

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7629257号
(P7629257)

(45)発行日 令和7年2月13日(2025.2.13)

(24)登録日 令和7年2月4日(2025.2.4)

(51)国際特許分類

F I

G 0 6 F 9/50 (2006.01) G 0 6 F 9/50 1 2 0 A

G 0 6 N 20/00 (2019.01) G 0 6 N 20/00

請求項の数 16 (全27頁)

(21)出願番号	特願2022-538947(P2022-538947)	(73)特許権者	390009531
(86)(22)出願日	令和2年12月14日(2020.12.14)		インターナショナル・ビジネス・マシー
(65)公表番号	特表2023-508076(P2023-508076		ンズ・コーポレーション
	A)		INTERNATIONAL BUSI
(43)公表日	令和5年2月28日(2023.2.28)		NESS MACHINES CORPO
(86)国際出願番号	PCT/IB2020/061886		RATION
(87)国際公開番号	WO2021/130596		アメリカ合衆国10504 ニューヨー
(87)国際公開日	令和3年7月1日(2021.7.1)		ク州 アーモンク ニュー オーチャード
審査請求日	令和5年5月25日(2023.5.25)		ロード
(31)優先権主張番号	16/724,613		New Orchard Road, A
(32)優先日	令和1年12月23日(2019.12.23)		rmonk, New York 105
(33)優先権主張国・地域又は機関	米国(US)		04, United States of
			America
		(74)代理人	100112690
			弁理士 太佐 種一

最終頁に続く

(54)【発明の名称】 アプリケーション・ベースのプロファイリングを用いた機械学習ワークロードの弾力的な実行

(57)【特許請求の範囲】

【請求項1】

コンピュータ・システムであって、
処理ユニットと、
前記処理ユニットに動作可能に結合されたメモリと
を含み、前記メモリは、
1または複数の機械学習（ML）アプリケーションについてMLアプリケーション実行データおよびリソース使用データを含み構成されるプロファイルを生成することと、
生成された前記プロファイルから1または複数の特徴および1または複数のシグネチャを識別することと、
MLアプリケーション実行パフォーマンスを1または複数の計算リソースに相関させるよう構成されたML実行モデルを構築することであって、前記ML実行モデルは、識別された前記1または複数の特徴および1または複数のシグネチャを利用する、構築することと、
前記ML実行モデルによって、アプリケーション実行をサポートするためにリソース管理コンポーネントから前記1または複数のMLアプリケーションのうちの少なくとも1つに1または複数のリソースを動的に割り当てることおよび要求することを行うように、後続のアプリケーション実行への1または複数の指令を適用することであって、前記ML実行モデルは、生成された前記プロファイルからの更新を受信して適用し、前記1または複数のMLアプリケーションの実行中に割り当てを変更するように構成される、適用することと

を前記処理ユニットに実行するようにさせる命令を格納する、コンピュータ・システム。

【請求項 2】

前記 ML 実行モデルは、1 または複数の ML 技術を使用して、前記 1 または複数の ML アプリケーションの少なくとも 1 つに対する 1 または複数のリソース要件を評価し、前記 1 または複数の ML アプリケーションの実行をサポートするための 1 または複数の計算リソースを識別するための出力を生成するように構成され、前記 1 または複数の指令は、生成された前記出力に一致する、請求項 1 に記載のコンピュータ・システム。

【請求項 3】

前記動的に割り当てることは、前記 ML 実行モデルが、リソース利用可能性およびアプリケーション処理サポートに関してリソース割り当てアクションを実行することを含み、

前記リソース割り当てアクションは、任意のリソースを削減または拡張するためのものであり、1 または複数の削減または拡張された前記リソースは、同一または異なる属性を有する、請求項 1 または 2 に記載のコンピュータ・システム。

【請求項 4】

前記プロファイルを生成することは、1 または複数の ML ルーチンに関するリソース使用データを監視し、収集し、監視した前記リソース使用データを使用して将来の ML アプリケーションのための割り当てを予測する、請求項 1 ～ 3 のいずれか 1 項に記載のコンピュータ・システム。

【請求項 5】

前記プロファイルを生成するためおよび実行モデルを構築するための前記 1 または複数の特徴および 1 または複数のシグネチャの識別は、ユーザ入力で導出される、請求項 1 ～ 4 のいずれか 1 項に記載のコンピュータ・システム。

【請求項 6】

コンピュータ・システムであって、
メモリに動作可能に結合された処理ユニットと、
アプリケーション・ベースのプロファイリングを用いて 1 または複数の機械学習ワークロードを動的に実行するためのツールを有する、前記処理ユニットと通信する人工知能 (AI) プラットフォームと

を含み、前記ツールは、
機械学習 (ML) アプリケーション実行データおよびリソース使用データを含み構成される結合プロファイルを生成するプロファイル・マネージャと、

生成された前記結合プロファイルから 1 または複数の特徴および 1 または複数のシグネチャを識別し、ML アプリケーション実行パフォーマンスおよびリソース使用のための ML 実行モデルを構築する機械学習マネージャであって、前記 ML 実行モデルは、識別された前記 1 または複数の特徴および 1 または複数のシグネチャを利用する、機械学習マネージャと、

