AI 特許紹介(42)

AI 特許を学ぶ！究める！
～Vision Transformer 特許～

2022 年 7 月 8 日

河野特許事務所
所長弁理士　河野英仁

「AI 特許紹介」シリーズは、注目すべき AI 特許のポイントを紹介します。熾烈な競争となっている第４次産業革命下では AI 技術がキーとなり、この AI 技術・ソリューションを特許として適切に権利化しておくことが重要であることは言うまでもありません。

AI 技術は Google, Microsoft, Amazon を始めとした IT プラットフォーマ、研究機関及び大学から毎週のように新たな手法が提案されており、また AI 技術を活用した新たなソリューションも次々とリリースされています。

本稿では米国先進 IT 企業を中心に、これらの企業から出願された AI 特許に記載された AI テクノロジー・ソリューションのポイントをわかりやすく解説致します。

1.概要

特許出願人　Google
出願日　2021 年 10 月 1 日
公開日　2022 年 4 月 7 日
公開番号 US2022/0108478
発明の名称　セルフアテンションベースのニューラルネットワークを使用した画像の処理

2017 年 12 月 Vaswani により画期的な論文「Attention Is All You Need」が発表され、Transformer として言語処理 AI の革新的進歩につながった。478 特許は、画像処理の機械学習アルゴリズムに CNN ではなく、Transformer を用いた Vision Transformer 特許に関する。

