特許翻訳の二段階自動後編集における事前訓練済みエンコーダーモデル・LLMの評価

Evaluation of Pretrained Encoder Models and LLMs for Two-Step Automatic Post-Editing in Patent Machine Translation

筑波大学　システム情報工学研究群　知能機能システム学位プログラム　博士前期課程 2 年
武馬　光星

2025 年筑波大学理工学群工学システム学類卒業。現在、筑波大学大学院システム情報工学研究群知能機能システム学位プログラム博士前期課程在学中。大規模言語モデル・機械翻訳の研究に従事。

筑波大学　システム情報系知能機能工学域　教授
宇津呂　武仁

1994 年京都大学大学院工学研究科電気工学第二専攻博士課程修了。博士 (工学)。京都大学 等を経て、2012 年より筑波大学システム情報系知能機能工学域教授。自然言語処理、大規模言語モデル、機械翻訳、ウェブマイニングの研究に従事。電子情報通信学会、情報処理学会、人工知能学会、言語処理学会、ACL 各会員。

NTTコミュニケーション科学基礎研究所　上席特別研究員
永田　昌明

1987 年京都大学大学院工学研究科修士課程修了。同年、日本電信電話株式会社入社。現在、NTT コミュニケーション科学基礎研究所 上席特別研究員。博士 (工学)。自然言語処理の研究に従事。電子情報通信学会、情報処理学会、人工知能学会、言語処理学会、ACL 各会員。

1 はじめに

近年の大規模言語モデル（Large Language Models; LLMs）の進歩により、LLMRefine[28] など、LLM の多段階推論による翻訳後編集手法が実現されている。また、Google の多段階パイプライン [3] のような大規模・体系的な設計も提案されている。しかし、機械翻訳パイプラインのすべての構成要素が LLM に依存する必要はない。特に、誤り検出タスクに関しては、事前学習済みエンコーダモデルを用いることで、より高精度かつ低コストに実現できる可能性がある [18]。Lukito[17] らは、ソーシャルメディア上の接続表現を検出する分類タスクにおいて、BERT ベースの分類器が GPT-3.5 Turbo を精度・再現率・F1 スコアのいずれの指標でも大きく上回ることを示している。

本研究では、トークン単位の誤り検出と LLM による訂正を組み合わせた二段階翻訳後編集（図 1）を提案する [4]。第 1 段階では、学習した多言語エンコーダモデルを用いてトークン単位での誤訳検出を行う。日英特許翻訳には誤りアノテーション付きデータセットが存在しないため、対訳特許データのターゲット文に人工誤りを付与し、トークン単位の誤訳検出用合成データを構築した。第 2 段階では、検出された誤りタグに基づき、LLM（GPT-4o）[19] が翻訳文を訂正する。

本研究では、法的および技術的要件が極めて厳格であり、翻訳誤りが許されずポストエディットが不可欠な特許分野において提案手法を評価した。誤り検出に関しては、日英および英日特許データセットを用いて学習済み多言語エンコーダモデルを評価した。その結果、F1 スコアおよび Matthews 相関係数（MCC）のいずれにおいても、LLM ベース手法を上回る性能を示し、トークン単位での誤訳検出能力の優位性が確認された。

翻訳訂正に関しては、(1) 人工的に誤りを挿入した文、(2) 繰り返し誤りを含む文、(3) 訳抜けを含む文の 3 種類のデータセットで実験を行った。その結果、誤り検出に多言語エンコーダモデルを用い、その後に LLM で翻訳文訂正を行う本提案手法は、BLEU スコア [20] において LLM を使ったベースライン手法を上回る性能を示した。一方で、

