AI 特許紹介(50)

AI 特許を学ぶ！究める！

～Neural Architecture Search (NAS)特許～

2023 年 3 月 10 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

特許権者　Google

出願日　2019 年 4 月 29 日

登録日　2021 年 6 月 8 日

登録番号 US11030523

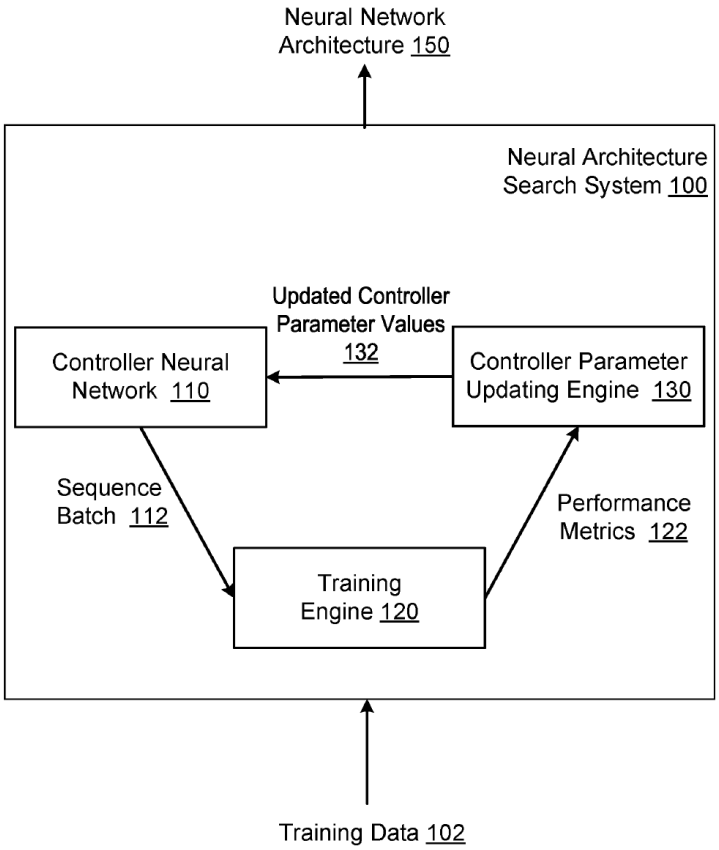
発明の名称　Neural Architecture Search

523 特許は、ニューラルネットワーク自体の構造を、強化学習を用いて自動で探索する NAS(Neural Architecture Search)技術に関する。

2.特許内容の説明

ニューラルネットワークは強力で柔軟なモデルであり、画像、音声、自然言語の理解における多くの困難な学習タスクに適している。しかしながら、ニューラルネットワークの設計は依然として困難である。523 特許は、リカレントニューラルネットワークを使用してニューラルネットワークのモデル記述を生成し、この RNN を強化学習でトレーニングして、検証セットで生成されたアーキテクチャの期待精度を最大化する。

下記図はニューラルアーキテクチャ検索システム 100 のブロック図である。



ニューラルアーキテクチャ検索システム 100 は、特定のタスクを実行するためにニューラルネットワークをトレーニングするためのトレーニングデータ 102 と、特定のタスクにおけるニューラルネットワークの性能を評価するための検証セット 104 とを取得する。そして、特定のタスクを実行するように構成された子ニューラルネットワークのアーキテクチャを決定するために、トレーニングデータ 102 と検証セット 104 を使用する。

このアーキテクチャは、子ニューラルネットワーク内の層の数、各層によって実行される操作、および子ニューラルネットワーク内の層間の接続（つまり、どの層が子ニューラルネットワーク内の他の層から入力を受け取るか）を定義する。

ニューラルアーキテクチャ検索システム 100 は、コントローラニューラルネットワーク 110、トレーニングエンジン 120、およびコントローラパラメータ更新エンジン 130 を含む。コントローラニューラルネットワーク 110 は、コントローラパラメータと呼ばれるパラメータを有し、コントローラパラメータに従って出力シーケンスを生成するように構成されたニューラルネットワークである。コントローラニューラルネットワーク

