浙江財経大學

2024-2025 学年度 第 2 学期

『自然言語処理』コース論文

題目：Transformer アーキテクチャに基づく日中翻訳の実装

学生氏名：　吴国涛（ゴ コクトウ）

学生番号：　220110900830

所属学院：　情報技術･人工知能学院

専攻名称：　人工智能

クラス　：　22 人工知能 2 組

2025 年 6 月

自然言語処理 Transformer アーキテクチャの研究———

Transformer アーキテクチャに基づく日中翻訳の実装

要旨: 近年、中日間における科学技術、文化、教育などの分野での交流はますます頻繁になっており、中日間の言語翻訳精度の向上は無視できない課題となっている。本研究では、PyTorch フレームワークに基づいて完全な Transformer アーキテクチャを実装し、自己注意機構（Self-Attention Mechanism）を利用して長距離依存の問題を最適化することで、基礎的な日本語から中国語への翻訳機能を実現した。プロジェクトには、データの前処理、エンコーダ・デコーダ構造の構築、学習および評価のプロセスが含まれる。Transformer アーキテクチャを包括的に理解し、その設計の有効性を証明する。実証実験により、機械翻訳における Transformer の能力が検証された。

キーワード: Transformer、機械翻訳、自己注意機構、エンコーダ、デコーダ

1. はじめに

現在、自然言語処理（NLP）の分野において、Transformer アーキテクチャは機械翻訳タスクにおいて主導的な地位を占めている。これは、自己注意機構の導入により、従来の RNN や LSTM モデルと比較して処理能力が優れ、学習効率が高いためである。英語の国際的な性質から、大部分の研究は主に英語の処理を対象としており、独特な言語的特徴（敬語体系、漢字の多義性など）を持ち、かつ交流需要が盛んな非主流言語である日本語に対する研究には顕著な不足が存在する。デコーダのみのモデルが特定の場面で優れた性能を示しているものの、完全な Transformer アーキテクチャは依然として NLP 技術を深く理解する上で高い価値を持つ。

グローバル化の背景において、アジアの大国である日中両国は少なくない課題に直面している。既存のシステムは高度に発展した現代の言語を正確に変換することが難しく、教育・科学研究分野の専門用語の翻訳には偏差があり、意味の不一致という現象もしばしば見られる。さらに、中日文化における敬語体系の違いは、不適切な処理によっては誤解や非礼な振る舞いを引き起こす可能性さえある。このような背景において、高品質な日中機械翻訳モデルを開発

することは、技術革新のニーズであるだけでなく、中日関係の発展を促進し、中国の国際的影響力を高めるための重要な戦略的措置でもある。本研究は、多国間協力や文明交流の促進に対して、深遠な現実的意義と戦略的価値を有する。

2．技術概要

2.1Transformer：

Transformerは、自己注意機構（Self-Attention Mechanism）に基づく深層学習モデルアーキテクチャであり、2017 年に Vaswani らによる論文『Attention is All You Need』で初めて提案された。このアーキテクチャは自然言語処理（NLP）分野で画期的な進歩を遂げ、現代の NLP モデルの中核的な基盤となっている。従来の再帰型ニューラルネットワーク（RNN）や畳み込みニューラルネットワーク（CNN）と比較して、Transformerは長いシーケンスデータを処理する際に顕著な優位性を示し、長距離の依存関係を効果的に捉え、シーケンスが長すぎることによる情報の損失という従来手法の問題を回避できる。さらに、Transformerはデータの順序処理に依存せず、並列計算メカニズムによって学習効率を大幅に向上させた。その核心コンポーネントであるマルチヘッドアテンション（Multi-Head Attention）により、モデルは入力シーケンスの異なる部分に同時に注意を向けることができ、モデルの表現能力と意味理解能力が強化されている。

2.2 入力と出力の処理：

NLP タスクにおいて、Transformer モデルの入力は通常テキストデータであり、出力は具体的なタスク（翻訳、分類、生成など）によって異なる。Transformerの入力処理は主に以下の 3 つのステップを含む。