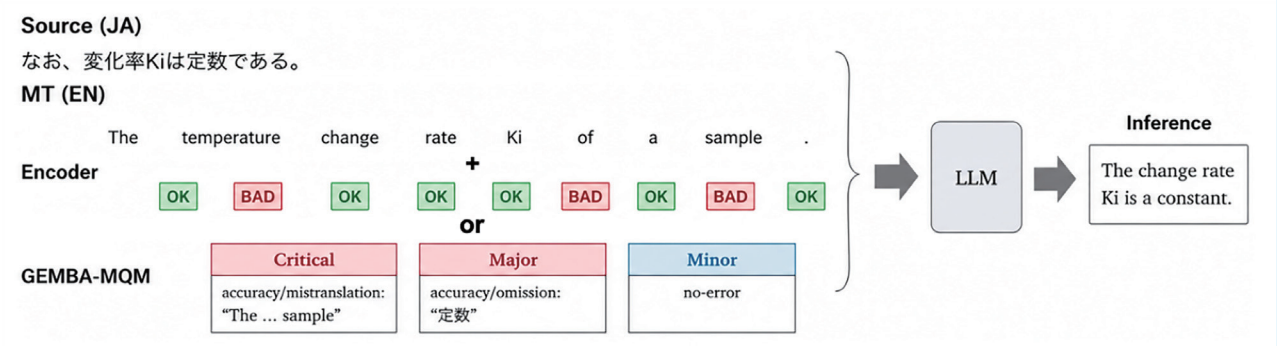

図１　提案手法の概要　第一段階で誤訳検出を行い、第二段階で誤訳訂正する

訳抜け誤りの訂正は依然として課題として残る。

以上の結果から、多段階 LLM 推論は強力であるものの、軽量なエンコーダモデルを選択的に統合することで、より高精度かつ効率的な機械翻訳誤り検出・訂正が可能であることが示唆された。

本研究の貢献は以下の３点にまとめられる：

・作成した合成特許データでエンコーダモデルを学習することで、一般的に高性能とされる LLM を上回る誤り検出精度を達成した。

・誤りを人工的に挿入した特許文データを構築することで、手作業によるアノテーションを必要としない教師あり学習を実現した。

・提案するエンコーダモデル -LLM 手法は、LLM のベースライン手法と比較して翻訳品質において統計的に有意な改善を示した。

2 関連研究

2.1　翻訳における単語レベル品質評価

単語レベル品質推定（word-level QE）は、機械翻訳（MT）の各トークンおよびギャップ位置に対して、OK／BAD のラベルを付与するタスクとして定式化されている。この設定は、WMT 共有タスクおよびその成果報告書 [8,25,29] を通じて標準化されており、参照訳を必要としないトークン／ギャップ単位でのニューラル手法や評価基準の発展を促進してきた。

初期のニューラルアーキテクチャとして、Predictor-Estimator フレームワーク [15] が挙げられる。この手法は、大規模な並列コーパスで学習した「予測器（Predictor）」と、品質推定（QE）アノテーション付きデータで学習した「推定器（Estimator）」を明確に分離する構成をとり、WMT17 で最高性能を達成した。その設計思想は後続のオープンソースツールキットに影響を与え、たとえば OpenKiwi[13] は、PyTorch 上で単語レベルおよび文レベル QE を統合的に扱う最先端システムを実装している。さらに、多言語事前学習済みエンコーダモデルを基盤とする TransQuest[22] は、WMT20 で最先端の性能を達成するとともに、高い多言語転移性能を示した。

本研究とより近い設定として、Wei らは bert-base-multilingual-cased（mBERT）に基づく教師あり単語レベル QE モデルを提案している [27]。このモデルでは、原文と機械翻訳文を連結した入力を与え、回帰ヘッドが各翻訳トークンの BAD ラベルとなる確率を推定する。本研究でもこの教師あり・トークンレベルの定式化を採用するが、一般ドメインの WMT データとは異なり、専門用語や文体の特異性が顕著な特許分野に適用する点が特徴である。また、多言語エンコーダモデルは、単一言語対にとどまらず、単語レベル QE における高いクロスリンガル汎化能力を示している [23]。

一方で、翻訳品質評価指標（MT metrics）の研究では、文レベルスコアからスパンレベルのフィードバックへと関心が移行している。Rei らは、COMET[24] を導入し、Guerreiro らは、これを拡張した xCOMET[11] を提案している。xCOMET は、文レベル評価に加えて誤りスパンの帰属も可能とし、WMT ベンチマークにおいて高い性能を示している。また、Alves らは、指標の頑健性を検証するため、幻覚（hallucination）、削除（deletion）、誤訳（mistranslation）などの制御された誤りを導入する合成誤り生成器 SMAUG を提案している [1]。xCOMET が主にメトリクスの頑健性評価のために合成誤りを利用しているのに対し、本研究では合成誤りを教師信号として用い、トークンレベルの誤り検出モデルを学習し、これを LLM による訂正工程の誘導に活用する。

2.2　LLM ベースの品質評価

近年、大規模言語モデル（Large Language Models; LLMs）は、参照訳を必要としないスパンレベルの機械翻訳（MT）評価者として活用されている。Kocmi らは、GEMBA-MQM[16] を提案した。これは GPT-4 を用いた評価手法であり、MQM フレームワーク [5] に基づき、固定の 3 ショットプロンプトを用いて誤りスパンおよび誤り種別を特定するものである。参照訳を必要としないにもかかわらず、WMT23 における実験では、システムレベルおよびセグメントレベルの双方で人手 MQM 評価との高い相関を示した。

