英語指示文と対象言語指示文に対して MLLM の生成するテキストは異なることが多くあり、それぞれの生成テキストに特徴があることが明らかになった。本節ではさらなる分析として、指示文の言語が MLLM 内部に与える影響を調査するために、活性化するニューロンについて分析する。

本研究では、先行研究 [33] に基づき、各 Transformer 層の feed-forward network における活性化関数の出力をニューロンとする。また、ニューロンの値が正である場合に活性化しているとみなす。以降では MLLM は qwen2-i、タスクはレビュー分類 (en label) の設定に注目する。

指示文と活性化ニューロン　同一のインスタンスに対して英語指示文を用いるプロンプトおよび対象言語指示文を用いるプロンプトの最終トークンを処理する際のニューロンを比較する。ニューロンの活性化のパターンは以下の 4 つに分類できる：

both_act：英語指示文と対象言語指示文の両方で活性化する

both_nonact：英語指示文と対象言語指示文の両方で活性化しない

only_en_act：英語指示文でのみ活性化する

only_tgt_act：対象言語指示文でのみ活性化する

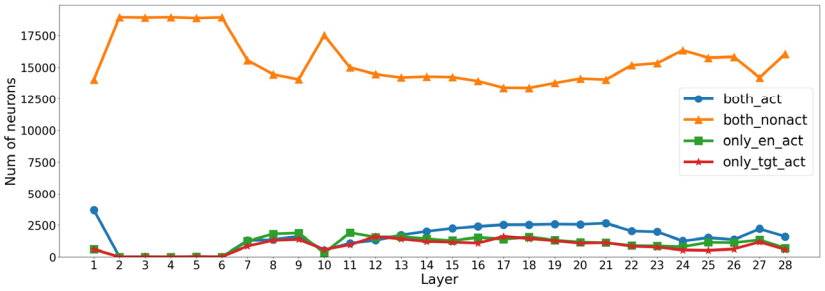

図 2: 英語指示文を用いるプロンプトおよび対象言語指示文を用いるプロンプトの最終トークン処理時のニューロンの状態の比較。

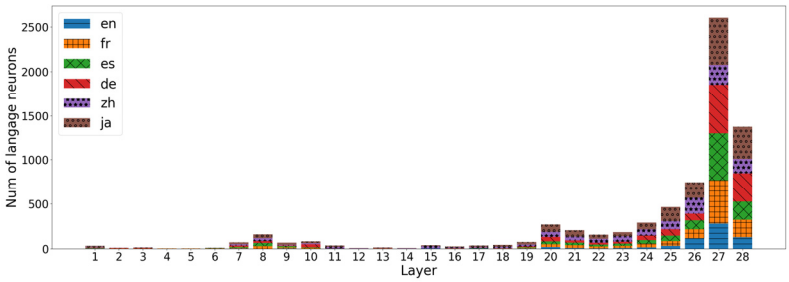

図 3: qwen2-i の各層における言語固有ニューロンの数。

図 2 に qwen2-i の各層においてそれぞれの活性化パターンに該当するニューロンの数を示す。結果として、英語指示文または対象言語指示文のみで活性化するニューロンが一定数存在することが確認された。このことは、指示文の言語に依存して活性化するニューロンが存在することを示唆している。

この結果を踏まえると、ある言語の指示文を用いる際に活性化するニューロンが、言語固有ニューロンと関連している可能性が考えられる。以降では、言語固有ニューロンを説明し、それらが指示文の言語の影響をどのように受けるのか分析する。

言語固有ニューロン　MLLM 内部で特定の言語に強く関連づけられて機能するニューロンの存在が示されており、それらは `言語固有ニューロン' と呼ばれる [28, 33]。言語固有ニューロンは、MLLM が特定の言語を処理する際に主に活性化し、他の言語ではほとんど活性化しない特徴を持つ。

本研究では、英語、ドイツ語、スペイン語、フランス語、中国語、日本語の言語固有ニューロンを特定するために LAPE [33] を用いる。言語固有のテキストコーパスには先行研究 [28] で用いられたデータを採用する。図 3 に qwen2-i の各層における言語固有ニューロンの数を示す。学習データの大部分を占める言語は言語固有ニューロンが少なくなることを Tang らは報告しており、本研究では qwen2-i では英語と中国語の言語固有ニューロンが比較的少ないことが確認された。また、qwen2-i では出力層に近い層に多く分布していることがわかった。

表 7: qwen2-i における各言語の *P(l)*。`tgt'は対象言語を示し、インスタンスの言語と一致する。 *P(en)* に水色を、*P(tgt)* に赤色をつける。

指示文	tgt	*P(l)* ×100					
		en	fr	es	de	zh	ja
en	fr	48.66	19.05	16.83	12.59	12.35	13.21
	es	48.68	15.94	22.00	12.60	12.06	13.12
	de	49.12	14.87	15.48	17.20	12.13	13.31
	zh	46.70	10.55	11.34	10.09	21.26	17.23
	ja	46.28	12.05	12.39	11.41	17.26	20.73
tgt	fr	31.30	72.14	34.94	21.81	11.90	15.27
	es	31.42	34.99	68.30	21.10	12.38	15.74
	de	33.28	26.48	24.69	66.10	13.29	18.06
	zh	24.79	8.05	7.55	8.13	50.72	23.98
	ja	23.31	13.17	13.83	18.32	32.44	59.97