2.特許内容の説明

図 1 は Vision Transformer のネットワーク構成図である。

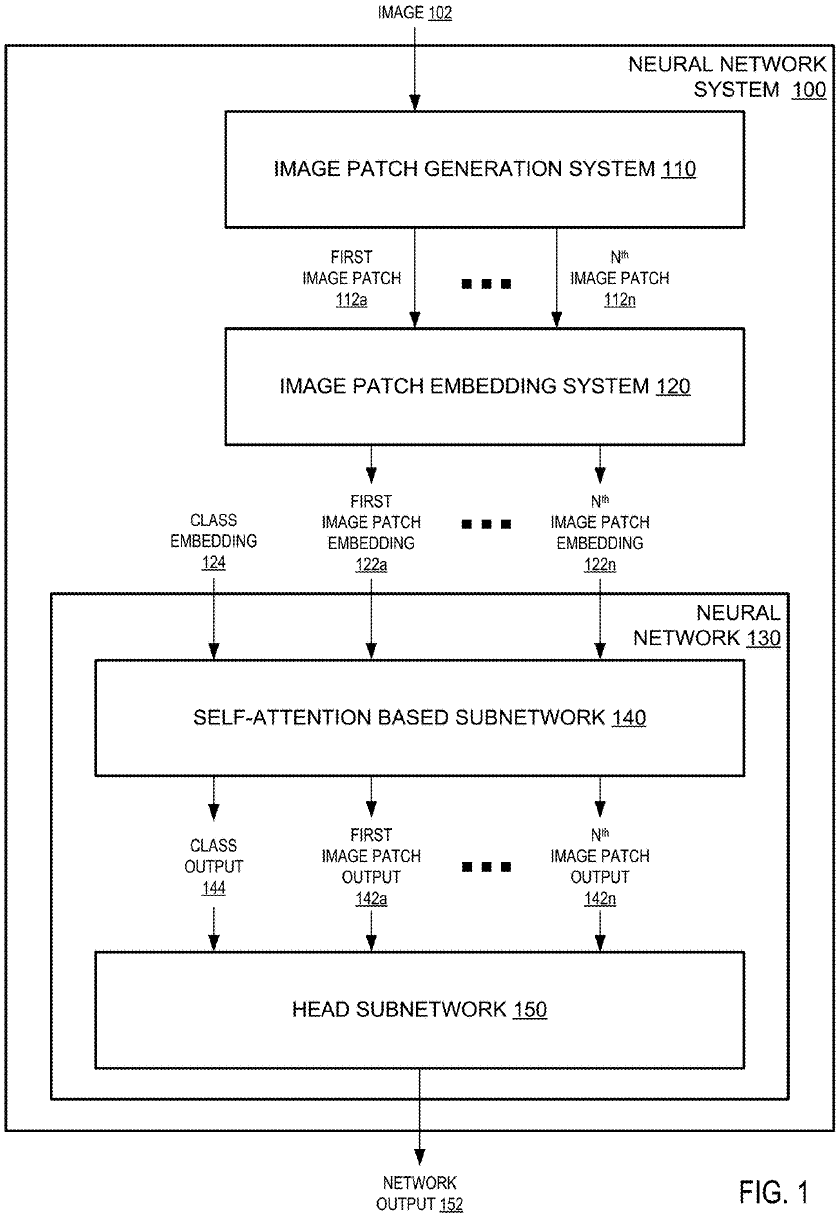


FIG. 1

ニューラルネットワークシステム 100 は、画像 102 を処理し、画像に関する予測を表すネットワーク出力 152 を生成する。ニューラルネットワークシステム 100 は、画像 102 を使用して任意の適切な機械学習タスクを実行する。

ニューラルネットワークシステム 100 は、画像パッチ生成システム 110、画像パッチ埋め込みシステム 120、およびニューラルネットワーク 130 を含む。ニューラルネットワーク 130 は、セルフアテンションベースのサブネットワークを含むセルフアテンションベースのニューラルネットワークである。

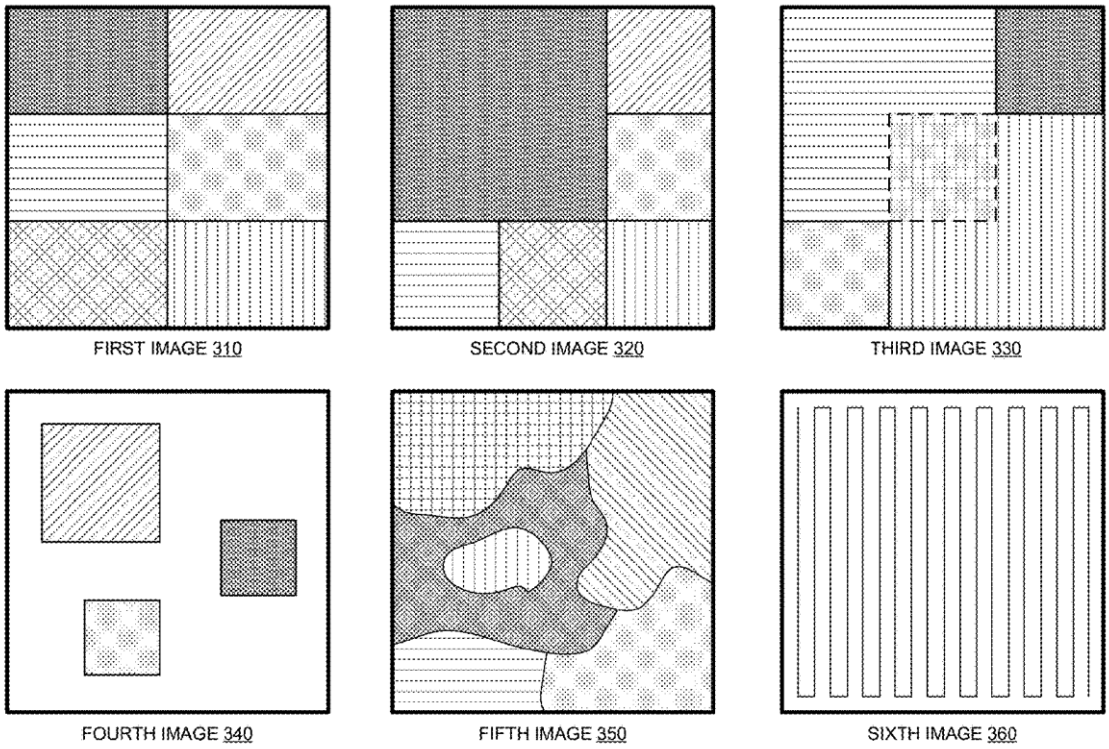
セルフアテンションベースのニューラルネットワークは、1 つまたは複数のセルフア

テンションニューラルネットワーク層を含むニューラルネットワークである。セルフアテンションニューラルネットワーク層は、入力として層入力要素のシーケンスを受け取り、層入力要素のシーケンスにアテンションメカニズムを適用して、層出力要素のシーケンスを生成する。特に、各層入力要素について、セルフアテンションニューラルネットワーク層は、層入力要素から導出された1つまたは複数のクエリを使用して、それぞれの出力要素を生成するために、層入力要素のシーケンス内の層入力要素にアテンションメカニズムを適用する。