一方で、近年の評価研究により、LLM ベース評価の限界も指摘されている。LLM ベースの指標は頑健性に欠ける傾向があり、バイアスや安定性に関する懸念が存在する。また、より広範な分析では、LLM による評価がプロンプト設計に敏感であり、しばしば複数の評価基準を混同する可能性があることが報告されており、信頼性の低下を招く場合がある [2]。これらの知見は、LLM ベース評価の慎重な運用の必要性を示すとともに、可能な場合にはスパンレベルの解釈可能なフィードバックや学習済みメトリクスとの併用の重要性を示唆している。

2.3　機械翻訳における後編集

Deguchi らは、自動後編集（Automatic Post-Editing; APE）を二つの解釈可能な段階に分解する Detector-Corrector フレームワークを提案している [6]。この手法では、XLM-RoBERTa に基づく検出器（detector）が 3 種類の二値分類タスク—MT-tag、MT-gap、SRC-tag—を通じて誤りスパンを特定し、その後、検出されたスパンのみを編集する訂正器（corrector）が動作する。この編集ベースのパイプラインは、翻訳編集距離（Translation Edit Rate; TER）の改善に加え、検出器による明示的な根拠に基づいて修正を説明可能にする点でも優れている。

本研究もこの二段階の直観を踏襲するが、検出器には多言語エンコーダモデルを採用し、日本語 - 英語特許の教師データで学習を行う。また、訂正器としては、検出器がマークしたスパンのみを修正するよう指示された LLM を組み合わせている。

一方で、LLM を用いたポストエディット手法も近年注目を集めている。Ki らは、外部からの MQM スタイルのフィードバックを LLM に与える手法を提案し、一般的なスコアからスパン／タイプ単位の詳細なアノテーションに至るまで、異なる粒度でのフィードバックを比較した [14]。その結果、中国語 – 英語、英語 – ドイツ語、英語 – ロシア語の各言語対において、TER、BLEU、COMET の全てで一貫した性能向上を確認し、特に粒度の高いフィードバックが最も大きな改善をもたらすことを示した。

また、Xu らは、LLMRefine を提案している [28]。これは、学習済みフィードバックモデルを用いて誤りを逐次的に特定・修正する反復的な枠組みであり、翻訳品質の向上を実現している。これらの手法が LLM による自由度の高い再翻訳を許容するのに対し、本研究では制約付き後編集を採用する。

3　誤訳検出

3.1　エンコーダモデルを用いた誤訳検出

事前学習済み多言語エンコーダモデルである mBERT、XLM-RoBERTa、mDeBERTa[12] を用いて、機械翻訳におけるトークンレベルの品質推定を行う。具体的には、エンコーダモデルが持つ事前学習知識を活用し、翻訳誤りを検出して各トークンに適切な誤りラベルを付与する。

エンコーダモデルの学習には、Deguchi らが行ったデータ拡張手法 [6] に従い、NTCIR-7[9]（1,798,571 文対）および NTCIR-8[10]（3,186,284 文対）の特許

表 1　本研究で使用したデータセットの統計情報

	Ja→En			En→Ja		
データセット	文数	トークン数 (Ja)	トークン数 (En)	文数	トークン数 (En)	トークン数 (Ja)
学習	8,000	268,247	254,239	8,000	246,885	280,313
開発	1,000	37,062	34,128	1,000	33,160	38,670
テスト (合成誤り)	200	7,016	6,869	200	6,680	7,278
テスト (繰り返し誤り)	200	26,454	19,805	-	-	-
テスト (訳抜け誤り)	200	67,613	11,545	-	-	-

平行コーパスから 1 万文をサンプリングし、以下の操作を適用して人工誤りデータを生成する。

- **削除（Deletion）**：トークンを 5% の確率で削除する
- **挿入（Insertion）**：トークンを 10% の確率で挿入する
- **置換（Replacement）**：トークンを 30% の確率で置換する

これらの操作確率は、Deguchi らの設定 [6] に基づいて設定している。挿入および置換に関しては、mBERT を用いて [MASK] トークンを補完する手法を採用する。具体的には、対象位置に [MASK] トークンを挿入し、mBERT により適切な単語を予測させる。その際、予測語と元の語の類似度を低減するため、Transformers Pipelines の出力候補の中からスコアが最も低いトークンを選択する。これにより人工的に誤りを生成し、誤りトークンには BAD タグを、その他のトークンには OK タグを付与して、教師あり学習用の学習データセットを構築する。この方法により、エンコーダモデルの学習用に 8,000 文のアノテーション付きデータを生成した。

3.2 LLM を用いた誤訳検出

LLM を用いた誤訳検出のために、Kocmi らが提案した GPT ベースの評価手法である GEMBA-MQM[16] を採用する。GEMBA-MQM フレームワークに基づき、以下の 2 つの設定の下で誤り検出を行う。