前記 ML 実行モデルを適用し、前記 ML 実行モデルが、アプリケーション実行をサポートするためにリソース管理コンポーネントからの 1 または複数のリソースを動的に割り当てることおよび要求することを含む、後続のアプリケーション実行への 1 または複数の指令を提供するディレクタと

を含み、前記動的に割り当てることは、前記 ML 実行モデルが、リソース利用可能性およびアプリケーション処理サポートに関してリソース割り当てアクションを実行することを含み、

前記リソース割り当てアクションは、任意のリソースを削減または拡張するためのものであり、1 または複数の削減または拡張された前記リソースは、同一または異なる属性を有し、前記コンピュータ・システムは、

アプリケーション実行パターンおよびリソース使用パターンに基づいて前記リソース割り当てアクションを呼び出すことを含む、1 または複数の ML アプリケーションの反復にわたり計算リソースの割り当てを変更する前記ディレクタを含む、コンピュータ・システム。

10

20

30

40

50

【請求項 7】

コンピュータ・システムであって、
メモリに動作可能に結合された処理ユニットと、
アプリケーション・ベースのプロファイリングを用いて 1 または複数の機械学習ワーク
ロードを動的に実行するためのツールを有する、前記処理ユニットと通信する人工知能（
AI）プラットフォームと
を含み、前記ツールは、
機械学習（ML）アプリケーション実行データおよびリソース使用データを含み構成さ
れる結合プロファイルを生成するプロファイル・マネージャと、
生成された前記結合プロファイルから 1 または複数の特徴および 1 または複数のシグネ
チャを識別し、ML アプリケーション実行パフォーマンスおよびリソース使用のための ML
実行モデルを構築する機械学習マネージャであって、前記 ML 実行モデルは、識別され
た前記 1 または複数の特徴および 1 または複数のシグネチャを利用する、機械学習マネ
ージャと、
前記 ML 実行モデルを適用し、前記 ML 実行モデルが、アプリケーション実行をサポート
するためにリソース管理コンポーネントからの 1 または複数のリソースを動的に割り当
てることおよび要求することを含む、後続のアプリケーション実行への 1 または複数の指
令を提供するディレクタと
を含み、前記プロファイル・マネージャは、1 または複数のコールバック関数を利用し
て、1 または複数のアプリケーション関連パラメータを用いてアプリケーション実行をプ
ロファイルし、リソース使用情報をプロファイルする、コンピュータ・システム。

【請求項 8】

1 または複数の機械学習（ML）アプリケーションについて ML アプリケーション実行デ
ータおよびリソース使用データを含み構成されるプロファイルを生成することと、
生成された前記プロファイルから 1 または複数の特徴および 1 または複数のシグネチャ
を識別することと、
ML アプリケーション実行パフォーマンスを 1 または複数の計算リソースに相関させる
ように構成された ML 実行モデルを構築することであって、前記 ML 実行モデルは、識別
された前記 1 または複数の特徴および 1 または複数のシグネチャを利用する、構築するこ
とと、
前記 ML 実行モデルによって、アプリケーション実行をサポートするために、リソース
管理コンポーネントから前記 1 または複数の ML アプリケーションのうちの少なくとも 1 つ
に 1 または複数のリソースを動的に割り当てることおよび要求することを行うように、後
続のアプリケーション実行への 1 または複数の指令を適用することであって、前記 ML 実
行モデルは、生成された前記プロファイルからの更新を受信して適用し、前記 1 または複
数の ML アプリケーションの実行中に割り当てを変更するように構成される、適用するこ
とと
を含む、方法。

【請求項 9】

前記 ML 実行モデルは、1 または複数の ML 技術を使用して、前記 1 または複数の ML
アプリケーションの少なくとも 1 つに対する 1 または複数のリソース要件を評価し、前記
1 または複数の ML アプリケーションの実行をサポートするための 1 または複数の計算リ
ソースを識別するための出力を生成するように構成され、前記 1 または複数の指令は、生
成された前記出力に一致する、請求項 8 に記載の方法。

【請求項 10】

前記動的に割り当てることは、リソース利用可能性およびアプリケーション処理サポート
に関してリソース割り当てアクションを実行することを含み、
前記リソース割り当てアクションは、任意のリソースを削減または拡張するためのもの
であり、1 または複数の削減または拡張された前記リソースは、同一または異なる属性
を有する、請求項 8 または 9 に記載の方法。

【請求項 1 1】

前記プロファイルを生成することは、1 または複数の ML ルーチンに関するリソース使用データを監視し、収集することと、監視した前記リソース使用データを使用して将来の ML アプリケーションのための割り当てを予測することを含む、請求項 8 ～ 1 0 のいずれか 1 項に記載の方法。

【請求項 1 2】

前記プロファイルを生成するためおよび実行モデルを構築するための前記 1 または複数の特徴および 1 または複数のシグネチャの識別は、ユーザ入力で導出される、請求項 8 ～ 1 1 のいずれか 1 項に記載の方法。