ニューラルネットワーク 130 は、セルフアテンションベースのサブネットワーク 140 を使用して、画像 102 のそれぞれのパッチを表す入力要素を含む入力シーケンスを処理する。このように、ニューラルネットワーク 130 は、画像 102 内の異なる位置にある異なるパッチにアテンションを向けるために、入力シーケンスにアテンションメカニズムを適用する。画像 102 のパッチは、並列処理を使用してセルフアテンションベースのサブネットワーク 140 によって処理される。

画像パッチ生成システム 110 は、画像 102 を処理し、画像 102 のｎ個の異なるパッチ 112a～n を生成する。各画像パッチ 112a～n は、画像 102 の複数の連続するピクセルを含む。すなわち、特定の画像パッチ 112a～n ごと、および特定の画像パッ 112a～n 内の任意のピクセルのペアについて、ペアの第1のピクセルからペアの第2のピクセルへのパスが存在し、パスは、 特定の画像パッチ 112a～n におけるピクセルのみを含む。

画像パッチ 112a～n は、任意の適切な方法で表すことができる。 例えば、各画像パッチ 112a～n は、画像パッチ 112a～n のピクセルを含む2次元画像、例えば、画像パッチ 112a～n 内のピクセルの空間的関係を維持する画像として表すことができる。
画像パッチの詳細例を図 3 に示す。



画像パッチ埋め込みシステム 120 は、画像 10 のｎ個の画像パッチ 112a～n を取得し、次に画像パッチ 112a～n のそれぞれの埋め込み 122a～n を生成する。各画像埋め込み 122a～n は、対応する画像パッチ 112a～n のピクセルを表し、対応する画像パッチ 112a～n のピクセルを処理することによって生成される。

各画像パッチ 112a～n が画像 102 の 2 次元サブ画像として表される形態では、122a～n を埋め込む各画像パッチは、対応する画像パッチ 112a～n の再形成(reshaped)されたバージョンである。例えば、画像パッチ埋め込みシステム 120 は、各画像パッチ 112a～n を「平坦化」して、画像パッチ 112a～n 内の各ピクセルを含む一次元テンソルである画像パッチ埋め込み 122a～n を生成する。特定の例として、各画像パッチ 112a～n が次元数 L×W×C を有する場合、ここで、Ｃは画像のチャネル数を表す（例えば、RGB 画像の場合、C=3)、画像パッチ埋め込み 122a～n は、寸法 1×(L·W·C)を有する画像パッチ埋め込み 122a～n を生成する。

画像パッチ埋め込み 122a～n は、特定のフォーマット、例えば、特定のサイズおよび形状を有する入力を受け入れるようにトレーニングを通じて構成されたニューラルネットワーク 130 によって処理される。したがって、画像パッチ埋め込みシステム 120 は、各画像パッチ 112a～n を、ニューラルネットワーク 130 によって必要とされる次元を有する座標空間に投影することができる。

例えば、画像パッチ埋め込みシステム 120 は、線形投影を使用して各画像パッチ 112a〜n を処理する。

$$z_i = x_i E_i + b_i$$

ここで、$z_i \in \mathbb{R}^D$ は i 番目の画像パッチ埋め込み 122a〜n、D はニューラルネットワーク 130 に必要な入力次元、$x_i \in \mathbb{R}^N$ は i 番目の画像パッチ 112 a〜n を含む 1 次元テンソル、Nis は i 番目の画像パッチ 112a〜n、$E_i \in R^{N \times D}$ は射影行列、$b_i \in I' \mathbb{R}^D$ は線形バイアス項である。

画像パッチ埋め込みシステム 120 が画像パッチ埋め込み 122a〜n を生成した後、ニューラルネットワークシステム 100 は、画像パッチ埋め込み 122a〜n からニューラルネットワーク 130 への入力として提供される入力シーケンスを生成する。