指示文と言語固有ニューロンの関係　各指示文を用いるプロンプトの最終トークン処理時に、各言語の言語固有ニューロンがどの程度活性化しているかを調べるために言語ごとに以下を計算する：

$$P(l) = \frac{\text{活性化した言語 } l \text{ の言語固有ニューロンの数}}{\text{言語 } l \text{ の言語固有ニューロンの数}}$$

表 7 に各対象言語においての *P(l)* の結果を示す。結果として、インスタンスは対象言語であるにもかかわらず、英語指示文のプロンプトでは英語の言語固有ニューロンが強く活性化するのに対し、対象言語の言語固有ニューロンの活性化は弱い傾向が確認された。一方で、対象言語指示文のプロンプトでは対象言語の言語固有ニューロンが強く活性化する傾向が見られた。この結果は、指示文の言語が MLLM 内部のニューロン活性化パターンに強く影響を与え、モデルが内部で処理する際の言語的な重点が指示文の言語によって変化することを示唆している。

2.2.6　おわりに

本研究では、Translationese の影響を排除し、MLLM において英語指示文と対象言語指示文の公平な比較を行った。実験結果から、どちらの指示文がより効果的であるかはタスクや分類ラベルによって異なる傾向があることを明らかにした。また、それぞれの指示を用いた場合に生成されるテキストの特徴や指示追従度、活性化するニューロンに違いが生じることを示した。

参考文献

[1] Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P. Xing, and Zhiting Hu. PromptAgent: Strategic planning with language models enables expert-level prompt optimization. In ICLR, 2024.

[2] Ayana Niwa and Hayate Iso. AmbigNLG: Addressing task ambiguity in instruction for NLG. In EMNLP, pp. 10733–10752, 2024.

[3] Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual generative language models. In EMNLP, pp. 9019–9052, 2022.

[4] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang,Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning. In ACL, pp. 15991–16111, 2023.

[5] Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. MEGA: Multilingual evaluation of generative AI. In EMNLP, pp. 4232–4267, 2023.

[6] Sauleh Eetemadi and Kristina Toutanova. Asymmetric features of human generated translation. In EMNLP, pp. 159–164, 2014.

[7] Shuly Wintner. Translationese: Between human and machine translation. In COLING, pp. 18–19, 2016.

[8] Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. TACL, Vol. 8, pp. 454–470,2020.

[9] Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In EMNLP Findings, pp. 12365–12394, 2023.

[10] Julen Etxaniz, Gorka Azkune, Aitor Soroa, Oier Lacalle, and Mikel Artetxe. Do multilingual language models think better in English? In EMNLP, pp. 550–564, 2024.

[11] Yotam Intrator, Matan Halfon, Roman Goldenberg, Reut Tsarfaty, Matan Eyal, Ehud Rivlin, Yossi Matias, and Natalia Aizenberg. Breaking the language barrier: Can direct inference outperform pre-translation in multilingual LLM applications? In NAACL, pp. 829–844, 2024.

[12] Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In EMNLP, pp. 2475–2485, 2018.

[13] Patrick Bareiß, Roman Klinger, and Jeremy Barnes. English prompts are better for NLI-based zero-shot emotion classification than target-language prompts. In ACM Web Conference, WWW '24, p. 1318–1326, 2024.

[14] Matthew Shardlow, Fernando Alva-Manchego, Riza Batista Navarro, Stefan Bott, Saul Calderon Ramirez, R´emi Cardon, Thomas Franc¸ois, Akio Hayakawa, Andrea Horbach, Anna H¨ulsing, Yusuke Ide, Joseph Marvin Imperial, Adam Nohejl, Kai North, Laura Occhipinti, Nelson Per´ez Rojas, Nishat Raihan, Tharindu Ranasinghe, Martin Solis Salazar, Marcos Zampieri, and Horacio Saggion. An extensible massively multilingual lexical simplification pipeline dataset using the MultiLS framework. In READI, pp. 38–46, 2024.

[15] Jipeng Qiang, Kang Liu, Ying Li, Yun Li, Yi Zhu, Yun-Hao Yuan, Xiaocheng Hu, and Xiaoye Ouyang. Chinese lexical substitution: Dataset and method. In EMNLP, pp. 29–42, 2023.

[16] Timo M¨oller, Julian Risch, and Malte Pietsch. GermanQuAD and GermanDPR: Improving non-English question answering and passage retrieval. In MRQA, pp. 42–50, 2021.

[17] Asier Guti´errez-Fandi˜no, Jordi Armengol-Estap´e, Marc P`amies, Joan Llop-Palao, Joaquin Silveira-Ocampo, Casimiro Pio Carrino, Carme Armentano-Oller, Carlos Rodriguez-Penagos, Aitor Gonzalez-Agirre, and Marta Villegas. MarIA: Spanish language models. Procesamiento del Lenguaje Natural, p. 39–60, 2022.

[18] Martin d'Hoffschmidt, Wacim Belblidia, Quentin Heinrich, Tom Brendl´e, and Maxime Vidal. FQuAD: French question answering dataset. In EMNLP Findings, pp. 1193–1208, 2020.

[19] Chih Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. DRCD: a Chinese machine reading comprehension dataset,2019.