【請求項 1 3】

機械学習 (ML) アプリケーション実行データおよびリソース使用データを含み構成される結合プロファイルを生成することと、

生成された前記結合プロファイルから 1 または複数の特徴および 1 または複数のシグネチャを識別し、ML アプリケーション実行パフォーマンスおよびリソース使用のための ML 実行モデルを構築することであって、前記 ML 実行モデルは、識別された前記 1 または複数の特徴および 1 または複数のシグネチャを利用する、構築することと、

前記 ML 実行モデルを適用し、アプリケーション実行をサポートするために、リソース管理コンポーネントからの 1 または複数のリソースを動的に割り当てることおよび要求することを含む、後続のアプリケーション実行への 1 または複数の指令を提供することとを含む方法であり、

前記動的に割り当てることは、リソース利用可能性およびアプリケーション処理サポートに関してリソース割り当てアクションを実行することを含む、

前記リソース割り当てアクションは、任意のリソースを削減または拡張するためのものであり、1 または複数の削減または拡張された前記リソースは、同一または異なる属性を有し、前記方法は、

アプリケーション実行パターンおよびリソース使用パターンに基づいて前記リソース割り当てアクションを呼び出すことを含む、1 または複数の ML アプリケーションの反復にわたり計算リソース割り当てを変更すること

をさらに含む、方法。

【請求項 1 4】

機械学習 (ML) アプリケーション実行データおよびリソース使用データを含み構成される結合プロファイルを生成することと、

生成された前記結合プロファイルから 1 または複数の特徴および 1 または複数のシグネチャを識別し、ML アプリケーション実行パフォーマンスおよびリソース使用のための ML 実行モデルを構築することであって、前記 ML 実行モデルは、識別された前記 1 または複数の特徴および 1 または複数のシグネチャを利用する、構築することと、

前記 ML 実行モデルを適用し、アプリケーション実行をサポートするために、リソース管理コンポーネントからの 1 または複数のリソースを動的に割り当てることおよび要求することを含む、後続のアプリケーション実行への 1 または複数の指令を提供することと

を含み、前記結合プロファイルを生成することは、1 または複数のコールバック関数を利用して、1 または複数のアプリケーション関連パラメータを用いてアプリケーション実行をプロファイルし、リソース使用情報をプロファイルする、方法。

【請求項 1 5】

プログラムがコンピュータ上で実行されたとき、請求項 8 ～ 1 4 のいずれか 1 項に記載の方法を実行するように適合されたプログラム・コード手段を含む、コンピュータ・プログラム。

【請求項 1 6】

アプリケーション・ベースのプロファイリングを用いて機械学習ワークロードの動的な実行をサポートするためのコンピュータ・プログラムを格納した記録媒体であって、前記コンピュータ・プログラムは、プロセッサに、請求項 8 ～ 1 4 のいずれか 1 項に記載の方

10

20

30

40

50

法を実行させるためのものである、記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本実施形態は、機械学習モデルをトレーニングして、機械学習ワークロードの実行をプロファイルすることに関する。より詳細には、本実施形態は、アプリケーション実行およびリソース使用の結合プロファイルを生成することと、トレーニングされた機械学習モデルにカプセル化された実行モデルを利用して、機械学習アプリケーション実行および対応するパフォーマンスをサポートするためにリソース管理システムと相互作用することに関する。

10

【背景技術】

【0002】

人工知能（AI）は、人間に関連した、コンピュータおよびコンピュータの挙動に向けられたコンピュータ科学の分野に関する。AIは、マシンが、情報に基づいて、所与のトピックにおける成功の機会を最大にするような決定を行うことが可能なインテリジェンスを意味する。より具体的には、AIは、データセットから学習して、問題を解決し、関連するレコメンデーションを提供することができる。例えば、人工知能コンピュータ・システムの分野では、自然言語システム（IBM（登録商標）Watson（登録商標）人工知能コンピュータ・システムまたは他の自然言語質問応答システム）は、システムが獲得した知識に基づいて自然言語を処理する。自然言語を処理するために、システムは、知識のコーパスから導出されたデータを用いてトレーニングされ得るが、もたらされる結果は、様々な理由で様々な程度の精度を有する可能性がある。

20

【0003】

人工知能（AI）のサブセットである機械学習（ML）は、アルゴリズムを利用してデータから学習し、このデータに基づいて予見を生成する。MLは、明示的にプログラムされていないタスクを実行することによって学習の挙動を実証することができるニューラル・ネットワークを含むモデルの作成を介したAIの応用である。MLワークロードは、大きなデータセット、データに対する高速かつ並列なアクセス、および学習をサポートするためのトレーニング用アルゴリズムを必要とする。MLワークロードをサポートするハードウェアの例には、これらに限定されるものではないが、グラフィック処理ユニット（GPU）などのアクセラレータ、中央処理ユニット（CPU）インフラストラクチャおよびストレージ・ソリューションが含まれる。MLワークロードに対するリソース使用の特性は、通常、デバッグおよび実行タイミングの調査のために収集されるリソース・プロファイリングを有する。リソース・プロファイリング収集は、現在、アプリケーションまたはワークロード実行とは独立した、タイミング・フレームワークを使用する。ワークロード実行とリソース使用との相関を改善するために、同一のタイミング・フレームワークを使用して、ワークロード実行およびリソース使用の結合プロファイリングを有することは有利であろう。さらに、結合プロファイリングは、異なる組のMLパラメータを用いて、タスクレベルまたは反復レベルでのような粒度の細かい相関を可能にするであろう。したがって、MLワークロードをサポートするために適切な量のリソースを用いて、結合プロ