- **0-shot**：事前の例示を与えずに誤り検出を実施する。
- **3-shot**：言語に依存しない 3 つの例示を与えた上で誤り検出を実施する。

Kocmi らの報告 [16] によれば、これらの設定のうち 3-shot 設定が GPT-4 を用いた際に最も高い誤り検出精度を示している。エンコーダモデルおよび LLM による誤訳検出は、後続の翻訳訂正プロセスの前処理として用いる。

表 2　合成誤りに対する誤訳検出の評価（日英翻訳）

モデル	ラベル	Precision	Recall	F1
GPT-4o (0-shot)	OK	0.843	0.077	0.142
	BAD	0.389	0.976	0.556
	TOTAL	F1: 0.298,	MCC: 0.111	
GPT-4o (3-shot)	OK	0.755	0.268	0.395
	BAD	0.409	0.853	0.553
	TOTAL	F1: 0.454,	MCC: 0.141	
mBERT	OK	0.855	0.883	0.869
	BAD	0.785	0.740	0.762
	TOTAL	F1: 0.830,	MCC: 0.631	
XLM-RoBERTa	OK	0.924	0.924	0.924
	BAD	0.868	0.868	0.868
	TOTAL	F1: 0.903,	MCC: 0.792	
mDeBERTa	OK	0.935	0.944	**0.940**
	BAD	0.901	0.887	**0.894**
	TOTAL	F1: **0.923**,	MCC: **0.834**	

表 3　合成誤りに対する誤訳検出の評価（日英翻訳）

モデル	ラベル	Precision	Recall	F1
GPT-4o (0-shot)	OK	0.804	0.287	0.423
	BAD	0.415	0.879	0.563
	TOTAL	F1: 0.474,	MCC: 0.190	
GPT-4o (3-shot)	OK	0.709	0.503	0.588
	BAD	0.419	0.634	0.504
	TOTAL	F1: 0.558,	MCC: 0.132	
mBERT	OK	0.870	0.880	0.875
	BAD	0.784	0.769	0.776
	TOTAL	F1: 0.839,	MCC: 0.655	
XLM-RoBERTa	OK	0.918	0.935	0.926
	BAD	0.881	0.852	0.866
	TOTAL	F1: 0.904,	MCC: 0.792	
mDeBERTa	OK	0.936	0.946	**0.941**
	BAD	0.903	0.885	**0.894**
	TOTAL	F1: **0.924**,	MCC: **0.835**	

4 LLM を用いた誤訳訂正

原文と翻訳文を入力として与え、LLM が翻訳誤りの内容と位置を分析したうえで訂正文を生成する。また、各訂正の根拠を提示することで、訂正過程の透明性と解釈可能性を高める。

さらに本研究では、前節で述べたエンコーダによる誤訳検出の結果を LLM への入力として利用する翻訳訂正手法を提案する。本実験では、以下の 2 種類のプロンプト設定を検証した。1 つ目は、第 1 段階の誤訳検出結果を参照しながら翻訳訂正を行うよう LLM に指示するプロンプトであり、2 つ目は、第 1 段階で誤りと判定された部分のみを修正し、それ以外の部分は変更しないよう指示するプロンプトである。

これらの設定を通じて、エンコーダモデルまたは LLM による誤り検出結果を翻訳訂正プロセスに組み込むことで、翻訳訂正の精度をさらに向上させることを目指す。

5 評価

5.1 データセット

本研究では、誤訳（mistranslation）、繰り返し誤り（repetition）、および訳抜け（omission）に焦点を当てる。これらの誤りタイプは特許機械翻訳において観察

されるだけでなく、特許文書において重要な意味的忠実性に深刻な影響を及ぼす。本研究では、以下の 3 種類のデータセットを用いて提案手法を評価する。

- **合成誤り特許データセット**（合成データ）
- **繰り返し誤り特許請求項データセット**（非合成データ）
- **訳抜け誤り特許請求項データセット**（非合成データ）

データセットの詳細（文数、トークン数、その他の統計情報を含む）は、表 1 に示す。

合成誤り特許データは、NTCIR-7 および NTCIR-8 の日本語 – 英語対訳特許文に対して、3.1 節で述べた手法に基づき、人工的に誤りを付与して生成したものである。合成誤り特許データを 200 文作成し、誤り検出および訂正能力の評価に使用した。さらに、繰り返し誤りおよび訳抜け誤りに対する訂正性能を評価するため、Transformer[26] により生成された特許請求項の日本語 – 英語翻訳文から、以下の基準に基づいて文を抽出した。

- **繰り返し誤り文**：参照訳の 2 倍以上の長さを持つ翻訳文
- **訳抜け誤り文**：参照訳の半分以下の長さを持つ翻訳文

評価には、2021 年に公開された特許文書の請求項から抽出した翻訳文を用いた。データの品質を確保するため、LaBSE 埋め込み [7] を用いて原文と参照訳の文類似度を算出し、類似度スコアが 0.8 以上 0.98 以下のペアのみを抽出した。その結果、繰り返し誤り文および訳抜け誤り文をそれぞれ200文ずつ評価データとして使用した。

