

普遍論理としての Transformer 理解

Understanding of Transformer Technologies as for Universal Logics

岡島 義憲

Yoshinori Okajima

情報統合技術研究合同会社
Info-Integnology Reasearch, LCC.

Abstract: Transformer architectures are regarded as having made great successes in language processing and various content generating domains, also but still being pointed out that it is not good at numerical operations. So, the reason of the weakness is analyzed by investigating into inner neural network operations, and will be discussed what is fundamental to overcome the issues and make the generative AI have more generality including numerical data processing.

1. はじめに：汎用化の源泉

Transformer アーキテクチャが登場した 2017 年当時、ニューラルネットワークを用いた機械翻訳技術の開発に関して、以下のニューラルネットワーク関連技術が注目されていた^[1]。

- (1) Sequence To Sequence 変換^[2]
- (2) Attention 機構^{[3][4]}
- (3) 再帰的構造の排除（脱 RNN）^[1]

Attention 機構は、再帰的構造を除くネットワークトポロジーに道を開き、これにより、Sequence To Sequence 変換回路の大規模化の道が開かれた。Transformer アーキテクチャは、ベクトル列データである Text 文の入力に回答して種々のベクトル列データ（コンテンツ）を生成のすることが可能であることが確認され、その応用範囲は、翻訳、Q&A、要約等の文章作成から、画像、音楽、プログラム等の各種コンテンツ生成へと広がった^[5]。

このように多くの生成分野の対応が可能となったことから、上記(1)から(3)の導入により、「大規模な事前学習や強化学習を行えば、汎用的 AI 能力を獲得できるのかもしれない」との期待も生まれた^[6]。しかし、一方で、数値計算等で必要となる「数の理解（数値情報）」は弱点との指摘もある^[7]。また、単語や文章の意味理解が不足していると思える文章生成を行うことがあるの評価もある^[8]

* 連絡先 E-mail : okajima@info-integnology.com

*1) 本論では、ニューラルネットワークを「回路」とみなして論を進める。

そこで、本論では、上記(1)から(3)の特徴を持つ「素回路」を先ず考え、その「素回路」を用いる形で Transformer アーキテクチャを再構成することで、汎用的能力の源泉の明確化を試みる。

以降、その「素回路」を「T 回路」と呼ぶ。

尚、本論では、ニューラルネットワークの学習処理に関する議論は行わず、推論動作における汎用的能力の源泉のみを議論する。

2. T 回路モデル：

前章(1)は、「T 回路がデジタルコンテンツの入出力を可能とする」との要件である。“Sequence”というのは、時系列に並ぶ言語や図形の情報であり、一般のデジタルコンテンツのデータは、ベクトル列だからである。

そこで、T 回路の入出力モデルを、図 1 にて抽象化する。

このモデルは、X 入力（ベクトル列情報 X）を元に、下部から入力される Y 入力（ベクトル列情報 Y）が変換処理され、上部の Y' 出力（ベクトル列情報 Y'）を出力する。