30

40

【発明の概要】

【0004】

実施形態は、効果的で効率的なMLアプリケーション実行を計算リソース割り当てと融合するための人工知能プラットフォームを採用するためのシステム、コンピュータ・プログラム製品および方法を含む。

【0005】

一実施形態において、システムは、人工知能（AI）プラットフォームとともに使用して、MLアプリケーション実行モデラーをトレーニングし、維持し、アプリケーション・

50

ベースのプロファイリングを使用して機械学習ワークロードを弾力的に実行するために提供される。図示されるように、システムは、メモリに動作可能に結合された、例えばプロセッサなどの処理ユニットおよび処理ユニットと通信するＡＩプラットフォームを有して提供される。ＡＩプラットフォームは、プロファイル・マネージャ、ＭＬマネージャおよびディレクタの形態のツールを埋め込まれている。プロファイル・マネージャは、ＭＬアプリケーション実行データおよびリソース使用データの両方から構成される結合プロファイルを生成するように機能する。ＭＬマネージャは、結合プロファイルから、１または複数の特徴およびシグネチャを識別し、ＭＬアプリケーション実行およびリソース使用のためのＭＬ実行モデルを構築するように機能する。ＭＬ実行モデルは、特徴（単数または複数）およびシグネチャ（単数または複数）を利用する。ディレクタは、ＭＬ実行モデルを適用し、後続のアプリケーション実行への１または複数の指令を提供するように機能する。ディレクタによるアプリケーションは、ＭＬ実行が、弾力的な割り当てがアプリケーション実行をサポートしながら、リソース管理コンポーネントからの１または複数のリソースを弾力的に割り当て、また、要求することをサポートしかつ可能にする。

【０００６】

別の実施形態においては、アプリケーション・ベースのプロファイリングを使用してＭＬワークロードの弾力的な実行をサポートするコンピュータ・プログラム製品が提供される。コンピュータ・プログラム製品は、そこに具現化されたプログラム・コードを有するコンピュータ可読ストレージ媒体を含み、プログラム・コードは、プロセッサによって実行可能であり、ＭＬアプリケーション実行モデラーをトレーニングしてかつ維持し、アプリケーション・ベースのプロファイリングを使用してＭＬワークロードを弾力的に実行する。プログラム・コードは、ＭＬアプリケーション実行データおよびリソース使用データの両方から構成される結合プロファイルを生成するために提供される。プログラム・コードは、結合プロファイルから１または複数の特徴およびシグネチャを識別し、ＭＬアプリケーション実行およびリソース使用のためのＭＬ実行モデルを構築するために提供される。ＭＬ実行モデルは、特徴（単数または複数）およびシグネチャ（単数または複数）を利用する。プログラム・コードは、ＭＬ実行モデルを適用し、その後のアプリケーション実行への１または複数の指令を提供する。プログラム・コードの適用は、ＭＬ実行が、弾力割り当てがアプリケーションの実行をサポートしながら、リソース管理コンポーネントから１または複数のリソースを弾力的に割り当て、要求することをサポートしかつ可能にする。

【０００７】

さらに別の実施形態においては、アプリケーション・ベースのプロファイリングを用いてＭＬワークロードを弾力的に実行するためにＭＬ実行モデルをトレーニングおよび維持することを含む、アプリケーション・ベースのプロファイリングを使用してＭＬワークロードの弾力的実行をサポートする方法が提供される。ＭＬアプリケーション実行データおよびリソース使用データの両方から構成される結合プロファイルが生成される。結合プロファイルからの１または複数の特徴および１または複数のシグネチャが識別され、ＭＬアプリケーション実行およびリソース使用のためのＭＬ実行モデルが構築される。ＭＬ実行モデルは、特徴（単数または複数）およびシグネチャ（単数または複数）を利用する。ＭＬ実行モデルが適用され、その後のアプリケーション実行への１または複数の指令を提供する。ＭＬ実行モデルの適用は、ＭＬ実行が、弾力割り当てがアプリケーション実行をサポートしながら、リソース管理コンポーネントから１または複数のリソースを弾力的に割り当て、かつ要求することをサポートし、可能とする。

【０００８】

１つの側面によれば、メモリに動作可能に結合された処理ユニットと、アプリケーション・ベースのプロファイリングを用いて１または複数の機械学習ワークロードを弾力的に実行するツールを有する、処理ユニットと通信する人工知能（ＡＩ）プラットフォームとを含むコンピュータ・システムが提供され、ここで、ツールは、機械学習（ＭＬ）アプリケーション実行およびリソース使用で構成される結合プロファイルを生成するプロファイ