2.2. Tokenization（分かち書き/トークン化）：

テキストデータはまず、モデルが処理可能な形式、すなわちトークン（Token）に変換される必要がある。このプロセスは Tokenizationによって実現され、テキストを単語、文字、または文字などの基本単位に分割し、定義済みの語彙リストにマッピングする。モデルによって異なる分かち書き戦略や語彙リストが採用される。例えば、GPT-3 モデルが中英文を処理する場合、40 万

の英語トークンや 80 万の中国語トークンを生成する可能性がある一方、ChatGLM モデルなどは異なる数を生成する場合がある。

2.2.2 Embedding（埋め込み）:

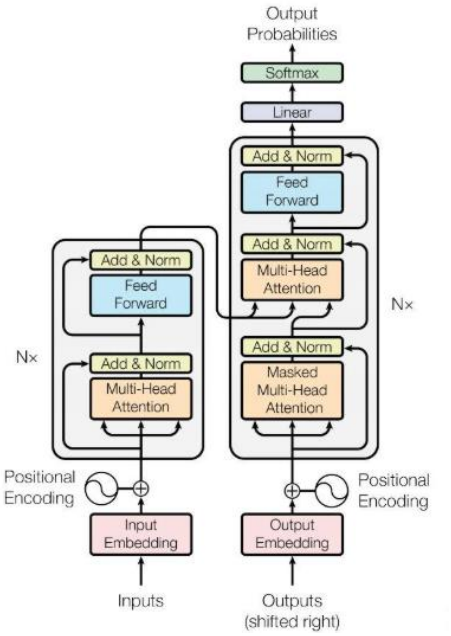
Tokenization の後、各トークンは固定次元のベクトル表現に変換される。このプロセスを Embedding と呼ぶ。Embedding により、テキストデータを連続的なベクトル空間にマッピングすることで、高次元の疎性問題を解決するだけでなく、ベクトル空間内でトークン間の意味的関係を表現することができる。モデルによって Embedding の次元は異なり、例えば BERT は 768 次元のベクトルを使用し、GPT-3 は 12288 次元のベクトルを使用する。

2.2.3 Positional Encoding（位置エンコーディング）:

Transformer モデルはシーケンスの順序処理に依存しないため、テキストの順序関係を捉えるために位置情報を導入する必要がある。Positional Encoding は、各トークンに位置ベクトルを追加することでこれを実現する。これらの処理ステップを通じて、Transformer モデルは元のテキストデータを深層学習モデルの処理に適した高次元ベクトル表現に変換し、後続の意味理解とタスク実行の基礎を築く。

2.3 Transformer の構造:

Transformer は古典的なエンコーダ・デコーダ（Encoder-Decoder）アーキテクチャを採用しており、Seq2Seq モデルの中核フレームワークを構成し、入力シーケンスに基づいてターゲットシーケンスを動的に生成することができる。



2.3.1 Embedding 層と Positional Encoding：

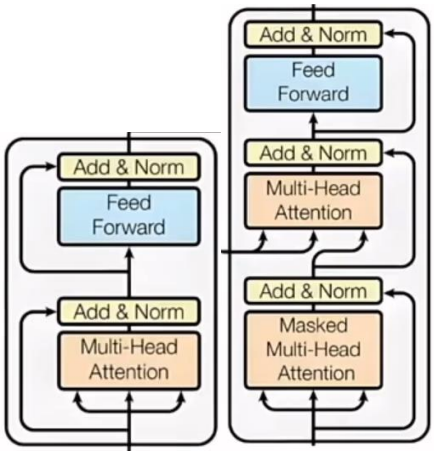
データ入力はまず Embedding 層を通過し、各単語を固定次元の意味ベクトルに変換する。その後、Positional Encoding を通じて位置情報を加え、位置ベクトルと意味ベクトルを加算することで、モデルが単語間の相対的な位置関係を捉えられるようにする。位置エンコーディングは正弦/余弦関数を用いて計算され、モデルにシーケンスの順序を理解させる。

$$\mathrm{PE}_{\mathrm{pos},2\mathrm{i}} = \sin(\mathrm{pos}/10000^{2\mathrm{i}/\mathrm{d}})$$
$$\mathrm{PE}_{\mathrm{pos},2\mathrm{i}+1} = \cos(\mathrm{pos}/10000^{2\mathrm{i}/\mathrm{d}})$$