110によって生成される各出力シーケンスは、子ニューラルネットワークのそれぞれの可能なアーキテクチャを定義する。

特に、各出力シーケンスは、複数の時間ステップのそれぞれにおける各出力を含み、出力シーケンスの各時間ステップは、子ニューラルネットワークのアーキテクチャの異なるハイパーパラメータに対応する。したがって、各出力シーケンスには、各時間ステップで、対応するハイパーパラメータのそれぞれの値が含まれる。

システム 100 は、コントローラニューラルネットワーク 110 をトレーニングしてコントローラパラメータの値を調整することによって、子ニューラルネットワークのアーキテクチャを決定する。トレーニング手順の反復中に、システム 100 は、コントローラパラメータの現在の値に従って、コントローラニューラルネットワーク 110 を使用してシーケンスバッチ 112 を生成する。

バッチ 112 の各出力シーケンスについて、トレーニングエンジン 120 は、トレーニングデータ 102 で出力シーケンスによって定義されたアーキテクチャを有する子ニューラルネットワークのインスタンスをトレーニングし、検証セット 104 でトレーニングされたインスタンスのパフォーマンスを評価する。

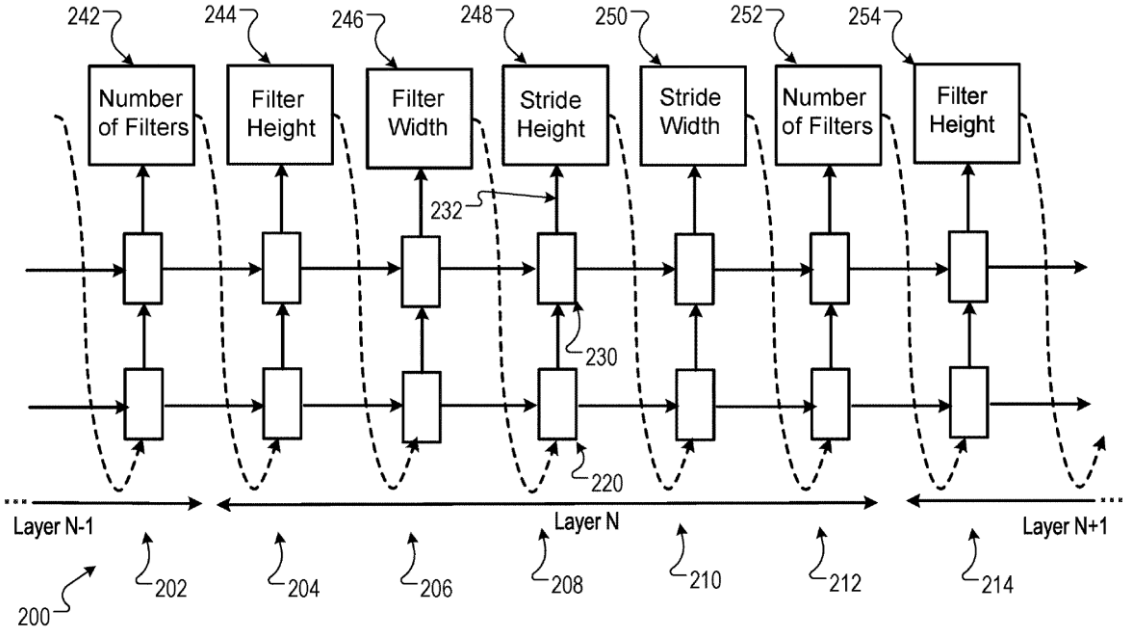
次に、コントローラパラメータ更新エンジン 130 は、バッチ 112 内の出力シーケンスの評価結果を使用して、コントローラパラメータの現在の値を更新し、タスクについてコントローラニューラルネットワーク 110 によって生成された出力シーケンスによって定義されるアーキテクチャの期待される性能を改善する。

このようにコントローラパラメータの値を繰り返し更新することにより、システム 100 はコントローラニューラルネットワーク 110 を訓練して、特定のタスクで性能を向上させた子ニューラルネットワークをもたらす出力シーケンスを生成することができる。すなわち、コントローラニューラルネットワーク 110 によって提案されたアーキテクチャの検証セット 104 で期待される精度を最大化する。

コントローラニューラルネットワーク 110 がトレーニングされると、システム 100 は、子ニューラルネットワークの最終的なアーキテクチャとして、検証セット 104 で最良のパフォーマンスを発揮したアーキテクチャを選択するか、またはコントローラのトレーニングされた値に従って新しい出力シーケンスを生成し、コントローラのパラメータを変更し、新しい出力シーケンスによって定義されたアーキテクチャを、子ニューラルネットワークの最終的なアーキテクチャとして使用する。