ル・マネージャと、生成された結合プロファイルから1または複数の特徴および1または複数のシグネチャを識別し、MLアプリケーション実行パフォーマンスおよびリソース使用のためのML実行モデルを構築する機械学習マネージャであって、ML実行モデルは、識別された1または複数の特徴および1または複数のシグネチャを利用する、機械学習マネージャと、ML実行モデルを適用し、アプリケーション実行をサポートするために、ML実行モデルが、リソース管理コンポーネントからの1または複数のリソースを弾力的に割り当てることおよび要求することを含む、後続のアプリケーション実行への1または複数の指令を提供するディレクタとを含む。

【0009】

別の側面によれば、アプリケーション・ベースのプロファイリングを用いて機械学習ワークロードの弾力的な実行をサポートするためのコンピュータ・プログラム製品が提供され、コンピュータ・プログラム製品は、そこに具現化されたプログラム・コードを有するコンピュータ可読ストレージ媒体を含み、プログラム・コードは、プロセッサによって、機械学習（ML）アプリケーション実行およびリソース使用で構成される結合プロファイルを生成することと、生成された結合プロファイルから1または複数の特徴および1または複数のシグネチャを識別し、MLアプリケーション実行パフォーマンスおよびリソース使用のためのML実行モデルを構築することであって、ML実行モデルは、識別された1または複数の特徴および1または複数のシグネチャを利用する、構築することと、ML実行モデルを適用し、アプリケーション実行をサポートするために、リソース管理コンポーネントからの1または複数のリソースを弾力的に割り当てることおよび要求することを含む、後続のアプリケーション実行への1または複数の指令を提供することとを行うように実行可能である。

【0010】

別の側面によれば、方法が提供され、方法は、機械学習（ML）アプリケーション実行およびリソース使用で構成される結合プロファイルを生成することと、生成された結合プロファイルから1または複数の特徴および1または複数のシグネチャを識別し、MLアプリケーション実行パフォーマンスおよびリソース使用のためのML実行モデルを構築することであって、ML実行モデルは、識別された1または複数の特徴および1または複数のシグネチャを利用する、構築することと、ML実行モデルを適用し、アプリケーション実行をサポートするために、リソース管理コンポーネントからの1または複数のリソースを弾力的に割り当てることおよび要求することを含む、後続のアプリケーション実行への1または複数の指令を提供することとを含む。

【0011】

これらおよび他の特徴および利点は、添付の図面と関連して以下の現時点で好ましい実施形態の詳細な説明から、明らかになるであろう。

【0012】

以下の図面を参照しながら、本発明の好適な実施形態について、単なる例示として説明する。

【図面の簡単な説明】

【0013】

【図1】好適な実施形態による、リソースを有効に使用してMLアプリケーションの効率的な実行をサポートするML結合プロファイルの概要を説明するためのブロック図。

【図2】好適な実施形態による、MLプログラムの効率的な実行および計算リソースの有効な割り当ておよび使用をサポートするための人工知能プラットフォームを有するコンピュータ・システムの概略図。

【図3】好ましい実施形態による、図2に示すような人工知能プラットフォームおよび対応するツール、並びに、それらに関連するアプリケーション・プログラム・インタフェースを示すブロック図。

【図4】好ましい実施形態による、MLモデリングおよび計算リソース割り当ての弾力性のためのプロセスを示すフロー図。

【図5】好ましい実施形態による、図1～図4に関して上述したシステムおよびプロセスを実現するためのクラウド・ベースのサポート・システムのコンピュータ・システム／サーバの一例を示すブロック図。

【図6】好ましい実施形態による、クラウド・コンピュータ環境を示すブロック図。

【図7】好ましい実施形態による、クラウド・コンピューティング環境によって提供される機能抽象モデル層のセットを示すブロック図。

【発明を実施するための形態】

【0014】

図面に示される特徴は、他に明示的に示されない限り、いくつかの実施形態のみであり、すべての実施形態を例示するものではないことを意味する。

【0015】

本実施形態の構成要素は、本明細書において図面中に一般的に説明され、また記載されるが、多種多様な構成で配置され、また設計されてもよいことが容易に理解されるであろう。よって、図面に示されるような、本実施形態の装置、システム、方法およびコンピュータ・プログラム製品の実施形態についての以下の詳細な説明は、特許請求されるように、実施形態の範囲を限定することを意図するものではなく、単に選択された実施形態を表すに過ぎない。

【0016】

本明細書全体を通して、「選択された実施形態」、「一実施形態」または「実施形態」との言及は、その実施形態に関連して説明された特定の特徴、構造または特性が、少なくとも1つの実施形態に含まれることを意味する。よって、本明細書全体にわたる種々の箇所での「選択された実施形態」、「一実施形態においては」という語句の出現は、必ずしも同一の実施形態を参照するとは限らない。

【0017】

図示された実施形態は、同様の部分が同様の数字によって一貫して指定される図面を参照することによって、最もよく理解されるであろう。以下の説明は、例としてのみ意図され、特許請求される実施形態に一致する、デバイス、システムおよびプロセスのある選択された実施形態を単に例示する。