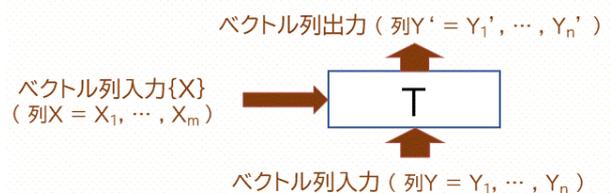


図 1. T 回路の入出力モデル

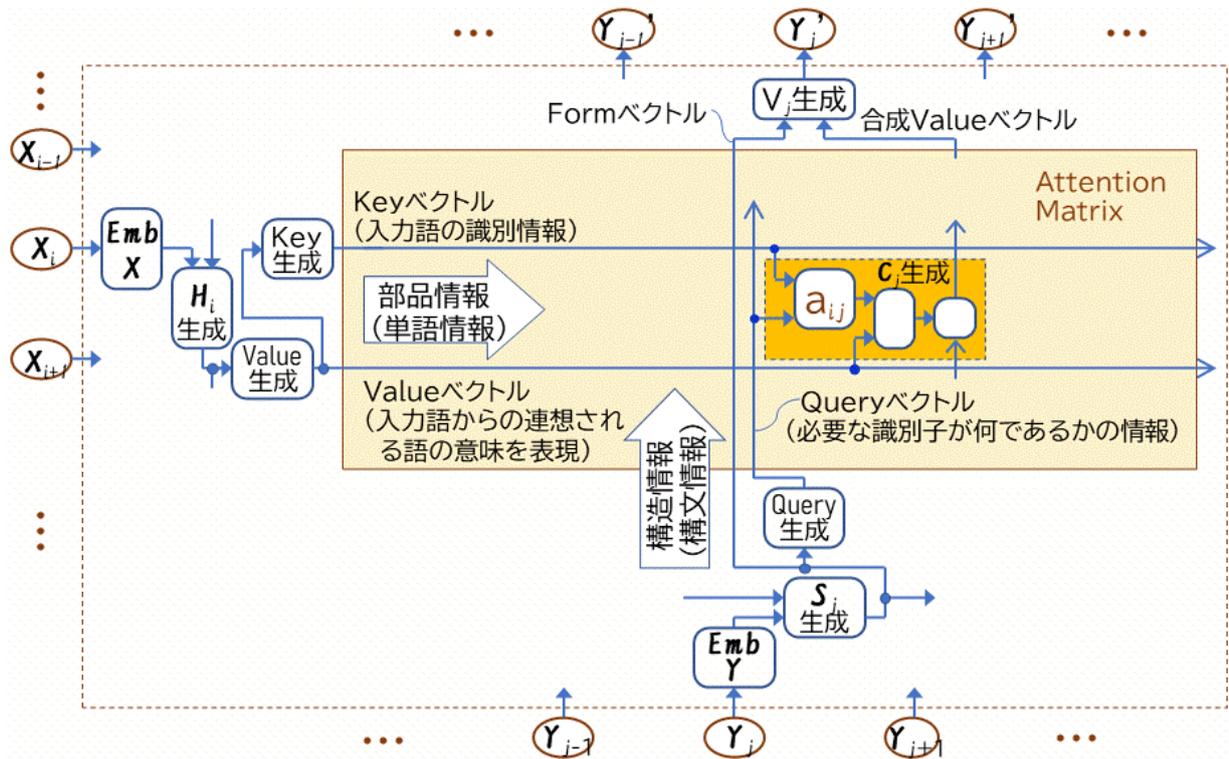


図 2. Attention 機構の例

T回路は、X入力から生成される情報とY入力から生成される情報が作る2次元のAttention-Matrixを内蔵する。Attention-Matrixは、Y側下部から送られるQueryベクトル^{*2}とX側から送られるKeyベクトル^{*2}の内積値を係数としてValueベクトル^{*2}に重みを付けし、X側で予測する「部品情報」であるValueベクトルの総合情報^{*2}をY側入力求める「構文構造情報^{*2}」の中に埋め込むという処理を行う。図中には、m個のX側ベクトルの内の1個($X_i; i=1, \dots, m$)と、n個のY側ベクトルの内の1対(Y_j と $Y'_j; j=1, \dots, n$)と、その交点に関する回路のみを記載している。

尚、この図のAttention機構は、Multi-Attentionとはなっていない。

文章生成アプリでは、X入力はユーザーの入力文章に関する情報であり、Y入力はユーザーの入力文章に対する応答文章(案)である。Y出力では、Y入力の応答文章(案)は更新されブラッシュアップされる。

Attention機構を持つことにより、T回路は、内部に再帰的構造を持たない。その内部回路を、文章生成や機械翻訳用に発表されているTransformer^[1,9,10]から推測検討すると、その構造は図2のような構成を取ると考える。

T回路内部の動作は以下である。

- i 番目のX側入力ベクトル(X_i)は、 $(i-1)$ 番目までのベクトル情報を踏まえて、「Y側出力のいずれかの箇所に現われるべき単語に関する確率情報」をニューラル回路(H_i)で生成する。そして、その確率的な情報から、2種類のベクトル、Value(i)とKey(i)を抽出する。Value(i)はその単語の意味を確率的に表

現し、Key(i)はその単語の識別情報を表現する。

- j 番目のY側入力ベクトル(Y_j)は、 $(j-1)$ 番目までのベクトル情報を踏まえて、「Y側出力文章が備えるべき構文形式に関する確率情報」をニューラル回路(S_j)で生成する。そして、その確率的な情報から、2種類のベクトルQuery(j)とForm(j)を抽出する。Query(j)は、Y側出力文がその位置にて必要とする語の属性情報を確率的に表現し、Form(j)は、その構文情報を確率的に表現する。

- Attention機構は、 X_i 側が予測する単語^{*2}の Y_j 側配置位置^{*2}をKey(i)ベクトルとQuery(j)ベクトルの内積値 $\{\sum a_{ij}\}$ によって割り出し、その単語^{*2}をY側入力設定した構文の中に埋め込む^{*3}。更に、