次に、ニューラルネットワーク検索システム 100 は、子ニューラルネットワークのアーキテクチャを指定するアーキテクチャデータ 150、すなわち、子ニューラルネットワークの一部である層、層間の接続性、および層によって実行される操作を指定するデータを出力する。例えば、ニューラルネットワーク検索システム 100 は、トレーニングデータを提出したユーザにアーキテクチャデータ 150 を出力する。

下記図は、出力シーケンスを生成するコントローラニューラルネットワーク 110 のブロック図である。



ダイアグラム 200 は、出力シーケンスの生成中に 7 つの例示的な時間ステップ 202～214 に対してコントローラニューラルネットワーク 110 によって実行される処理を示す。7 つの時間ステップ 202～214 のそれぞれは、ニューラルネットワークアーキテクチャの異なるハイパーパラメータに対応する。

コントローラニューラルネットワーク 110 は、リカレントニューラルネットワークであり、時間ステップごとに、前の時間に対応するハイパーパラメータの値を入力として受け取るように構成され、指定された出力シーケンスをステップインし、入力を処理してリカレントニューラルネットワークの現在の隠れ状態を更新する。

図の例では、時間ステップ 208 で、層 220 および 230 は、前の時間ステップ 206 からのハイパーパラメータの値を入力として受け取り、層 220 および 230 の隠れ状態を時間ステップ 206 から更新して、出力として隠れ状態 232 を生成する。

コントローラニューラルネットワーク 110 はまた、出力シーケンスにおける各時間ステップに対するそれぞれの出力層、例えば、時間ステップ 202～214 それぞれに対する出力層 242～254 を含む。各出力層は、タイムステップで更新された隠れ状態を含む出力層入力を受け取り、タイムステップでのハイパーパラメータの可能な値に対するスコア分布を定義するタイムステップの出力を生成する。

例えば、各出力層は、最初に出力層の入力を、対応するハイパーパラメータの可能な値の数の適切な次元に射影し、射影された出力層の入力にソフトマックスを適用して、ハイパーパラメータの複数の可能な値のそれぞれについて、タイムステップでの各スコアを生成する。時間ステップ 208 の出力層 248 は、隠れ状態 232 を含む入力を受信し、ストライドハイトハイパーパラメータの複数の可能な値のそれぞれについて各スコアを生成する。

したがって、出力シーケンス内の所与の時間ステップのハイパーパラメータ値を生成するために、システム 100 は、コントローラニューラルネットワークへの入力として、出力シーケンスの前の時間ステップでのハイパーパラメータの値を提供し、コントローラニューラルネットワークは、タイムステップでのハイパーパラメータの可能な値に対するスコア分布を定義するタイムステップの出力を生成する。

出力シーケンスの最初の時間ステップでは、先行する時間ステップがないため、システム 100 は代わりに所定のプレースホルダー入力を提供する。次いで、システム 100 は、スコア分布に従って可能な値からサンプリングして、出力シーケンスの時間ステップにおけるハイパーパラメータの値を決定する。特定のハイパーパラメータが取ることができる可能な値は、トレーニングの前に固定されており、可能な値の数はハイパーパラメータごとに異なる。

図の例では、子ニューラルネットワークは畳み込みニューラルネットワークであり、ハイパーパラメータは、子ニューラルネットワーク内の畳み込みニューラルネットワーク層ごとのハイパーパラメータを含む。特に、図において、時間ステップ 202 は子ニューラルネットワークの畳み込み層Ｎ－１のハイパーパラメータに対応し、時間ステップ 204～212 は畳み込み層Ｎのハイパーパラメータに対応し、時間ステップ 214 は畳み込み層Ｎ+1 のハイパーパラメータに対応する。

図の例では、畳み込み層の場合、層によって実行される操作を定義するハイパーパラメータは、層のフィルタの数、各フィルタのフィルタの高さ、各フィルタのフィルタ幅、

各フィルタを適用するストライドの高さ、および各フィルタのストライド幅である。

3.クレーム

523 特許のクレーム 1 は以下の通りである。

1. 方法において、

複数のコントローラパラメータを有するコントローラニューラルネットワークを使用して、コントローラパラメータの現在の値に従って、出力シーケンスのバッチを生成し、バッチ内の各出力シーケンスは、特定のニューラルネットワークタスクを実行するように構成された子ニューラルネットワークのそれぞれのアーキテクチャを定義しており、