画像パッチ埋め込み 122a〜n に対応する入力シーケンスの入力要素を生成するために、ニューラルネットワークシステム 100 は、(i)画像パッチ埋め込み 122a〜n と、(ii)画像パッチ埋め込み 122a〜n に対応する画像パッチ 112a〜n の画像 102 内の位置を表す位置埋め込み、を組み合わせる。例えば、ニューラルネットワークシステム 100 は、位置埋め込みを画像パッチ埋め込み 122a〜n に追加する。位置埋め込みを組み込むことによって、ニューラルネットワークシステム 100 は、空間情報、例えば、画像内の各画像パッチの相対的位置を符号化することができ、ニューラルネットワーク 130 によって利用され、ネットワーク出力 152 を生成することができる。

画像 102 の各画像パッチ 112a〜n に対応する位置埋め込みは整数である。例えば、画像 102 の左上にある第 1 の画像パッチは、「1」の位置埋め込みを有し、第 1 の画像パッチのすぐ右側にある第 2 の画像パッチは、「2」の位置埋め込みを有する。また位置の埋め込みは機械学習することもできる。例えば、ニューラルネットワーク 130 のトレーニング中に、トレーニングシステムは、ニューラルネットワーク 130 のトレーニングエラーを、ニューラルネットワーク 130 を介して位置埋め込みに逆伝播することによって、位置埋め込みを同時に学習することができる。

入力シーケンスを生成した後、ニューラルネットワークシステム 130 は、入力シーケンスをニューラルネットワーク 130 への入力として提供する。ニューラルネットワーク 130 は、入力シーケンスを処理して、ネットワーク出力 152 を生成する。ニューラルネットワーク 130 は、セルフアテンションベースのサブネットワーク 140 を使用して入力シーケンスを処理して、出力シーケンスを生成する。

セルフアテンションベースのサブネットワーク 140 は、それぞれが層入力シーケンスを受け取り、層入力シーケンスにセルフアテンションメカニズムを適用して層出力シーケンスを生成する 1 つまたは複数のセルフアテンションニューラルネットワーク層を含む。セルフアテンションベースのサブネットワーク 140 が出力シーケンスを生成した後、ニューラルネットワーク 130 は、出力シーケンスの 1 つまたは複数の要素をヘッドサブネットワーク 150 に提供する。

ヘッドサブネットワーク 150 は、 n 個の画像パッチ出力 142a～n を処理する。ヘッドサブネットワーク 150 は、画像パッチ出力 142a～n を組み合わせて（例えば、グローバル平均プーリングを使用して）組み合わせパッチ出力を生成し、次に組み合わせパッチ出力を処理してネットワーク出力 152 を生成する。例えば、ヘッドサブネットワーク 150 は、1 つまたは複数のフィードフォワードニューラルネットワーク層および／または線形分類器を使用して、組み合わされたパッチ出力を処理する。

別の例として、ヘッドサブネットワーク 150 は、クラス出力 144 のみを処理してネットワーク出力 152 を生成するように構成することができる。すなわち、クラス出力 144 は、画像 102 の最終表現を表すことができ、ヘッドサブネットワーク 150 は、クラス出力 144 を処理して、画像 102 についての予測を表すネットワーク出力 152 を生成することができる。例えば、ヘッドサブネットワーク 150 は、1 つまたは複数のフィードフォワードニューラルネットワーク層を備えた多層パーセプトロンを含むことができる。

セルフアテンションベースのサブネットワーク 140 およびヘッドサブネットワーク 150 は、単一の機械学習タスクで、エンドツーエンドで同時に訓練される。たとえば、トレーニングシステムは、トレーニング入力シーケンス（それぞれのトレーニング画像を表す）と、対応するグラウンドトゥルースネットワーク出力、つまり トレーニング入力シーケンスの処理に応答してニューラルネットワーク 130 が生成する必要のあるネットワーク出力 152 とを含む複数のトレーニングサンプルを有するトレーニングデータセットを使用して教師ありトレーニングプロセスを実行する。