[20] Phillip Keung, Yichao Lu, Gy¨orgy Szarvas, and Noah A. Smith. The multilingual Amazon reviews corpus. In EMNLP, pp. 4563–4568, 2020.

[21] Lucy Park. Naver sentiment movie corpus v1.0, 2015.

[22] Rhio Sutoyo, Said Achmad, Andry Chowanda, Esther Widhi Andangsari, and Sani M. Isa. PRDECT-ID: Indonesian product reviews dataset for emotions classification tasks. Data in Brief, Vol. 44, p. 108554, 2022.

[23] Peter Devine. Tagengo: A multilingual chat dataset. In MRL, pp.106–113, 2024.

[24] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report, 2024.

[25] MistralAI. Mistral NeMo, 2024.

[26] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, H´erve J´egou, and Tomas Mikolov. FastText.zip: Compressing text classification models, 2016.

[27] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzm´an, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In LREC, pp. 4003–4012, 2020.

[28] Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. In NAACL, pp. 6919–6971, 2024.

[29] Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Th´eo Dehaze, and Sebastian Ruder. Understanding and mitigating language confusion in LLMs. In EMNLP, pp. 6653–6677, 2024.

[30] Yuri Lin, Jean-Baptiste Michel, Erez Aiden Lieberman, Jon Orwant, Will Brockman, and Slav Petrov. Syntactic annotations for the Google Books NGram corpus. In Min Zhang, editor, Proceedings of the ACL 2012 System Demonstrations, pp. 169–174, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[31] Taku Kudo and Hideto Kazawa. Japanese web n-gram version 1, 2009.

[32] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength natural language processing in python. 2020.

[33] Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. Language-specific neurons: The key to multilingual capabilities in large language models. In ACL, pp. 5701–5715, 2024.

2.3 潜在変数付き Transformer を用いた機械翻訳の性能検証

愛媛大学　小倉　知也
愛媛大学　二宮　　崇
愛媛大学　後藤　功雄

2.3.1　はじめに

近年、グローバル化とインターネットの普及が進む中で、異なる言語を用いる人々が交流する機会が増加しており、国際的なコミュニケーションの重要性が高まり続けている。このような現代社会において、言語の壁がコミュニケーションを阻む大きな障害となりうるため、翻訳技術の進展は非常に重要である。人手による翻訳は、コストと時間の面で課題となることが多いが、深層ニューラルネットワークを用いた様々なニューラル機械翻訳モデル[1–5]が登場し、特にTransformer[5]が提案されたことにより、高速かつ高精度な機械翻訳が提供されている。

しかしながら、Transformer を含む従来のニューラル機械翻訳モデルでは同一の入力文に対し単一の値をもつベクトルに従って機械翻訳を行うため、単一の翻訳文しか生成されず、翻訳に多様性が得られないという問題がある。翻訳において、翻訳元の単語と翻訳先の単語が一対一で対応するようなことは稀であり、多くの場合同じ意味をもちながら異なる形で表現される翻訳候補がいくつか存在する。例えば、「Thank you for your help.」という英語文が与えられたとき、日本語文への翻訳として「手伝ってくれてありがとう。」や「あなたのご協力に感謝しています。」、「助けてくれてありがとう。」など多様な表現が考えられる。このような翻訳における表現方法の多様さは、言語にニュアンスの違いをもたらす。その場面や雰囲気に適したニュアンスの違いを考慮することは、柔軟なコミュニケーションを実現するために重要な要素である。

本稿では、このような翻訳の多様さに焦点を当て、従来のニューラル機械翻訳モデルであるTransformer[5]に確率分布に従う潜在変数を導入する手法を提案する。提案手法により、多様性に富んだ翻訳文が生成されることが期待される。また、提案手法では、潜在変数の結合方法として二つの手法「Concat」と「Add」を提案する。機械翻訳タスクにおける提案手法の翻訳性能と多様性を評価するために、ASPEC データセット[6]を用いて英語文から日本語文への機械翻訳タスクの実験をした。実験の結果、翻訳性能の評価において、提案手法はベースラインと比較して、表層的な類似度を測る評価指標と意味的な類似度を測る評価指標の両方で低下したが、表層的な類似度を測る評価指標の低下幅は大きかった一方、意味的類似度を測る評価指標の低下幅は小さかった。また、多様性の評価において、提案手法はベースラインと比較して多様性が向上した。二つの結合手法を比較すると、「Concat」よりも「Add」の方が高い多様性を示した。

2.3.2　従来法

本節では、提案手法の潜在変数の導入に関わる潜在変数モデル Variational AutoEncoder（以下、VAE）[7]、Conditional Variational AutoEncoder（以下、CVAE）[8]、Transformer-Based Conditioned Variational Autoencoder（以下、T-CVAE）[9]について説明する。