5.2　評価手順

5.2.1　誤訳検出

翻訳文中の各トークンには、誤り位置に対して BAD タグを付与し、トークンレベルでの評価を可能にしている。エンコーダモデルおよび LLM はいずれも、図 1 に示すようにトークン単位でのタグ付けを行った。日本語のトークン化には、MeCab と UniDic 辞書を併用した。翻訳誤り検出の評価には、以下のモデルを用いた。

1. **LLM（GPT-4o）– 0-shot**
2. **LLM（GPT-4o）– 3-shot**
3. **mBERT**
4. **XLM-RoBERTa**
5. **mDeBERTa**

評価指標には、F1 スコアおよび Matthews 相関係数（MCC）を用いた。

5.2.2　誤訳訂正

誤り訂正の評価には、人工的に誤りを付与した特許文、繰り返し誤りを含む特許請求項文、そして訳抜け誤りを含む特許請求項文の 3 種類のデータを使用した。訂正処理は LLM を用いて行い、入力として原文、翻訳文、およびエンコーダモデルまたは LLM による誤り検出結果を与えた。使用モデルは GPT-4o（gpt-4o-2024-08-06）であり、デコーディングには貪欲法（temperature = 0.0）を用いた。特に指定がない限り、その他のパラメータはデフォルト設定を採用した。

評価に使用したモデルの組み合わせは以下の通りである。

1. LLM 単独訂正：
誤り検出を行わず、LLM（GPT-4o）のみで単一ステップの翻訳訂正を実施。

2. LLM 検出（GEMBA-MQM, 0-shot）＋ LLM 訂正：
GEMBA-MQM（0-shot）を用いた LLM（GPT-4o）による誤り検出の後、LLM（GPT-4o）による翻訳訂正を実施。

3. LLM 検出（GEMBA-MQM, 3-shot）＋ LLM 訂正：
GEMBA-MQM（3-shot）を用いた LLM（GPT-4o）による誤り検出の後、LLM（GPT-4o）による翻訳訂正を実施。

4. mBERT 検出＋ LLM 訂正：
mBERT によるトークンレベルの誤り検出の後、LLM（GPT-4o）による翻訳訂正を実施。

5. XLM-RoBERTa 検出＋ LLM 訂正：
XLM-RoBERTa によるトークンレベルの誤り検出の後、LLM（GPT-4o）による翻訳訂正を実施。

6. mDeBERTa 検出＋ LLM 訂正：
mDeBERTa によるトークンレベルの誤り検出の後、LLM（GPT-4o）による翻訳訂正を実施。

ベースラインおよび提案手法：

ベースライン：LLM 検出（GEMBA-MQM, 3-shot）＋ LLM 訂正（制約なし）

LLM は GEMBA-MQM（3-shot）を用いて誤り検出を行い、続く訂正段階では翻訳全体の任意の部分を編集可能（非制約的）とする。

提案手法：エンコーダベース検出＋ LLM 訂正（制約付き）

表 4　翻訳訂正の BLEU スコア。各セルの表記 x/y は、x が制約なし（unrestricted）訂正（翻訳文中の任意の部分を修正可能）による LLM 訂正スコア、y が制約付き（restricted）訂正（検出された誤り部分のみを修正）による LLM 訂正スコアを示す。Synthetic（Ja → En）および Synthetic（En → Ja）の列における Δ は、y とベースラインスコア（下線付きスコア）との差分を表す。Repetition および Omission の列では、Δ はさらに x とベースラインスコアとの差分も併記し、制約なし訂正を適用した場合の性能変化を示している。Δ に付されたアスタリスク（*）は、ベースラインとの間に統計的に有意な差があることを示す（$p < 0.05$）。