トレーニングシステムは、ニューラルネットワーク 130 を使用してトレーニング入力シーケンスを処理して、それぞれの予測ネットワーク出力を生成し、(i)予測されたネットワーク出力と、(ii)対応するグラウンドトゥルースネットワーク出力との間のエラーに従って、ヘッドサブネットワーク 150 およびセルフアテンションベースのサブネットワーク 140 へのパラメータ更新を決定する。例えば、トレーニングシステムは、ヘッドサブネットワーク 150 およびセルフアテンションベースのサブネットワーク 140

の両方を介してエラーを逆伝播し、確率的勾配降下法を実行することによって、パラメータの更新を決定する。

下記図はセルフアテンションベースのニューラルネットワーク 200 を示す図である。

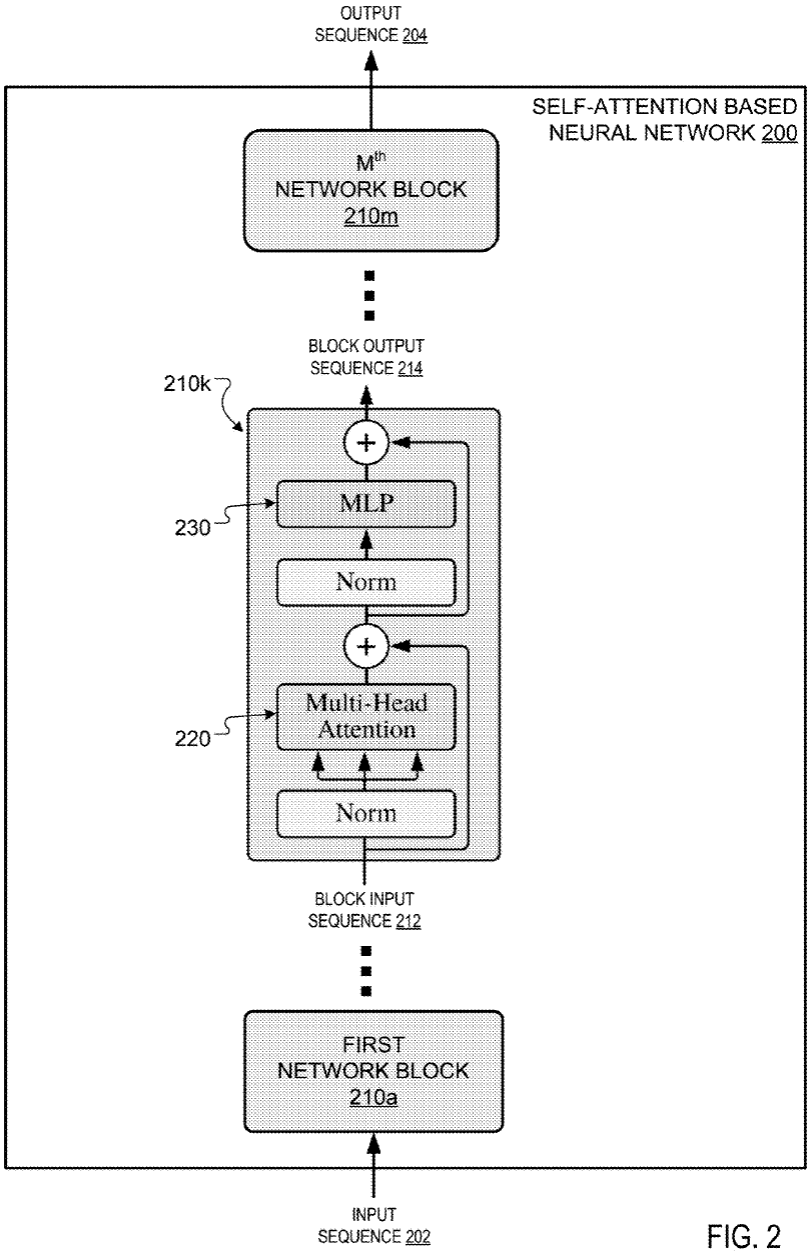


FIG. 2

セルフアテンションベースのニューラルネットワーク 200 は、画像を処理し、画像についての予測を生成するように構成されたニューラルネットワークシステム、例えば、図 1 のニューラルネットワークシステム 100 の構成要素である。特に、セルフアテンションベースのニューラルネットワークは、画像に関する予測を示すネットワーク出力を