2.3.2.1 VAE

VAE（Variational AutoEncoder）[7]は、データに内在する観測できない隠れた構造を潜在変数として表現する潜在変数モデルの一つであり、主に確率分布に従って生成した潜在変数を用いて、データの生成と再構成を行う深層ニューラルネットワークモデルである。VAE は、Encoder と Decoder から構成される。Encoder は、観測データを入力として受け取り、確率分布に従う潜在変数に変換する役割を担う。具体的には、入力データを受け取り、確率分布の形状を決定するパラメータを出力する。例えば、確率分布として正規分布を仮定した場合、Encoder は平均と分散を出力する。Encoder から出力された平均と分散を用いて潜在変数を取得するときに、VAE では再パラメータ化トリックを用いて潜在変数を取得することで、誤差逆伝播に基づくニューラルネットワークの学習を実現する。再パラメータ化トリックでは、標準正規分布に従うノイズ $\epsilon \sim N(0,1)$ と、Encoder から出力された平均 μ と標準偏差 σ を用いて、以下の式のように潜在変数を算出する。

$$z = \mu + \epsilon \cdot \sigma$$

Decoder は、サンプリングされた潜在変数を入力として受け取り、元のデータを再構成するようにデータ生成を行う。

2.3.2.2 CVAE

CVAE（Conditional Variational AutoEncoder）[8]は、通常の VAE を拡張したモデルであり、入力データに加えて条件付けデータも考慮した潜在変数モデルである。CVAE は、VAE と同様に Encoder と Decoder から構成される。VAE との主な違いは、Encoder と Decoder の入力に追加でラベルデータを条件付けデータとして組み込む部分である。この拡張により、CVAE は、データと条件付けデータと潜在変数の間の関係を学習することができ、条件付きデータ生成ができるという特徴がある。例えば、数字の手書き文字データを生成する場合、特定の数字ラベルを条件として与えることで、その数字ラベルに基づいて文字データを生成できる。

2.3.2.3 T-CVAE

T-CVAE（Transformer-Based Conditioned Variational Autoencoder）[9]は、Transformer の上に CVAE を構築したモデルである。なお、先行研究では、T-CVAE はストーリー補完タスク用に設計されている。ストーリー補完タスクとは、不完全なストーリーに対して欠けているプロットを生成するストーリー補完タスクである。CVAE による確率分布に従う潜在変数を導入することで、Transformer の出力に多様性をもたらし、より多様なプロットの生成を実現している。以下では、ストーリー補完タスクのための T-CVAE がどのように学習されるかを説明する。不完全なストーリー x および欠落したプロット y が与えられたとき、T-CVAE は不完全なストーリー x のみから潜在変数の事前分布 $p(z|x)$ を推定し、不完全なストーリー x と欠落したプロット y から潜在変数の事後分布 $q(z|x,y)$ を推定する。具体的には、Transformer Encoder の出力を多層ニューラルネットワークに通すことで潜在変数の事前分布を推定（分布の平均と標準偏差を計算）し、

Transformer Decoder の出力を全結合層に通すことで潜在変数の事後分布を推定（分布の平均と標準偏差を計算）する。そして、推定された分布からサンプリングされた潜在変数 z を用いて欠落したプロットの生成を行う。

学習時は、以下の式で表される変分下限を最大化することによって、T-CVAE モデルの最適化を行う。

$$\log p(y|x) = \log \int_z p(y|x,z)\, p(z|x)\, dz$$
$$\geq \mathbb{E}_{q(z|x,y)}[\log p(y|x,z)\,] - D_{KL}[q(z|x,y) \parallel p(z|x)]$$

D_{KL} はカルバック・ライブラー・ダイバージェンス（KL ダイバージェンス）である。この変分下限に従って T-CVAE の最適化が進むと、KL ダイバージェンスが小さくなるように、すなわち、不完全なストーリー x のみから推定した潜在変数の事前分布と、不完全なストーリー x と欠落したプロット y の両方から推定した潜在変数の事後分布が近づくように学習が進む。これにより、推論時にも、不完全なストーリーから適切な事前分布を推定した上で欠落したプロットを予測できる。

2.3.3 提案手法

本節では、提案手法となる確率分布に従う潜在変数を Transformer に導入した機械翻訳について説明する。まず、提案手法の学習手法について説明する。そして、潜在変数の結合部分において、提案手法で採用した「Concat」と「Add」という二つの結合手法について説明する。また、学習時の工夫点である KL アニーリングについても説明する。

2.3.3.1 提案手法の学習方法

提案手法では、多様性のある機械翻訳を目指して、Transformer に確率分布に従う潜在変数を導入する。先行研究である T-CVAE[9]を参考に、潜在変数モデル CVAE を利用することで確率分布に従う潜在変数を取得する。しかし、先行研究の T-CVAE は、ストーリー補完タスク用に提案されたものであるため、Transformer Encoder と Transformer Decoder の間で Attention 層を共有している。この設定は、原言語文と目的言語文が異なる分布をもつ機械翻訳タスクには適していないため、提案手法では、Transformer Encoder と Transformer Decoder の間で Attention 層を共有しない、機械翻訳タスク用の Transformer の上に CVAE を構築する。

提案手法のモデルアーキテクチャを図 1 に示す。提案手法は、図 1 のように、二つの Transformer Encoder と二つの Transformer Decoder、そして、潜在変数の事前分布と事後分布を推定するための VAE Encoder と CVAE Encoder から構成される。なお、これらの構成要素は全て異なる別々のモデルである。

以下では、提案手法モデルがどのように学習されるかを説明する。原言語文 x および目的言語文 y が与えられたとき、提案手法モデルは原言語文 x のみから潜在変数の事前分布 $p(z|x)$ を推定し、原言語文 x と目的言語文 y から潜在変数の事後分布 $q(z|x,y)$ を推定する。具体的には、原言語文を Transformer Encoder に入力して得られる出力を、VAE Encoder に通すことで潜在変数