*2 これらのベクトルは、確率値を成分とするベクトルである。

・最終出力段 (Y'側回路) にて、 V_j 生成 (出力単語ベクトルの生成) が行われ、その列の Y'_j ベクトルとして出力する。(図2)。

Attention 機能は、主に、学習時に獲得した H_i モジュール ($i=1, \dots, m$) と S_j モジュール ($j=1, \dots, n$) の生成機能に左右される。

つまり、T 回路は、学習時に獲得した Attention 機能を元に、X 側入力情報から Y 側出力コンテンツに現われるべき単語別の生成確率値を求め、Y 側入力が必要なデータ構造と照らし合わせて、出力データの時系列構造の最適箇所に最適単語を配置している。

V_j モジュール ($j=1, \dots, n$) も学習時に機能を獲得し、Attention 機構から送られて来た情報と S_j モジュールの出力を元に、出力する単語*2を予測する。

尚、Y 側下方の入力 ($Y_j; j=1, \dots, n$) は構文情報であったのに対し、Y 側上方への出力 ($Y'_j; j=1, \dots, n$) は単語情報列である点に注意する必要がある。

次章で必要となる T 回路の縦積み時には、Y 側下方の入力情報タイプと Y 側上方への出力情報タイプを揃える必要があるので、縦積みの T 回路の間には、単語情報列から構文情報を抽出する回路を挟めなくてはならない。そのためには、Self-Attention 機能を有する T 回路を使うのが適切であろう。

3. T 回路を用いた生成回路の構成

生成回路がユーザーからの入力文章を元に所望のコンテンツを生成するには、先ず、生成回路はユーザーがどのようなタスクを求めているかを理解しなくてはならない。そのタスク情報は、ユーザーの入力文章を解析すれば知ることができる筈である。

タスクとしては、以下のような種類が考えられる。

- ・ 質疑応答 (Question-Answering)
- ・ 翻訳 (Sentence Translation)
- ・ 読解 (Reading Comprehension)
- ・ 要約 (Summarization)

タスクを理解すると、生成回路の出力となる文章が備えるべき「文章スタイル*5」を、学習によって獲得した「複数の構文形式」の各々に付与する確率値として表現することが可能となる。

「複数の構文形式」に対して確率が設定されるので、Y 側文章の生成プロセスは基本的には並列的かつ複合的に進められるべきである。最終的に出力する文章は、並列して進めた複数の処理を合成するか、選択する。そのような並列的かつ複合的な処理を進めるには、例えば、図3のように、並列に処理を進めつつ、情報融合も可能な構造が必要である。

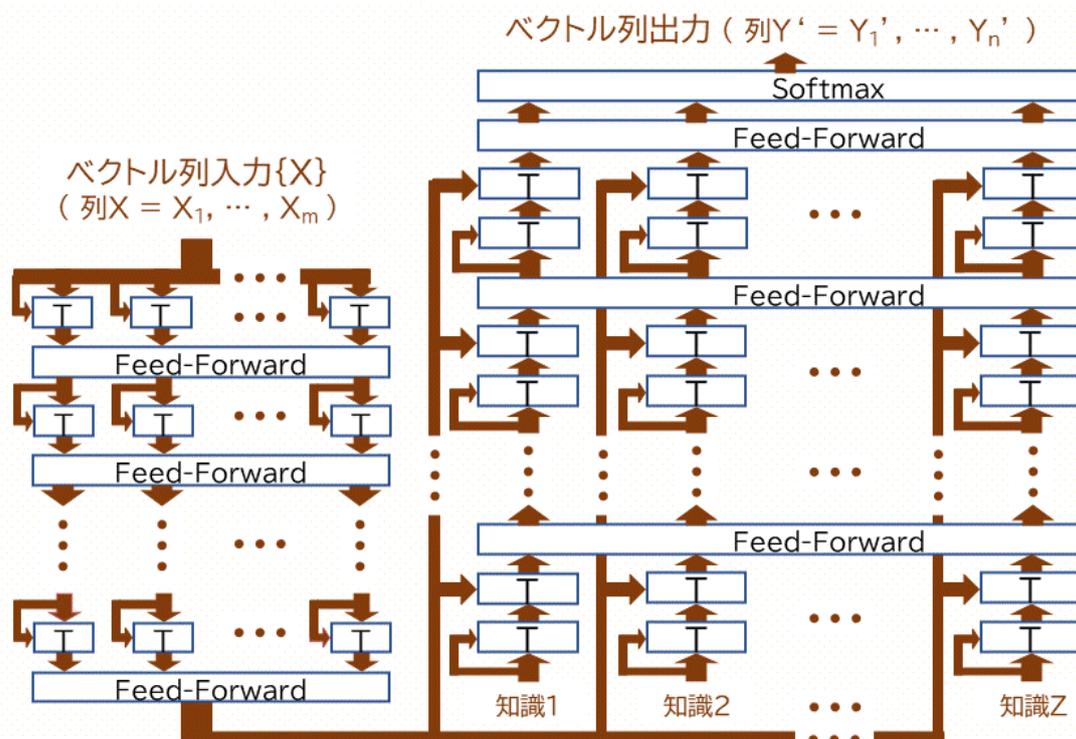


図3. 生成 AI の別案図 (T 回路をマトリックス展開)