生成するように構成されたニューラルネットワーク、例えば、図1のニューラルネットワーク 130 のサブネットワークである。

セルフアテンションベースのニューラルネットワーク 200 は、画像を表し、複数の入力位置のそれぞれにそれぞれの入力要素を含む入力シーケンス 202 を処理する。例えば、入力シーケンス 202 は、画像の複数の画像パッチのそれぞれを表す各入力要素を含む。セルフアテンションベースのニューラルネットワーク 200 は、入力シーケンス 202 を処理して、入力シーケンス 202 と同じ長さを有する、すなわち、入力シーケンス 202 内の入力要素と同じ数の出力要素を有する出力シーケンス 204 を生成する。

セルフアテンションベースのニューラルネットワーク 200 は、M個のネットワークブロック 210a～m、M≧1 のシーケンスを含む。各ネットワークブロック 210a～m は、入力シーケンス 202 内の各入力位置のそれぞれのブロック入力要素を含むブロック入力シーケンスを受信する。

すなわち、各ブロック入力要素は、入力シーケンス 202 のそれぞれの入力要素に対応する。各ネットワークブロック 210a～m は、ブロック入力シーケンスを処理し、入力シーケンス内の複数の入力位置のそれぞれについてそれぞれのブロック出力要素を含むブロック出力シーケンスを生成する。各ブロック入力シーケンス 212 は、入力シーケンスがニューラルネットワーク 200 によって処理されるときに、入力シーケンス 202 内の要素の数を保持する。

シーケンス内の第1のネットワークブロック 210a は、入力シーケンス 202 を受信する。シーケンス内の後続の各ネットワークブロック 210a～m は、ブロック入力シーケンスとして、シーケンス内の先行するネットワークブロック 210a～m によって生成されたそれぞれのブロック出力シーケンスを受信する。M番目および最終ネットワークブロック 210m のブロック出力シーケンスは、出力シーケンス 204 である。

各ネットワークブロック 210a～m は、1つまたは複数のセルフアテンションニューラルネットワーク層を含む。第k番目のネットワークブロック 210k を参照すると、ネットワークブロック 210k は、単一のセルフアテンションニューラルネットワーク層 220 を含む。セルフアテンションニューラルネットワーク層 220 は、ブロック入力シーケンス 212 内のそれぞれのブロック入力要素を取得し、ブロック入力要素にアテンションメカニズムを適用する。

また、ネットワークブロック 210k は、層正規化層の出力をセルフアテンションニュ

ーラルネットワーク層 220 に提供する前に、最初に層正規化層をブロック入力シーケンス 212 に適用する。

各特定の入力位置に対応するそれぞれのブロック入力要素（またはその処理されたバージョン）について、セルフアテンションニューラルネットワーク層 220 は、入力位置で（すなわち、 他のブロック入力位置、および一部の実装では、それ自体）、特定の入力位置のブロック入力要素から派生した 1 つ以上のクエリを使用して、特定の位置のそれぞれの出力を生成する。セルフアテンションニューラルネットワーク層 220 の出力は、各入力位置に対応するそれぞれの層出力要素を含む層出力シーケンスである。

セルフアテンションベースのニューラルネットワーク 200 内のセルフアテンションニューラルネットワーク層のいくつかまたはすべて（例えば、図 2 に示されるセルフアテンションニューラルネットワーク層 220）は、マルチヘッドセルフアテンションニューラルネットワーク層である。マルチヘッドセルフアテンションニューラルネットワーク層は、異なるアテンションメカニズムを並列に適用して、層出力要素のそれぞれのシーケンスを生成し、次に、層出力要素の複数のシーケンスを組み合わせて、層出力要素の最終シーケンスを生成する。

セルフアテンションベースのニューラルネットワーク 200 内のセルフアテンションニューラルネットワーク層のいくつかまたはすべて（例えば、図 2 に示されるセルフアテンションニューラルネットワーク層 220）は、ブロック入力シーケンス内のそれぞれのブロック入力要素の位置情報を、アテンション機構に組み込む。