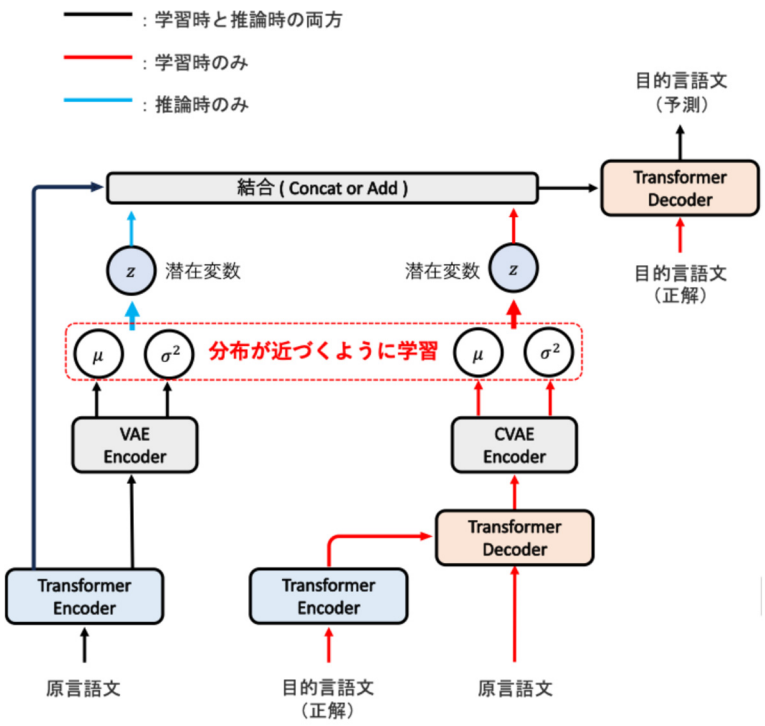

図 1:　提案手法のモデルアーキテクチャ

の事前分布の平均と標準偏差を推定する。

また、目的言語文を Transformer Encoder に入力して得られる出力と原言語文の両方を Transformer Decoder に入力して得られる出力を、CVAE Encoder に通すことで潜在変数の事後分布の平均と標準偏差を推定する。そして、推定されたこれらの分布から、再パラメータ化トリックを用いて潜在変数を決定的にサンプリングする。学習時には、事後分布からサンプリングされた潜在変数を、原言語文を Transformer Encoder に入力して得られる出力と結合することで潜在変数結合ベクトルを生成する。そして、この潜在変数結合ベクトルと目的言語文の両方を Transformer Decoder へ入力することで目的言語文の生成を行う。損失は、潜在変数の事前分布と事後分布の間の類似度を測る KL ダイバージェンスと、生成した目的言語文と正解の目的言語文との誤差を測る NLL Loss の和を使用し、この損失を最小化するように学習を進める。なお、提案手法では、潜在変数の事前分布と潜在変数の事後分布として正規分布を仮定しているため、$p(z|x) \sim N(\mu_1, \sigma_1^2)$、$q(z|x,y) \sim N(\mu_2, \sigma_2^2)$ と定義される。このとき、正規分布間の KL ダイバージェンス[11]は以下の式で算出される。

$$D_{KL}(p \parallel q) = \int_{-\infty}^{\infty} p(z|x) \ln \frac{p(z|x)}{q(z|x,y)}\, dx = \ln\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\,\sigma_2^2} - \frac{1}{2}$$

推論時には目的言語文が与えられないため、事前分布からサンプリングされた潜在変数を、原言語文を Transformer Encoder に入力して得られる出力と結合することで潜在変数結合ベクトルを生成する。そして、この潜在変数結合ベクトルを Transformer Decoder へ入力することで目的言語文の生成を行う。

2.3.3.2 潜在変数の結合

事前分布または事後分布からサンプリングされた潜在変数を、原言語文を Transformer Encoder に入力して得られる出力と結合し、潜在変数結合ベクトルを生成する方法として、「Concat」と「Add」の二つの結合手法を説明する。「Concat」は、事前分布または事後分布からサンプリングされた潜在変数を、原言語文を Transformer Encoder に入力して得られる出力と直接連結する結合手法である。提案手法では、原言語文の先頭単語の直前の位置に連結した。「Add」は、事前分布または事後分布からサンプリングされた潜在変数を、原言語文を Transformer Encoder に入力して得られる出力に加算する結合手法である。ただし、潜在変数は 1 階テンソルで、Transformer Encoder 出力は 2 階テンソルであるため、このままではテンソル同士で加算できない。そのため、提案手法では、潜在変数を原言語文の単語系列の長さ分だけ複製することで形状を揃えた後に、テンソル同士で加算した。