バッチ内の各出力シーケンスについて、

特定のニューラルネットワークタスクを実行するために、トレーニングデータの出力シーケンスによって定義されたアーキテクチャを有する子ニューラルネットワークのそれぞれのインスタンスをトレーニングし、トレーニングデータは、それぞれが各ターゲットトレーニング出力に関連付けられた複数のトレーニング入力を含み、

トレーニングの後、検証データを使用して、特定のニューラルネットワークタスクでの子ニューラルネットワークのトレーニング済みインスタンスのパフォーマンスを評価することに基づいて、特定のニューラルネットワークタスクでの子ニューラルネットワークのトレーニング済みインスタンスのパフォーマンスメトリックを決定し、検証データは、トレーニングデータとは異なる 1 つ以上のトレーニング入力を含み、

コントローラニューラルネットワークのコントローラパラメータの現在の値を調整するために、子ニューラルネットワークのトレーニング済みインスタンスのパフォーマンスメトリックを使用する。

4. 本特許に関連する論文

本特許に関する論文“NEURAL ARCHITECTURE SEARCH WITH REINFORCEMENT LEARNING?”[1]が、Google Brain の Barret Zoph 氏らにより公表されている。

本論文では、リカレントニューラルネットワークを使用してニューラルネットワークのモデル記述を生成し、この RNN を強化学習でトレーニングして、検証セットで生成

[1] Barret Zoph, Quoc V. Le “NEURAL ARCHITECTURE SEARCH WITH REINFORCEMENT LEARNING” arXiv:1611.01578v2 [cs.LG] 15 Feb 2017

されたアーキテクチャの期待精度を最大化する仕組みが記載されている。

下記表は Neural Architecture Search およびその他の最先端モデルのパフォーマンスを示す。

Model	Depth	Parameters	Error rate (%)
Network in Network (Lin et al., 2013)	-	-	8.81
All-CNN (Springenberg et al., 2014)	-	-	7.25
Deeply Supervised Net (Lee et al., 2015)	-	-	7.97
Highway Network (Srivastava et al., 2015)	-	-	7.72
Scalable Bayesian Optimization (Snoek et al., 2015)	-	-	6.37
FractalNet (Larsson et al., 2016)	21	38.6M	5.22
with Dropout/Drop-path	21	38.6M	4.60
ResNet (He et al., 2016a)	110	1.7M	6.61
ResNet (reported by Huang et al. (2016c))	110	1.7M	6.41
ResNet with Stochastic Depth (Huang et al., 2016c)	110	1.7M	5.23
	1202	10.2M	4.91
Wide ResNet (Zagoruyko & Komodakis, 2016)	16	11.0M	4.81
	28	36.5M	4.17
ResNet (pre-activation) (He et al., 2016b)	164	1.7M	5.46
	1001	10.2M	4.62
DenseNet ($L = 40, k = 12$) Huang et al. (2016a)	40	1.0M	5.24
DenseNet($L = 100, k = 12$) Huang et al. (2016a)	100	7.0M	4.10
DenseNet ($L = 100, k = 24$) Huang et al. (2016a)	100	27.2M	3.74
DenseNet-BC ($L = 100, k = 40$) Huang et al. (2016b)	190	25.6M	3.46
Neural Architecture Search v1 no stride or pooling	15	4.2M	5.50
Neural Architecture Search v2 predicting strides	20	2.5M	6.01
Neural Architecture Search v3 max pooling	39	7.1M	4.47
Neural Architecture Search v3 max pooling + more filters	39	37.4M	3.65

CIFAR-10 データセットでは、テストセットの精度の点で、人間が発明した最高のアーキテクチャに匹敵する新しいネットワークアーキテクチャをゼロから始める方法で設計することができる。本論文における CIFAR-10 モデルは、3.65 のテストエラー率を達成した。これは、同様のアーキテクチャスキームを使用した以前の最先端のモデルよりも 0.09%良く、1.05 倍高速である。

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 3」、「ブロックチェーン 3.0(共著)」がある。