例えば、特定のブロック入力要素に関してアテンションを適用する場合（つまり、特定のブロック入力要素に対応するそれぞれのレイヤー出力要素を生成する場合）、セルフアテンションニューラルネットワーク層は、画像内の特定のブロック入力要素に対応する画像パッチの位置を表すアテンション位置埋め込みを識別する。

特定のブロック入力要素に対応するそれぞれの層出力要素を生成するとき、セルフアテンションニューラルネットワーク層は、以下の 2 つの異なるアテンション計算を実行する。

(i) 特定のブロック入力要素から生成されたクエリが、それぞれのブロック入力要素から生成されたキーのセットにアテンドする第 1 のアテンション計算（すなわち上述したアテンション機構）

(ii) 特定のブロック入力要素のアテンション位置埋め込みから生成されたクエリが、それぞれのブロック入力要素のアテンション位置埋め込みから生成されたキーのセット

にアテンドする第 2 のアテンション計算

次に、セルフアテンションニューラルネットワーク層は、２つのアテンション計算の出力を組み合わせて、例えば、２つのアテンション計算の出力の合計を決定することによって、特定のブロック入力要素の最終層出力要素を生成する。

ネットワークブロック 210a～m は、セルフアテンションニューラルネットワーク層の出力をセルフアテンションニューラルネットワーク層への入力と組み合わせる残差接続層を含む。また、ネットワークブロック 210a～m は、セルフアテンションニューラルネットワーク層の入力または出力に層正規化を適用する層正規化層を含む。

ネットワークブロック 210a～m は、１つまたは複数の位置ごとのフィードフォワードニューラルネットワーク層を含む。例えば、k 番目のネットワークブロック 210k は、フィードフォワードニューラルネットワーク層 230 を含む。フィードフォワード層 230 は、入力シーケンス 202 の入力位置ごとに、その位置で入力要素を受け取り、その位置で入力要素に一連の変換を適用して、その位置の出力要素を生成する。例えば、変換のシーケンスは、それぞれが活性化関数、例えば、非線形要素ごとの活性化関数、例えば、ReLU 活性化関数によって分離された２つ以上の学習された線形変換を含む。

出力シーケンス 204 を生成した後、セルフアテンションベースのニューラルネットワーク 200 は、出力シーケンス 204 を１つまたは複数の下流システムに提供する。例えば、セルフアテンションニューラルネットワーク 200 は、出力シーケンス 204 を１つまたは複数のヘッドニューラルネットワークに提供して、それぞれの機械学習タスクの予測を生成する。

3.クレーム

478 特許のクレーム 1 は以下の通りである。

1. 方法において、

複数のピクセルを含む１つまたは複数の画像を取得し、

１つまたは複数の画像の各画像について、画像の複数の画像パッチを決定し、各画像パッチは、画像のピクセルの異なるサブセットを含み、

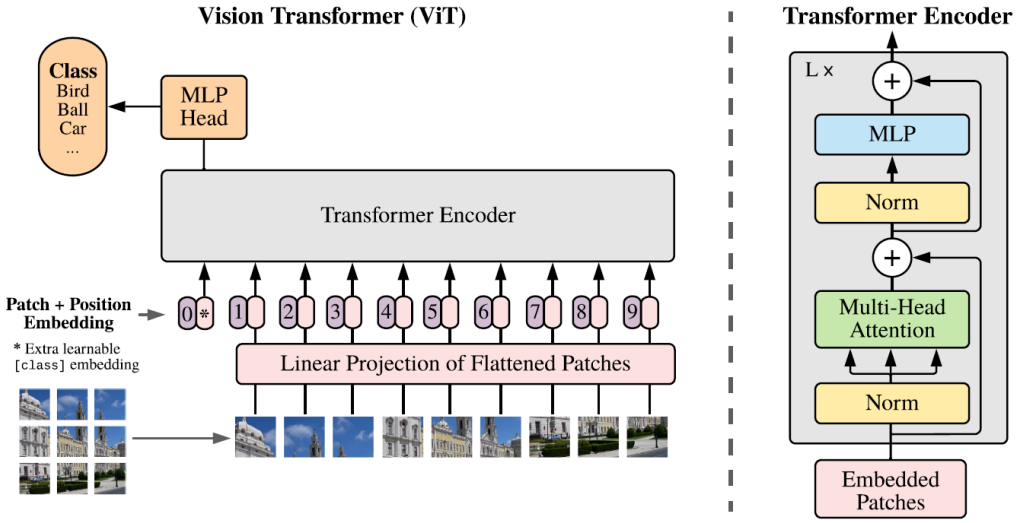
１つまたは複数の画像の各画像について、複数の入力位置のそれぞれに、各入力要素を含む入力シーケンスを生成するために、対応する複数の画像パッチを処理し、複数の入力要素は、それぞれの異なる画像パッチに対応し、