2.3.2 Encoder：

各エンコーダ層は 2 つのモジュールを含む：Multi-Head Attention モジ

ュールは自己注意機構を実現し、多角的に語彙間の意味関係を観察する。Feed-Forward（フィードフォワード）はネットワークの特徴をさらに抽出する。



2.3.3 Decoder:

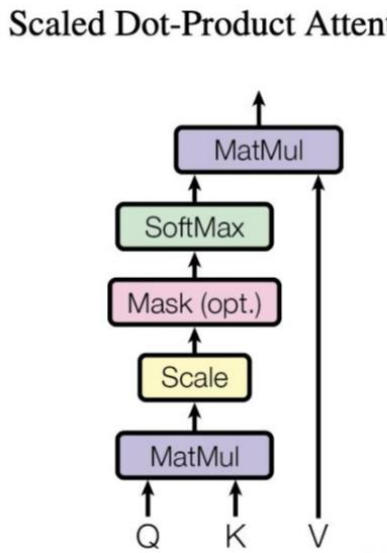
デコーダの各層は 3 つのモジュールを含む: Masked Multi-Head Attention はエンコーダの Multi-Head Attention モジュールに似ているが、後続情報の遮蔽（マスキング）を実現しており、モデルが既存の情報のみに基づいて処理・出力するようにする。Multi-Head Attention はエンコーダ処理後のデータとデコーダのデータを結合する。Feed-Forward はネットワークの特徴をさらに抽出する。最終的に Linear 層と Softmax 関数によってターゲット単語の確率分布を生成する。

2.3.4 Attention 機構:

Transformer の中核は Attention 機構である。Attention 機構は、人間が言語を理解する際に、無関係な部分を無視して特定の重要な情報に意識的に注意を集中することを模倣している。モデルはこの考え方を利用し、入力シーケンスを処理する際、現在のタスクに応じてどの単語に注目すべきかを自動的に

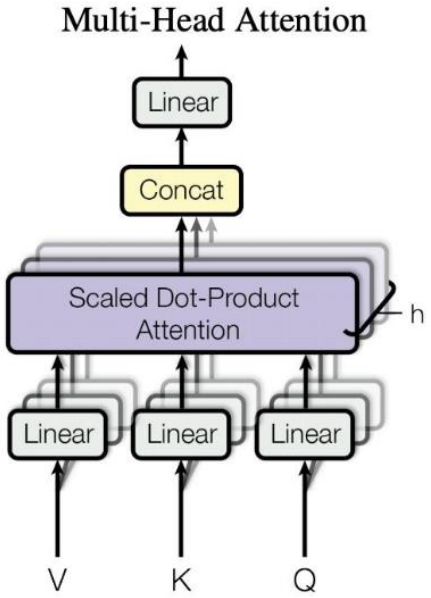
判断する。注意機構は、入力のクエリベクトル（Query）とそれに対応するキーベクトル（Key）との類似度を計算してスコアリングを行い、そのスコアの高低に基づいてバリュー（Value）情報に異なる重みを付与し、加重和を求めて最終的な出力を得る。

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$



2.3.5 Multi-Head Attention 機構:

これは Attention 機構を改良したもので、複数の並列するアテンションヘッド（Attention Head）を導入している。並列処理と複数のアテンションヘッドの結果を結合することを通じて、それぞれ異なる角度からデータの特徴を捉え、モデルの理解能力と表現力をさらに強化している。