2.3.3.3 KL アニーリング

提案手法モデルの学習時の工夫として、KL アニーリング[11]を適用している。これは KL vanishing[12]という問題に対処するためのものである。KL vanishing とは、学習時に変分下限の項 $\mathbb{E}_{q(z|x,y)}[\log p(y|x,z)]$ で表される損失が最適化される前に、KL ダイバージェンスの損失について過度に最適化されてしまい、その結果、モデルが潜在変数 z を考慮しなくなるという問題である。KL アニーリングにより、KL ダイバージェンスの損失について過度に最適化されてしまう、すなわち、KL ダイバージェンスの損失が小さくなりすぎてしまうことを防ぐために、定数 b より小さくならないように、以下の式を用いて設定する。

$$\sum_{k=1}^{|x|} \max(b, D_{KL}[q(z|x,y) \parallel p(z|x)])$$

なお、定数 b は、以下の式を用いて調整する。

$$b = 1 \qquad \text{if } s < \frac{M}{2}$$

$$b = \frac{(M-1)}{\frac{M}{2}} \quad \text{otherwise}$$

ただし、s は学習ステップ数を表し、M は最大学習ステップ数である。

2.3.4 実験

2.3.4.1 実験設定

提案手法の機械翻訳タスクにおける性能を評価するために、英語文から日本語文への機械翻訳タスクの実験をした。提案手法として、「Concat」により潜在変数を連結して結合したモデルと、「Add」により潜在変数を加算して結合したモデルの 2 種類を用いた。また、Transformer 単体のモデルをベースラインとした。データセットは、英語文と日本語文ペアの対訳コーパスからな

る ASPEC[6]データセットを使用した。Moses[13]の clean-corpus-n.perl を用いて、ASPEC の train1.txt ファイルからトークン系列の長さが 101 以上の文を除いた上位 100,000 件を学習データとした。また、検証データとして dev.txt ファイル（1,790 件）を使用し、評価データとしての test.txt ファイル（1,812 件）を使用した。なお、英語文の前処理には Moses の tokenizer.perl を用い、日本語文の前処理には KyTea[14]を用いた。

Transformer 内の埋め込みベクトル次元数は 512、Feed Forward 層における中間層のベクトル次元数 2,048、Transformer Encoder および Decoder の繰り返し処理は 6 回、MultiHeadAttention 層の Head 数は 8、ドロップアウト率[15]は 0.1 とした。学習時のバッチサイズは 64、エポック数は 20、最適化手法は Adam[16]を使用し、学習率は Transformer の元論文[5]と同様に以下の式で設定した。

$$d_{model}^{-0.5} \cdot \min(stepnum^{-0.5}, stepnum \cdot warmupsteps^{-1.5})$$

ただし、d_{model} は Transformer 内の埋め込みベクトル次元数である 512、$stepnum$ は学習ステップ数のことであり、$warmupsteps$ は 4,000 とした。推論時には、ビームサイズを 4 としたビームサーチ探索[17]を使用した。

提案手法の機械翻訳タスクにおける性能を、翻訳性能と多様性の二つの観点で評価した。まず、評価データに対してモデルが生成した翻訳文と正解翻訳文から BLEU と BERTScore を算出し、翻訳性能を評価した。また、多様性を評価するために、推論時のランダムシードを 10 回変えて機械翻訳を行い、モデルが生成した 10 個の翻訳文の間で類似度を測定した。この類似度が小さいほど、多様性のある翻訳文が生成されているといえる。具体的には、モデルが生成した 10 個の翻訳文の全ての組み合わせ（90 通り）で BLEU を算出し、その平均値を多様性の評価値とした。

2.3.4.2 実験結果

翻訳性能に関する評価実験から得られた結果を表 1 に示し、多様性に関する評価実験から得られた結果を表 2 に示す。また、多様性に関する評価実験で得られた翻訳文の一例を表 3 に示す。ただし、各モデルで得られた 10 個の翻訳文のうち、重複した文は削除して表にまとめている。

表 1 の翻訳性能に関する評価においては、ベースラインと比較して提案手法「Concat」では BLEU が 0.96 ポイント低下、BERTScore が 0.01 ポイント低下した。提案手法「Add」では BLEU が 0.85 ポイント低下、BERTScore が 0.01 ポイント低下した。提案手法における BLEU の低下幅が、BERTScore の低下幅よりも目立つ結果となった。BLEU が二文間の表層的な類似度を測る評価指標であることと、BERTScore が意味的な類似度を測る評価指標であることを考慮すると、提案手法が生成した翻訳文は、意味的な正しさは保持しながらも表層的な単語の差異が増加したといえる。つまり、提案手法では、確率分布に従う潜在変数の影響から、より多様性のある翻訳文を生成できたと考えられる。

表 2 の多様性に関する評価では、BLEU がベースラインでは 100.00 ポイント、提案手法「Concat」では 80.32 ポイント、提案手法「Add」では 59.15 ポイントであり、提案手法の BLEU がベースラインと比較して小さくなった。これは、提案手法が確率分布に従う潜在変数を導入しており、この潜在変数がランダムシードによって変化したことでモデルが生成した翻訳文に多様性が生ま

表 1：翻訳性能に関する評価値

モデル	BLEU↑	$F_{BERTScore}$↑
Transformer	**31.62**	**0.85**
CVAE-Transformer-Concat	30.66	0.84
CVAE-Transformer-Add	30.77	0.84

表 2：多様性に関する評価値

モデル	BLEU↓
Transformer	100.00
CVAE-Transformer-Concat	80.32
CVAE-Transformer-Add	**59.15**