	手法	合成誤り（日英）		合成誤り（英日）		繰り返し誤り		訳抜け誤り	
		BLEU (x / y)	Δ	BLEU (x / y)	Δ	BLEU (x / y)	Δ	BLEU (x / y)	Δ
1	訂正なし	31.69	−14.72	32.63	−6.46	21.49	−4.79	19.09	−45.40
2	LLM 単独訂正	47.25	+0.84	40.35	+1.26 *	26.70	+0.42	62.50	−1.99
3	LLM 検出（GEMBA-MQM, 0-shot）＋ LLM 訂正	43.88 / 44.05	−2.36	37.58 / 39.96	+0.87	26.16 / **27.25**	+0.97	62.31 / 62.82	−1.67
4	**Baseline:** LLM 検出（GEMBA-MQM, 3-shot）＋ LLM 訂正	46.41 / 48.14	+1.73 *	39.09 / 42.05	+2.96 *	26.28 / 26.58	+0.30	**64.49** / **65.62**	+1.13
5	**Proposed:** mBERT 検出＋ LLM 訂正	48.44 / 49.76	**+3.35** *	39.10 / 42.61	**+3.52** *	27.42 / 23.46	**+1.14** * / −2.82	59.73 / 28.00	−4.76 / −36.49
6	**Proposed:** XLM-RoBERTa 検出＋ LLM 訂正	**48.83** / **50.71**	**+4.30** *	39.21 / **43.76**	**+4.67** *	27.41 / 23.15	**+1.13** * / −3.13	56.38 / 25.57	−8.11 / −38.92
7	**Proposed:** mDeBERTa 検出＋ LLM 訂正	48.03 / 50.69	**+4.28** *	**39.22** / **43.76**	**+4.67** *	**28.15** / 22.63	**+1.87** * / −3.65	57.73 / 22.60	−6.76 / −41.89

エンコーダモデル（mBERT、XLM-RoBERTa、または mDeBERTa）により誤り検出を行い、訂正段階では検出器が誤りとしてマークしたスパンのみを修正対象とし、それ以外のトークンは変更不可とする（制約付き訂正）。

英日翻訳訂正においては、LLM の出力結果に文字間のスペースが挿入される場合があったため、これらのスペースを除去した。

訂正後の翻訳は、sacreBLEU（v2.4.3）[21] を用いて算出した BLEU スコアにより評価した。BLEU は、システム翻訳と参照訳の n-gram 一致率を測定する代表的な自動評価指標であり、高い忠実性が求められる特許翻訳の評価に適している。表 4 に示す BLEU スコアの向上が統計的に有意かを検証するため、SacreBLEU に実装されている paired-bootstrap 再標本化検定（-paired-bs オプション）を用いて有意差検定を実施した。

5.3　評価結果

5.3.1 誤訳検出の評価

合成誤りデータを用いた評価（表 2,3）において、学習を行ったエンコーダモデルは、両方向で LLM ベースの検出手法を大きく上回る性能を示した。特に、mDeBERTa が最も高い精度を達成し、続いて XLM-RoBERTa および mBERT が高いスコアを示した。一方、GPT-4o を誤訳検出器として用いた場合は、3-shot プロンプトを適用しても精度が大きく劣り、0-shot 設定ではさらに低下した。GPT-4o の出力分析の結果、ほとんどのトークンに対して BAD タグを付与する傾向が確認された。そのため、BAD タグの再現率は比較的高い一方で、OK タグの再現率が著しく低下することが判明した。

これらの結果から、合成誤りデータを用いたコンパクトなエンコーダモデルの教師あり学習は、LLM プロンプトによる誤り検出よりも、トークンレベルの誤訳検出において高い効果を発揮することが示唆された。

5.3.2 誤訳訂正の評価

表 4 に示すように、最適な手法は誤りの種類および翻訳方向によって異なる。いずれの設定においても、検出器と訂正器を組み合わせた手法は、訂正前を一貫して上回った。提案するエンコーダモデルを検出器とし、LLM を訂正器とした構成は性能を有意に向上させたものの、すべての代替手法に対して一律に優れているわけではなかった。

合成誤り（日英）においては、提案手法が LLM のみで訂正を行う手法および LLM ベース検出手法を上回った。最も高いスコアは XLM-RoBERTa 検出＋ LLM 訂正で得られ、mBERT 検出＋ LLM 訂正がそれに続いた。いずれも LLM 単独訂正を上回っている。表 5 に示す定性的な例から、これらのパイプラインは操作された入力中の誤訳を安定的に修正しており、トークンレベルの誤りタグが LLM 訂正の有効な手がかりとして機能していることが確認できる。

合成誤り（英日）では、第 1 段階の誤り検出結果を参照して翻訳を修正するように LLM へ指示した場合、LLM 単独訂正が最高の BLEU を達成した。しかし、第 1 段階で検出された誤りスパンのみを修正し、それ以

表５　提案手法による合成誤りの訂正例

原文:
ステップＳ１１において、プライマリプーリ１１への入力トルクを計算する。
参照訳文:
In a step S11 , an input torque to the primary pulley 11 is calculated .
合成誤り文:
In a processing stepd , The input torque to be primary pulley 11 is achieved :
提案手法の出力:
In step S11, the input torque to primary pulley 11 is calculated.