【0018】

機械学習は、データを準備し、アルゴリズムをトレーニングし、機械学習モデルを生成し、次いで予測を作成および改良するプロセスに従う。機械学習モデルは、概して、教師あり機械学習、教師なし機械学習、強化機械学習および深層学習といった基本的なカテゴリの1つに分類される。例えば、深層学習は、人間の脳が情報を処理するやり方のモデルとしてニューラル・ネットワークを採用する。ニューラル・ネットワークの基本ユニットは、ニューロンと参照され、これは、典型的には層に編成される。ニューラル・ネットワークは、ニューロンの抽象バージョンと共通点がある多数の相互接続された処理ユニットをシミュレートすることによって動作する。ニューラル・ネットワークには、典型的には、入力フィールドを表すユニットを有する入力層と、1または複数の隠れ層と、ターゲットフィールドを表す単一のユニットまたは複数のユニットを有する出力層とを含む、3つの部分がある。これらのユニットは、様々な接続強度または重みをもって接続されている。入力データが第1層に与えられ、各ニューロンから次の層のすべてのニューロンに値が伝播される。最終的には、出力層から結果が配信される。深層学習複素ニューラル・ネットワーク（Deep learning complex neural networks）は、人間の脳がどのように機能するかをエミュレートするように設計されているため、コンピュータは、不完全に定義された抽象概念および問題をサポートするようにトレーニングされ得る。ニューラル・ネットワークおよび深層学習は、画像認識、音声およびコンピュータ・ビジョンのアプリケーションにおいてしばしば使用される。機械学習は、データに基づいて、学習し、予測を生成することができる。機械学習は、また、効率的なワークロードの実行に影響を与えるために、結合プロファイリング・データを使用して、細粒度の間隔でのリソースの有効な割り当てに関する洞察を得ることができる。

10

20

30

40

50

【0019】

MLモデルは、モデルの処理ニーズを適切にサポートするために広範なハードウェア要件を有することが理解される。このようなハードウェア・リソース要件の例には、これらに限定されるものではないが、RAM、ディスク空間、GPUおよびCPUが含まれる。リソースの管理は、複数のMLモデル間でのリソースの共有をサポートするために割り当てられてもよい。本明細書において示され、また説明されるように、また、図面において実証されるように、MLワークロード情報は、アプリケーション実行をサポートするためにリソースを適切に割り当てるために、リソース情報に相関される。MLルーチン上でのリソース使用は、実行中に、監視され、また収集され、その後、後続のMLワークロードに対するリソースの予測および割り当てのために使用される。よって、ML技術は、後続のMLワークロードのためのリソース割り当てのために利用される。

10

【0020】

図1を参照すると、リソースの有効な使用とともにMLアプリケーションの効率的な実行をサポートするためのML結合プロファイルの概要を示すブロック図100が提供される。図示されるように、ML実行モデル110は、MLアプリケーション実行およびリソース使用のために開発される。ML実行モデル110は、学習し、予測し、他のMLアプリケーションにリソースを割り当てるために機械学習技術を用いる。より具体的には、ML実行モデル110は、ML技術に基づくパフォーマンス予測モデルである。実行モデルは、MLアプリケーションの以前の実行からの履歴データ120と、以前のML実行モデル反復の最近収集されたデータ130とを用いて構築される。ML実行モデル110の機能性は、アプリケーション実行パフォーマンスを適切な計算リソースに相関させることである。本明細書に示すように、ML実行モデル110は、リソース管理システム140と相互作用し、アプリケーション実行のためにリソースを効果的かつ効率的に1または複数のMLモデル150に誘導する。したがって、本明細書で示されるように、ML実行モデル110は、アプリケーションおよびリソース管理の結合プロファイルを生成または収集するためのコンポーネントとして機能する。

20

【0021】

図2を参照すると、MLプログラムの効率的な実行および計算リソースの有効な割り当ておよび使用をサポートするための人工知能プラットフォームを有するコンピュータ・システム200の概略図が示されている。図示されるように、サーバ210は、ネットワーク接続205を横断して、複数のコンピューティング・デバイス280、282、284、286、288および290と通信して提供される。サーバ210は、バス214を介してメモリ216と通信する、例えばプロセッサなどの処理ユニット212で構成される。プロセッサは、CPUまたはGPUであってもよい。サーバ210は、機械学習モデルを、アプリケーション実行およびリソース使用の結合プロファイルとしてトレーニングするためのコラボレーションをサポートする人工知能(AI)プラットフォーム250を有して示される。コンピューティング・デバイス280、282、284、286、288および290は、1または複数の有線もしくは無線またはその両方のデータ通信リンクを介して互いに、および他のデバイスまたはコンポーネントと通信することができ、ここで、各通信リンクは、1または複数の配線、ルータ、スイッチ、送信機、受信機などを含んでもよい。このネットワーク化された配置において、サーバ210およびネットワーク接続205は、通信検出、認識および解決を可能にする。サーバ210の他の実施形態は、本明細書で示されているもの以外のコンポーネント、システム、サブシステムもしくはデバイスまたはその組み合わせと共に使用されてもよい。