表 3：多様性に関する評価実験で得られた翻訳文の一例（重複削除済み）

	原文
	Research and development of alkali and phosphate types were finalized.
モデル	翻訳文
Transformer	アルカリ型とりん酸塩型の研究開発を行った。
CVAE-Transformer-Concat	アルカリ及びりん酸塩型の研究開発を行った。
	アルカリ・りん酸塩型の研究開発を行った。
CVAE-Transformer-Add	アルカリ型とりん酸塩型の研究開発を進めた。
	アルカリ型およびりん酸塩型の研究開発を行った。
	アルカリ・りん酸塩型の研究開発を行った。
	アルカリ型及びりん酸塩型の研究開発を行った。

れていると考えられる。なお、推論時の Transformer はランダムシードによって変化する要素がないため、モデルが生成した翻訳文も変化せず、BLEU は 100.00 となっている。提案手法の中では「Add」の方が「Concat」よりも BLEU が小さくなった。つまり、提案手法「Add」の方が提案手法「Concat」よりも多様性のある翻訳文を生成しているといえる。表 3 の翻訳文の一例を見ても、提案手法「Add」の方が提案手法「Concat」よりも多様性に富んだ翻訳文を生成していることが確かめられる。また、翻訳性能において提案手法「Concat」と提案手法「Add」の間で評価値の差がほとんどないことから、提案手法「Add」は機械翻訳の性能を保ちながら多様性を高めているといえる。

2.3.4.3 考察

多様性の評価においては、ベースラインと比較して提案手法「Concat」と提案手法「Add」の両方で BLEU が低下し、提案手法「Concat」では BLEU が 80.32 ポイント、提案手法「Add」では BLEU が 59.15 ポイントだった。提案手法の中では「Concat」よりも「Add」の方が BLEU が小さいため、提案手法「Concat」よりも提案手法「Add」の方が多様性のある翻訳文を生成したといえる。このような結果となった理由として、提案手法「Concat」では潜在変数をそのまま結合したが、提案手法「Add」では潜在変数を原言語文の単語系列の長さ分だけ複製した後に結合したため、提案手法「Add」の方がより潜在変数の影響を大きく受けたと考えられる。翻訳性能の評価において提案手法「Add」と提案手法「Concat」の間で評価値の差がほとんどないことを考

慮すると、多様性のある機械翻訳のための手法として提案手法「Add」の方がより良い手法であるといえる。

2.3.5 まとめ

本稿では、機械翻訳の翻訳性能と多様性の向上を目指して、CVAE に基づく確率分布に従う潜在変数を Transformer に導入した機械翻訳手法を提案した。実験の結果、提案手法はベースラインと比較して、表層的な類似度を測る評価指標と意味的な類似度を測る評価指標の両方で低下したが、意味的類似度を測る評価指標の低下幅は小さかった。また、多様性の評価において、提案手法はベースラインと比較して多様性が向上することがわかった。今後、言い換え生成や逆翻訳によるデータ拡張など、多様な翻訳を必要とする応用にこれらの手法が用いられることが期待される。

参考文献

[1] Ilya Sutskever、 Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In Advances in Neural Information Processing Systems, Vol. 27, 2014.

[2] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, 2016.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In 3rd International Conference on Learning Representations, 2015.

[4] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421, 2015.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In Advances in Neural Information Processing Systems 30, pp. 5998–6008. 2017.

[6] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian Scientific Paper Excerpt Corpus. In Proceedings of the Tenth International Conference on Language Resources and Evaluation, pp. 2204–2208, 2016.

[7] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In 2nd International Conference on Learning Representations, 2014, Conference Track Proceedings, 2014.

[8] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised Learning with Deep Generative Models. In Advances in Neural Information Processing Systems, Vol. 27, 2014.

[9] Tianming Wang and Xiaojun Wan. T-CVAE: Transformer-Based Conditioned Variational Autoencoder for Story Completion. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, pp. 5233–5239, 2019.

[10] Joram Soch and Carsten Allefeld. Kullback-Leibler Divergence for the Normal-Gamma Distribution, 2016.

[11] Raphael Shu, Jason Lee, Hideki Nakayama, and Kyunghyun Cho. Latent-variable non-autoregressive neural machine translation with deterministic inference using a delta posterior. In AAAI 2020 - 34th AAAI Conference on Artificial Intelligence, pp. 8846–8853, 2020.

[12] Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical Annealing Schedule: A Simple Approach to Mitigating KL Vanishing. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 240–250, 2019.

[13] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondˇrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, pp. 177–180, 2007.

[14] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 529–533, 2011.

[15] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, Vol. 15, No. 56, pp. 1929–1958, 2014.

[16] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In 3rd International Conference on Learning Representations, 2015.

[17] Felix Stahlberg and Bill Byrne. On NMT Search Errors and Model Errors: Cat Got Your Tongue? In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 3356–3362, 2019.