表６　提案手法による繰り返し誤りの訂正例

原文:
a. 配電ハードウェアの構成部品として、少なくとも１つの受動電磁センサをインストールするステップと、 b.
参照訳文:
a.Installing at least one Passive Electromagnetic Sensor as a component of distribution hardware;b.
機械翻訳文:
a. installing at least one passive electromagnetic sensor as a component part of the electrical distribution hardware;b. controlling the at least one passive electromagnetic sensor to emit electromagnetic radiation;c. controlling the at least one passive electromagnetic sensor to emit electromagnetic radiation;d. controlling the at least one passive electromagnetic sensor to emit electromagnetic radiation;e. controlling the at least one passive electromagnetic sensor to emit electromagnetic radiation;f. ... controlling the at least one passive electromagnetic sensor to emit electromagnetic radiation;g
提案手法の出力:
a. installing at least one passive electromagnetic sensor as a component part of the electrical distribution hardware; b.

表７　提案手法による訳抜け誤りの訂正例

原文:
前記 NK 細胞は、血液または細胞株に由来し、好ましくは、細胞株に由来し、より好ましくは、前記細胞株に由来する NK は NK92 細胞株であることを特徴とする請求項 12 に記載の免疫細胞。
参照訳文:
The immune cell of claim 12, wherein the NK cell is derived from blood or a cell line; preferably, from a cell line; and more preferably, the NK cell from a cell line is NK92 cell line.
機械翻訳文:
The immune cell of claim 12, wherein the NK cell is derived from blood or a cell line.
提案手法の出力:
The immune cell of claim 12, characterized in that the NK cell is derived from blood or a cell line, preferably from a cell line, and more preferably from the aforementioned cell line, specifically the NK92 cell line.

外を保持するようにプロンプトを変更すると、提案手法が LLM 単独訂正を上回り、XLM-RoBERTa または mDeBERTa 検出＋ LLM 訂正により最高スコアを達成した。この傾向は合成誤り（日英）においても観察され、第２のプロンプト設定の方がより高いスコアを示した。これらの結果は、第１段階の高精度な誤り検出が翻訳訂正全体の品質向上に寄与していることを示唆している。

繰り返し誤り（Repetition）に対しては、mDeBERTa 検出＋ LLM 訂正が最高の BLEU を示し、次いで mBERT 検出＋ LLM 訂正が続いた。表６の例から、提案手法は重複したスパンを効果的に除去しつつ、句読点や体裁を保持していることが確認できる。これらの結果は、提案手法が繰り返し誤りにも有効であることを示している。

一般的に、制約なし訂正では高い BLEU スコアが得られる傾向にある一方で、訂正範囲を検出器が指摘したスパンに限定すると、多くのエンコーダ -LLM 構成で性能低下が見られた。このことは、誤り検出結果の利用と LLM 固有の訂正能力とのバランスを取ることの重要性を示している。

検出結果の分析から、この性能低下は第１段階の検出において繰り返し誤りを十分に識別できなかったことに起因すると考えられる。したがって、制約付き手法が制約なし手法と同等以上の性能を発揮するためには、誤り検出の精度および再現率の双方を向上させることが重要である。特許請求項文は一般的な特許文よりも長文である傾向があるため、今後は長文データを用いてエンコーダモデルを学習させ、長文処理（特に繰り返し構造）への対応を強化する必要がある。

訳抜け誤り（Omission）に関しては、LLM 検出（3-shot）＋ LLM 訂正が最も高い性能を示し、次いで LLM 単独訂正が続いた。表４に示すように、未訂正翻訳の

BLEU スコアが 19.09 であったのに対し、提案手法では 56.65 と大幅な向上が見られた。表 7 の例では、提案手法が特許翻訳における欠落情報を効果的に復元している様子が確認できる。一方で、エンコーダベース提案手法はこの誤りタイプにおいて劣った結果となった。誤訳や繰り返し誤りが既存のターゲットトークンに基づいて観測可能であるのに対し、訳抜けはターゲット側のトークンから直接的に観測できないためであることが考えられる。これにより、ターゲット側のトークンタグ付けでは検出性能が制限される一方で、アラインメントに基づくシーケンス構造を考慮した検出（例：対応するターゲットが存在しないソーストークンの特定）は、訳抜け検出により適していると考えられる。したがって、今後はアラインメント情報に基づく信号を統合することで、訳抜け誤りへの対応範囲を拡張することが有望である。

6 おわりに

本研究では、特許文書における誤訳検出のために学習した事前学習済み多言語エンコーダモデルと、LLM による訂正処理を組み合わせることで、BLEU スコアにおいて統計的に有意な改善が得られることを示した。特に、誤訳および繰り返し誤りの処理において他の手法を上回る性能を示した。エンコーダモデルによる高精度な誤り検出が、LLM による誤りトークンの訂正を効果的に支援し、翻訳全体の品質向上に寄与していることが確認された。

さらに、人工的に生成した特許データで学習を行うことにより、人手アノテーションに依存せずに誤り検出モデルを訓練できることを示した。これらの結果から、誤りデータで訓練されたエンコーダベースモデルは、一般に幅広いタスクで高性能を示す LLM をも上回りうることが示唆された。特に、特許翻訳のような専門的領域における誤り検出といった特定タスクにおいて、その有効性が確認された。