１つまたは複数の画像を特徴付けるネットワーク出力を生成するために、１つまたは

複数のセルフアテンションニューラルネットワーク層を含むニューラルネットワークを使用して入力シーケンスを処理する。

4. 本特許に関連する論文

本特許に関する論文“AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE”[1]が、Alexey Dosovitskiy 氏らにより公表されている。下記図は、Vision Transformer 及びエンコーダの構成を示す説明図である。



Vision Transformer は、画像を固定サイズのパッチに分割し、それぞれを線形に埋め込み、位置の埋め込みを追加し、結果のベクトルのシーケンスを標準の Transformer エンコーダに入力する。

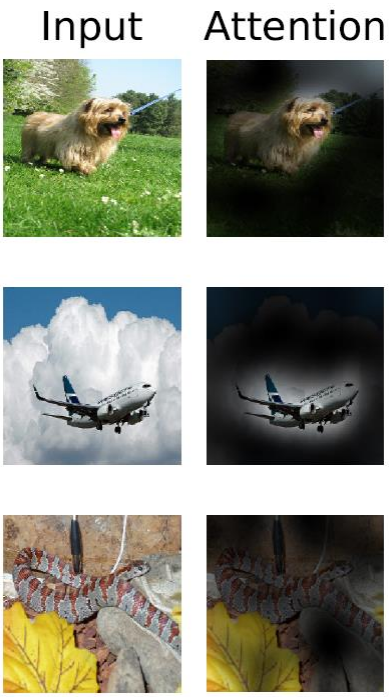
論文には、Vision Transformer の最大のモデルである ViT-H/14 と ViT-L/16 を、文献の最先端の CNN と比較した結果が示されている。下記図は対比結果を示すテーブルである。

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby
“AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE” arXiv:2010.11929v2 [cs.CV] 3 Jun 2021

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

BigTransfer（BiT）（Kolesnikov et al、2020）は、大規模な ResNet を使用して教師あり転送学習を実行するものである。NoisyStudent（Xie et al、2020）は、ImageNet と JFT で半教師あり学習を使用してトレーニングされた大規模な EfficientNet で、ラベルが削除されたものである。テーブルは、人気のある画像分類ベンチマークに関する最新技術との比較を示しており、3 回のファインチューニング実行で平均した精度の平均と標準偏差を示している。JFT-300M データセットで事前トレーニングされた VisionTransformer モデルは、すべてのデータセットで ResNet ベースのベースラインを上回り、事前トレーニングに必要な計算リソースも大幅に少なくなる。

下記図は、出力トークンから入力スペースへのアテンションの例を示す図である。



Vision Transformer が物体認識に関しアテンションしている箇所が特定できる。

以上

著者紹介

河野英仁

河野特許事務所、所長弁理士。立命館大学情報システム学博士前期課程修了、米国フランクリンピアースローセンター知的財産権法修士修了、中国清華大学法学院知的財産夏季セミナー修了、MIT(マサチューセッツ工科大学)コンピュータ科学・AI 研究所 AI コース修了。

AI 特許コンサルティング、医療 AI 特許コンサルティングの他、米国・中国特許の権利化・侵害訴訟を専門としている。著書に「世界のソフトウエア特許(共著)」、「FinTech 特許入門」、「AI/IoT 特許入門 2.0」、「ブロックチェーン 3.0(共著)」がある。