30

40

【0022】

AIプラットフォーム250は、種々のソースから入力202を受け取るように構成されて本明細書に示される。例えば、AIプラットフォーム250は、ネットワーク205からの入力を受信し、本明細書においてコーパスまたは知識ベースとも参照されるデータソース270を利用して、出力または応答内容を生成してもよい。AIプラットフォーム250は、機械学習コラボレーションをサポートおよび可能とし、アプリケーション・ベ

50

ースのプロファイリングを使用した機械学習ワークロードを弾力的に実行するツールとともに提供される。

【0023】

A Iプラットフォーム250は、アプリケーション実行、並びにリソース割り当ておよび管理の間のコラボレーションを可能にするためおよびサポートするためのプラットフォームとして機能する。一実施形態においては、A Iプラットフォーム250は、ネットワーク205を横断した1または複数のコンピューティング・デバイス280～290のような、サーバ210に動作可能に結合されたメンバに応答出力を通信する。

【0024】

A Iプラットフォーム250は、プロファイル・マネージャ252と、機械学習（ML）マネージャ254と、ディレクタ256と、リソース・マネージャ258とを含む、ニューラル・モデル・コラボレーションをサポートするためのいくつかのツールを有するように本明細書において示されている。プロファイル・マネージャ252は、MLアプリケーション実行およびリソース使用から構成される結合プロファイル（joint profile）を生成または収集するように機能する。結合プロファイルは、MLアプリケーション実行の詳細とML実行に対応するリソース使用データとを組み合わせる。MLアプリケーション実行の詳細の例には、これらに限定されるものではないが、バッチサイズ、学習率、反復回数、モデル精度などが含まれる。リソースの例には、これらに限定されるものではないが、利用されたCPUの数量、CPUタイプ、利用されたGPUの数量、GPUタイプなどが含まれる。プロファイル・マネージャ252は、MLプロファイルからの信号を監視または受信し、対応するMLモデルおよび一実施形態においてはMLモデルの一部であるMLルーチンによるリソース使用を収集する。より詳細には、MLモデルの構築またはトレーニングおよびMLモデル内の対応するMLルーチンは、ルーチンの機能性および対応する処理をサポートするために、物理リソースおよび仮想リソースを含む計算リソースの使用を必要とすることが当該技術分野において理解される。プロファイル・マネージャ252は、対応するMLモデルおよびMLモデルの一部であるMLルーチンから、これらのリソースおよび関連するリソース消費に関連するデータを収集する。収集されたデータは、後続のMLアプリケーション実行のためのリソース割り当て、つまり、後続のアプリケーションが、現在のワークロードと、比較する際に同一であるか、増加されるかまたは減少されるか、あるいは、パラメータのセットまたは構成が異なるかを予測するために使用される。本明細書に示されるように、プロファイル・マネージャ252は、本明細書においてコールバック関数A252Aとして示される少なくとも1つのコールバックを利用して、1または複数の関連するパラメータでアプリケーション実行をプロファイルし、リソース使用情報をプロファイルする。一実施形態においては、リソース使用情報のプロファイルの作成は、アプリケーションの実行をサポートするために使用されまたは必要とされる計算リソースの識別であり、1つの実施形態においては、アウトラインである。したがって、プロファイル・マネージャ252は、そのパラメータのセットでのアプリケーション実行に沿った使用される計算リソースおよび使用の特性を捕捉するように機能する。

【0025】

本明細書で示されるように、サーバ210は、データソース270に動作可能に結合される。データソース270は、MLアプリケーションに由来するMLモデルのライブラリ272と有して本明細書に示されている。ただ1つのライブラリが示されるが、データソース270は、複数のライブラリで構成されてもよく、ライブラリ272は、例示のためのものであることが理解される。ライブラリ272は、MLモデルA274AおよびMLモデルB274B含むMLモデルを有して示されている。MLモデルの数量は、例示の目的のためであり、限定的であるとみなすべきではない。一実施形態においては、MLモデルは、主題または機能性に基づいてライブラリにおいて編成される。MLモデルの各々は、本明細書において結合プロファイルとも参照される、動作可能に結合されたまたは関連付けられたプロファイルを有して示される。より具体的には、MLモデルA274Aが、プロファイルA276Aとともに示され、MLモデルB274BがプロファイルB276

10

20

30

40

50

Bとともに示される。ライブラリ272におけるMLモデル274Aおよび274Bは、必ずしも1つの特定のプロファイルと結合しない。MLモデルは、複数の結合プロファイルを用いて構築されてもよい。例えば、図示されているように、MLモデルN274Nは、例えば、プロファイルC276C、プロファイルD276DおよびプロファイルN276Nの組み合わせなど、プロファイルの組み合わせとして示される。各プロファイルは、対応するまたは動作可能なように結合されたモデルの実行と、動作可能に結合されたモデルの実行をサポートするためのリソース使用または要件とを組み合わせる。例えば、一実施形態においては、MLモデルは、同じパラメータを有するが、計算リソースの異なるセットで構築されてもよい。結果として得られるモデルは同じであるが、最終的にモデルが収束するタイミングは、リソースの制約に起因して異なるであろう。