3. シンポジウム開催報告

3. シンポジウム開催報告：第 8 回特許情報シンポジウム

静岡大学　綱川　隆司
（シンポジウム副実行委員長）

3.1　開催概要

特許情報シンポジウムはアジア太平洋機械翻訳協会および日本特許情報機構の主催で 2010 年に第 1 回が開催されて以来ほぼ隔年で開催されており、2024 年 11 月 7 日（木）に第 8 回特許情報シンポジウムを開催した。2021 年に開催された第 6 回以降はオンライン形式による開催となっている。シンポジウムは本研究会の委員がプログラム委員として企画し、当日は綱川が副実行委員長として司会進行を務め、事前登録者数 151 名に対して当日の同時最大参加人数は 96 名程度に達した。例年、特許情報に関連する研究者、実務に携わる方、および政府関係者による発表および議論を行っており、今回は 5 名の招待講演者からの講演およびパネルディスカッションが実施された。シンポジウム終了後のアンケート結果から総じて満足度は高く、成功裏に終了したといえる。

3.2　開催概要

3.2.1　招待講演（1）

岩永 寛道 氏（特許庁総務部総務課　課長補佐）
講演題目「特許庁における AI の活用の取組について」

特許庁による AI 技術の活用について、特許、商標、意匠の実体審査業務の効率化・高度化を図る取組みが紹介された。過去からの実証事業や各年度のアクションプラン策定を経て、機械翻訳、先行技術検索、画像検索、分類付与、ランキング表示など具体的なツールを開発し、審査の質向上と業務効率化に寄与する試みが展開されている。

3.2.2　招待講演（2）

野中 尋史 氏（愛知工業大学経営学部経営学科　准教授）
講演題目「特許を対象とするスコアリングモデルおよび大規模言語モデルの研究」

特許の権利期間や引用情報などの素性を用いて、DeepSurv などのニューラルハザードモデルを適用し、特許権利期間の予測を試みるとともに、技術の重要性や成長性を定量化するため、特許の価値評価を目的としたスコアリングモデルの開発と大規模言語モデルの活用に関する野中氏の研究グループによる研究が紹介された。企業の研究開発や M&A などの意思決定に有用な評価指標の整備を目標としている。

3.2.3　招待講演（３）

谷川 英和 氏（IRD 国際特許事務所　所長）

講演題目「生成ＡＩを用いた特許文書品質向上のための取り組み」

生成 AI や機械学習を活用して特許文書の品質向上を図る取り組みについて紹介された。特許実務の効率化と品質向上を目指し、発明着想から出願までの各フェーズにおいて、生成 AI による定性的評価や、機械学習を用いた定量的評価、さらに品質チェックツールの開発を通じ、特許文書の正確性や明確性、読みやすさを向上させる方法が説明された。

3.2.4　招待講演（４）

大澤 豊 氏（大澤特許事務所　所長）

講演題目「弁理士業務への生成ＡＩの活用可能性と将来像 ～一人の実務者の視点から～」

大澤氏の講演では生成 AI の弁理士業務への応用可能性とその将来展望が示された。生成 AI を利用した翻訳や特許調査、文書作成の効率化について具体例を交えながら解説し、現状の課題や制約（誤回答、情報更新の遅れ等）も指摘された。さらに、将来的には特定用途向けの AI が普及し、外部依頼から社内処理へ移行する可能性や、業務効率化による収益向上が期待される。

3.2.5　招待講演（５）

永田 昌明 氏（NTT コミュニケーション科学基礎研究所）

講演題目「JaParaPat: 大規模日英特許対訳コーパス」

長田氏の講演では、日本と米国の特許出願データから約 100 万文対の対訳文書を収集して作成された大規模日英特許対訳コーパス JaParaPat について紹介された。生成 AI やビッグデータ解析を用いて特許文献の精度向上、効率的な情報検索・評価の自動化が進められており、これら技術が弁理士業務や知財戦略に与える影響、今後の業務変革の可能性についての展望が示された。

3.2.6　パネルディスカッション

モデレータ：須藤 克仁 氏

パネリスト：岩永 寛道 氏、野中 尋史 氏、大澤 豊 氏、永田 昌明 氏

パネルディスカッションでは、以下の二つの事前質問をベースに各パネリストによる議論が展開された。

① 生成 AI は特許等知財の創出や強化につながるか？

② 今本邦に求められる生成 AI に関わる技術は何か？

岩永氏は、生成 AI が特許審査における分類付与、検索式提案、要約生成などで効率化と質向上に寄与できると述べ、野中氏は技術創造やデータ整備の重要性を指摘した。大澤氏は、低コストで新規着想を得るブレインストーミングとしての有用性を評価する一方、知的財産権の創出力低

下への懸念も示した。永田氏は、論文からの特許出願生成の可能性や日本語検索精度の向上、特許庁のデータ提供強化が必要であるとの意見が交わされ、全体として生成 AI の活用は期待されつつも、信頼性やデータ整備、法的整合性の確保が課題であるとの結論に至った。

3.3 所感

特許情報の処理においても生成 AI を中心とした AI 技術は注目されつつあるものの、実務上で浸透しているのは一部にとどまり、研究向けや実務向けのデータの整備や AI 出力の正確さの確保、英語と比較したときの日本語での性能強化など、特許分野において多くの課題があることが浮き彫りになったと感じている。一方、AI の性能が向上する中で特許情報処理のような専門的で困難さを持った分野は今後の研究分野として焦点が当たる可能性をもっており、継続した研究開発が望まれるところである。

2024年度AAMT/Japio特許翻訳研究会報告書

発 行 日　2025 年 3 月

発　　行　一般財団法人　日本特許情報機構（Japio）
〒135−0016　東京都江東区東陽町 4 丁目1番 7 号
佐藤ダイヤビルディング
TEL：(03) 3615−5511　FAX：(03) 3615−5521

編　　集　一般社団法人　アジア太平洋機械翻訳協会（AAMT）