一方で、訳抜け誤りに関しては、誤り検出と訂正の両方を LLM のみで行うモデルが提案手法を上回る結果となり、現行のトークンレベルタグ付け手法の限界が明らかとなった。これらの結果は、誤りの種類に応じて訂正戦略および誤り表現手法を最適化することが、翻訳品質のさらなる向上に不可欠であることを示している。

参考文献

[1] D. Alves et al. Robust MT evaluation with sentence-level multilingual augmentation. In Proc. 7th WMT, pp. 469–478, 2022.

[2] A. Bavaresco et al. LLMs instead of human judges? a large scale empirical study across 20 NLP evaluation tasks. In Proc. 63rd ACL, pp. 238–255, 2025.

[3] E. Briakou et al. Translating step-by-step: Decomposing the translation process for improved translation quality of long-form texts. In Proc. 9th WMT, pp. 1301–1317, 2024.

[4] K. Buma, et al. Two step automatic post editing of patent machine translation based on pretrained encoder models and LLMs. In Proc. 14th IJCNLP and 4th AACL: SRW, pp. 218–231, 2025.

[5] A. Burchardt. Multidimensional quality metrics: a flexible system for assessing translation quality. In Proc. TC, pp. 1–7, 2013.

[6] H. Deguchi et al. Detector-corrector: Edit-based automatic post editing for human post editing. In Proc. 25th EAMT, pp. 191–206, 2024.

[7] F. Feng et al. Language-agnostic BERT sentence embedding. In Proc. 60th ACL, pp. 878–891, 2022.

[8] E. Fonseca et al. Findings of the WMT 2019 shared tasks on quality estimation. In Proc. 4th WMT, pp. 1–10, 2019.

[9] A. Fujii et al. Overview of the patent translation task at the NTCIR-7 workshop. In Proc. 7th NTCIR, pp. 389–400, 2008.

[10] A. Fujii et al. Overview of the patent translation task at the NTCIR-8 workshop. In Proc. 8th NTCIR, pp. 371–376, 2010.

[11] N. Guerreiro et al. xCOMET: Transparent machine translation evaluation through fine-grained error detection. TACL, pp.

979–995, 2024.
[12] P. He et al. DeBERTaV3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.
[13] F. Kepler et al. OpenKiwi: An open source framework for quality estimation. In Proc. 57th ACL, pp. 117–122, 2019.
[14] D. Ki et al. Guiding large language models to post-edit machine translation with error annotations. In Findings of NAACL, pp. 4253–4273, 2024.
[15] H. Kim et al. Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation. In Proc. 2nd WMT, pp. 562–568, 2017.
[16] T. Kocmi et al. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In Proc. 8th WMT, pp. 768–775, 2023.
[17] J. Lukito et al. Comparing a BERT classifier and a GPT classifier for detecting connective language across multiple social media. In Proc. EMNLP, pp. 19140–19153, 2024.
[18] Motasem S Obeidat et al. Do llms surpass encoders for biomedical NER?, 2025.
[19] OpenAI. GPT-4o system card, 2024. https://arxiv.org/abs/2410.21276.
[20] K. Papineni et al. BLEU: a method for automatic evaluation of machine translation. In Proc. 40th ACL, pp. 311–318, 2002.
[21] M. Post. A call for clarity in reporting BLEU scores. In Proc. 3rd WMT, pp. 186–191, 2018.
[22] T. Ranasinghe et al. TransQuest: Translation quality estimation with cross-lingual transformers. In Proc. 28th COLING, pp. 5070–5081, 2020.
[23] T. Ranasinghe et al. An exploratory analysis of multilingual word-level quality estimation with cross-lingual transformers. In Proc. 59th ACL and the 11th IJCNLP, pp. 434–440, 2021.
[24] R. Rei et al. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In Proc. 7th WMT, pp. 578–585, 2022.
[25] L. Specia et al. Findings of the WMT 2018 shared task on quality estimation. In Proc. 3rd WMT, pp. 689–709, 2018.
[26] A. Vaswani et al. Attention is all you need. In Proc. 31st NeurIPS, pp. 5998–6008, 2017.
[27] Y. Wei et al. Extending word-level quality estimation for post-editing assistance, 2022. https://arxiv.org/abs/2209.11378.
[28] W. Xu et al. LLMRefine: Pinpointing and refining large language models via fine-grained actionable feedback. In Findings of NAACL, pp. 1429–1445, 2024.
[29] C. Zerva et al. Findings of the WMT 2022 shared task on quality estimation. In Proc. 7th WMT, pp. 69–99, 2022.