3. 実験設計とケース分析:

3.1 データの準備:

JParaCrawl からダウンロードしたパラレルデータセットを使用した。このデータセットは日本電信電話株式会社（NTT）によって作成され、「NTT によって作成された最大規模の公開中日パラレルコーパス」と説明されている。また、提供されている SentencePiece 分かち書きモデルを使用した。

3.2 モデル構造:

3.2.1 Seq2SeqTransformer:

PyTorch に組み込まれている方法を使用し、エンコーダとデコーダを持つモデルを構築する。モデルの出力を語彙リストにマッピングし、結果の確率分布を生成するために使用し、さらに順伝播関数を定義する。

3.2.2 TokenEmbedding:

埋め込み次元を定義し、入力された単語をベクトルに変換する。

3.2.3 PositionalEncoding:

計算式を定義し、単語と単語の間の位置関係を計算し、各単語ベクトルに位置情報を加える。

3.2.4 Generate_square_subsequent_mask:

デコーダ内のターゲットマスクを生成し、モデル学習時に未来のデータに注目しないようにする。

3.3 損失関数と最適化アルゴリズム:

torch.nn.CrossEntropyLoss（交差エントロピー損失関数）と、torch.optim.Adam（適応モーメント推定オプティマイザ）を使用した。

3.4 ハイパーパラメータの設定:

モデルフレームワーク：Transformer 構造
エンコーダ、デコーダ層数：3
単語埋め込み次元：512
マルチヘッドアテンションのヘッド数：8
フィードフォワードニューラルネットワークの隠れ層次元：512
学習バッチサイズ、総エポック数：16
学習率：0.0001
Adam モメンタムパラメータ：(0.9, 0.98)

3.5 評価指標:

モデルが生成した文と実際の文の差異を計算するために、BLEU スコアを使用した。

3.6 実験結果:

わずか 1 時間のデータ学習で BLEU スコアは 0.6 以上に達し、損失値の収束も比較的明白であり（1.7 まで収束）、その効果は悪くないことが見て取れる。

```
translated_sentence = translate(
    transformer,
    "日本のコンビニのサービスには 本当に感心させられる。",
    ja_vocab,
    zh_vocab,
    ja_tokenizer
)

print(translated_sentence)
```

```
_日本 便利店 的 服务 真 的 令 人 佩服
```

```
print("\n开始评估训练好的模型...")
avg_bleu, exact_match_rate = evaluate_trained_model(transformer, test_iter)
```

```
开始评估训练好的模型...

模型性能评估:
测试句子数: 100
平均BLEU分数: 0.6414
```

3.7 モデル比較:

同時に、従来の RNN モデルも構築して学習を行ったところ、学習時の損失値の収束速度が非常に遅く（5.6 までの収束）、3 時間学習させてもその効果は Transformer モデルには遠く及ばないことが明らかになった（BLEU スコアはわずか 0.05）。一方、Transformer モデルはわずか 1 時間で BLEU スコア 0.6 以上に達した。

3.8 適用シーンと限界:

3.8.1 適用シーン:

機械翻訳における広範な応用に加え、Transformer アーキテクチャは多様なタスクにおいても優れた性能を示している。

コンピュータビジョン分野： 畳み込みニューラルネットワークが長らくこの分野を支配していたが、Transformer はグローバル情報を処理する能力を備えているため、Vision Transformer に代表されるように、徐々にその潜在能力を示している。

時系列予測タスク： あるデータの発展過程を観測し、一定の法則に従っ

て未来のデータを予測するタスクにおいて、勾配消失や勾配爆発のため従来の構造では長距離予測が困難であったが、Transformer の自己注意機構はこの課題に有効に対処できる。

マルチモーダル学習： マルチモーダル手法は、テキスト、画像、音声などの異なる種類の情報源を融合することに注力しており、近年クロスモーダルタスクが広範な注目を集めている。CLIP はこの方向性の重要な成果の一つである。

3.8.2 存在する限界:

モデルが巨大で複雑： Transformer は大量のパラメータを含んでおり、モデル学習時の計算コストが膨大になる。そのため、モデルの軽量化と計算効率の向上は研究の重点となり得る。

膨大なコーパスデータを必要とする： 特定のアプリケーションシナリオであればあるほど、モデルの学習には豊富なデータサポートが必要となり、そうでない場合は過学習（Overfitting）現象が発生しやすい。

解釈可能性（Explainability）の欠如： Transformer の構造は複雑で内部メカニズムの解読が難しく、従来の線形モデルやツリーモデルと比較して、その「ブラックボックス」的な特徴が一部の応用において受容を制限している。

ハイパーパラメータに敏感： アーキテクチャが精緻であるため、モデルはパラメータや構造の調整に対して敏感であり、最適な構成案を得るためには通常、大量の試行が必要となる。

4. 今後の実験探求の方向性

\

4.1 アーキテクチャに関する最適化と改善:

将来の研究者は、パラメータ数を拡張し学習リソースを増やすことでモデル性能をさらに向上させる、より大規模な事前学習モデルを Transformer アーキテクチャを中心に構築し続けることができる。それと同時に、既存のアーキテクチャに対して構造的な最適化を行い、例えば簡素化された Decoder-only アーキテクチャを採用するなどして、モデルをより軽量かつ高効率で、コスト

パフォーマンスに優れたものにすることもできる。あるいは、Transformer 構造から脱却し、全く新しい、より潜在能力のあるニューラルネットワークアーキテクチャを探索し、根本的にモデル能力の向上を推進することも可能である。

4.2 新技術の導入:

研究が進むにつれて、Transformer 体系の中核的な手法（自己注意機構など）を改善するための新興技術を開発できる。例えば、RetNet はマルチスケール保持（Multi-Scale Retention）メカニズムを導入して Transformer 体系のマルチヘッドアテンション機構を置き換えた。これは本質的に RNN と Transformer を結合したものであり、それによって学習プロセスの並列化、推論効率の向上、さらにはモデル性能の向上を実現している。

4.3 翻訳スタイルの最適化:

口語表現は常に機械翻訳における大きな課題である。特にインターネットの文脈では、ネット用語や日常的な非公式表現が多種多様であり、従来の翻訳モデルでは自然で人間に近い表現の翻訳を生成することが困難になっている。したがって、現代の人間の自然言語に近いコーパスをいかに効果的に収集・構築し、モデルの生成戦略を最適化して「機械っぽさ」を低減させるかは、将来深く研究すべき重要な方向性である。

4.4 より多くの領域への応用拡大:

Transformer アーキテクチャの提案は自然言語処理分野で卓越しているだけでなく、今日ではそのモデリング思想が調整・最適化を経て、コンピュータビジョン、音声処理、時系列予測など多くの分野ですでに適用されており、同時に他分野のモデル設計にも重要なアイデアを提供できる。

5. まとめ:

本研究は、中日間の機械翻訳ニーズを巡って、完全な Transformer アーキテクチャに基づいた日中翻訳モデルを設計・実装した。マルチヘッド自己注意機構とエンコーダ・デコーダ構造を結合させたものである。実験結果は、Transformer モデルが機械翻訳タスクにおいて顕著な優位性を持つことを示している。今後、モデルアーキテクチャのさらなる最適化、革新、および分野を超えた応用に伴い、Transformer に基づく機械翻訳システムおよびその他のシステムは、より多くの実際のシーンでその役割を発揮し、多言語機械翻訳と人工知能の発展に寄与するであろう。

参考文献

[1] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio Neural Machine Translation by Jointly Learning to Align and Translate(2014)

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin Attention is All you need (2017)

[3] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, Tie-Yan Liu On Layer Normalization in the Transformer Architecture (2020)

[4] TB Brown，B Mann，N Ryder，M Subbiah，D Amodei Language Models are Few-Shot Learners (2020)