4. 考察

T回路の目指す生成動作は、「学習により獲得した Attention 機能を元に、X 側入力情報から Y 側出力コンテンツに現われるべき単語別の生成確率値を求め、Y 側入力が求めるコンテンツ構造と照らし合わせて、出力コンテンツの時系列構造の最適箇所に最適単語を配置する」ことであった*³。

コンテンツ構築に用いる単語の識別番号である「コード」は、デジタルコンテンツ生成にて普遍的に現れる Text や Audio 以外のコンピュータプログラムで用いる Instruction コードや Computer Graphics の静止画や動画を構成するコンテンツコードでもよく、また、「下位階層の部品コードを求め、そのコードを想定する構造の最適箇所に配置する」というプロセスは、多くのデジタルコンテンツ生成にて現れ得るプロセスである。

そのことからすると、「T回路は、デジタルコンテンツ生成に関するダイナミズムを表現する上での普遍的機能を提供し得る」といえる*³。

但し、1章で触れたように、Transformer アーキテクチャや Large Language Model を用いた生成 AI 回路には、少なくとも以下2点で大きな弱点が指摘されている。

- (1) 単語の持つ意味への理解が不足と思える文章生成を行うことがある^[8]。及び、
- (2) Attention 機能は数値の持つ本質的な意味を理解した生成動作を行うことができない^[7]。との点である*⁴。

ここで、1点目の問題は、素論理の機能が担うべき課題ではなく、システム全体のアーキテクチャやシステムの学習戦略の課題とみなすべきだろう^[8]。Real 世界とのかかわりを持たずに、デジタル情報をバッチ処理するだけの学習プロセスでは、言語と背景の関係を学習する機会が余りに乏しい。

しかし、(2)点目の問題は、推論動作における本質的な弱点であり、素論理としての T 回路の欠点と見做すべきだと思われる。

そもそも、汎用的生成 AI を求めるのであれば、人間の活動を含む自然界のダイナミズムをエミュレーションする時に必要な

- (a) 数値に関する論理

*³ 「T回路の目指す生成動作」とは、学習後の T 回路の目標機能のことを言っている。

- (b) 情報に関する論理
の2種類が必要とされる筈である^[11]。

T回路(Attention 機能)は、後者の「情報に関する論理」のエミュレーションを担うが、T回路には数値演算機能は実装されていない。

数値演算能力を持つ生成 AI を求めるのであれば、デジタルコンピュータを内蔵すべきなのである。

即ち、この問題は、生成 AI システムの中にデジタルコンピュータ・サブシステムをどのように接続させるのかという問題であり、生成 AI システムがどのようにそのコンピュータプログラムを自律的に生成し、どのようにしてそのプログラムを走らせるのかという問題であると思われる。

参考文献

- [1] A. Vaswani, et al.: "Attention Is All You Need",
- [2] I. Sutskever, et al.; "Sequence to Sequence Learning with Neural Networks." Advances in neural information processing systems 27, 2014.
- [3] Dzmitry Bahdanau, et al.: "Neural machine translation by jointly learning to align and translate." 3rd International Conference on Learning Representations, ICLR 2015.
- [4] Minh-Thang Luong, et al.: "Effective approaches to attention based neural machine translation." Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015.
- [5] Thompson, A.D.: "Integrated AI: The sky is infinite (2022 AI retrospective)", 2022.
- [6] Ganguli, Deep, et al.: "Predictability and surprise in large generative models." Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency. 2022.
- [7] Hongwei Han, et al. "LUNA: Language Understanding with Number Augmentations on Transformers via Number Plugins and Pre-training" Computer Science, 2022.
- [8] Maren Pielka, et al.: "A Linguistic Investigation of Machine Learning based Contradiction Detection Models: An Empirical Analysis and Future Perspectives." ICMLA 2022.
- [9] Britz, Denny, et al.: "Massive Exploration of Neural Machine Translation Architectures." Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017.
- [10] Tom B. Brown, et al. (2020) "Language Models are Few-Shot Learners" Advances in neural information processing systems 33, pp.1877-1901, 2020.

[11] 清水義夫著；「圏論による論理学（高階論理とトポス）」、東京大学出版会、平文社、2007年