10

【0026】

本明細書で示されているMLマネージャ254は、プロファイル・マネージャ252に動作可能に結合され、プロファイル、例えば276Aおよび276Bから、1または複数の特徴およびシングネチャを識別するように機能する。特徴およびシングネチャの例は、機械学習フレームワーク、学習バッチサイズ、割り当てられたCPUの数、使用されるメモリのサイズ、各反復についての実行時間またはアルゴリズム収束時間である。MLマネージャ254は、動作可能に結合されたML実行モデル254Aとともに本明細書において示されている。より具体的には、モデル254Aは、ML技術を使用してMLアプリケーション実行のための計算リソース使用を予測するMLモデルである。MLマネージャ254は、モデル254Aを構築し、管理し、モデル254Aを実行して、対応するプロファイル、例えば276A、からの1または複数の特徴およびシングネチャを利用することによって、対応するMLモデル、例えば274Aに対する計算リソースの要件を評価し、かつ、予測する。一実施形態においては、結合プロファイルの生成およびMLモデル254Aの構築のための特徴および特性、例えばシングネチャなどの識別は、ユーザ入力を用いて導出される。したがって、MLマネージャ254は、モデル274Aおよび274Bなどの、他のMLモデルの機能および処理をモデル化するMLモデルを作成するように機能して、MLモデル処理をサポートするための計算リソース割り当てを容易にする。

20

【0027】

ディレクタ256は、ML実行モデル254Aを適用するように機能し、モデルからの出力254Bを利用して、その後のアプリケーション実行への指令を提供する。出力254Bは、1または複数のMLアプリケーションの効率的かつ効果的な実行をサポートするのに必要な計算リソースの形態である。リソースの割り当ては、弾力的、例えば動的であり、実行中に変更される可能性がある。ディレクタ256は、ML実行モデル254Aに基づく出力254Bを利用して、アプリケーション実行をサポートするためにリソース管理コンポーネントから1または複数の計算リソースを弾力的に割り当て、また、要求する。本明細書に示すように、リソース・マネージャ258は、ディレクタ256に動作可能に結合されるように示される。一実施形態においては、出力254Bは、リソース・マネージャ258に、アプリケーション実行をサポートするための計算リソースおよびリソースの指定を示す指令とともに転送される。本明細書で説明されるように、リソース割り当ては、弾力的、例えば、非静的であるとして定義される。同様に、出力254Bに対応する指令は、リソース割り当てアクションの形式である。ML実行モデル254Aは、リソース割り当てアクション、例えば、指令、を実行して、アプリケーション処理をサポートするために利用可能なリソースを要求する。リソース割り当てアクションは、従前のアプリケーション処理および実行に対してリソースを削減または拡張してもよく、または、1つの実施形態においては、リソース割り当てアクションは、リソース要求を維持してもよい。同様に、要求されたリソースは、対応する属性を有する。リソース割り当てアクションは、アプリケーション処理および実行をサポートするリソース属性を識別し、これは、リソース割り当てに類似して弾力的である。したがって、ディレクタ256は、MLモデル254Aからの出力をサポートおよび処理するように機能する。

30

40

【0028】

50

反復は、MLアプリケーションで使用される用語であり、アルゴリズムのパラメータが更新される回数を示す。対応するニューラル・ネットワークのトレーニングは、複数回の反復を必要とする。本明細書に記載される機能性に関して、計算リソース割り当ては、ML実行モデルの要件およびリソースの利用可能性に基づいて、反復にわたって変化する可能性がある。ML実行モデル254Aは、MLモデルの動的特性および反復に応答し、対応する出力254Bは、この動的な性質を反映する。ディレクタ256は、MLアプリケーションの反復にわたって計算リソースの割り当てを変更してもよい。一実施形態においては、リソース割り当ての変更は、アプリケーション実行パターン、リソース使用パターンまたは両者の組み合わせに基づいてもよい。リソース割り当てに対する任意の変更が、MLモデル反復にわたる出力254Bおよび対応するリソース割り当てアクション、例えば、指令において反映される。

10

【0029】

いくつかの例示的な実施形態においては、サーバ210は、ニューヨーク州アーモンクのインターナショナル・ビジネス・マシーンス・コーポレーションから入手可能なIBM（登録商標）Watson（登録商標）システムであってもよく、以下に説明する例示的な実施形態のメカニズムを用いて拡張されてもよい。本明細書で示されかつ説明されるシステムは、反復的な最適化アルゴリズムに基づく機械学習を実装するためのツールを含む。ツールは、アプリケーション・ベースのプロファイリングを使用してMLワークロードを弾力的に実行することを可能にし、それによって、MLアプリケーションおよびそれらに対応するニューラル・ネットワークに、またモデル構築またはトレーニングの複数回の反復を横断して、計算リソースを効果的かつ効率的に割り当てる。

20

【0030】

プロファイル・マネージャ252、MLマネージャ254、ディレクタ256およびリソース・マネージャ258（以下、総称してAIツールまたはAIプラットフォーム・ツールと参照する）は、サーバ210のAIプラットフォーム250内で具体化または統合されるものとして示される。AIツールは、ネットワーク205越しにサーバ210に接続された別個のコンピューティング・システム（例えば、290）で実装してもよい。具体化される場所では、AIツールは、反復的なやり方で、MLアプリケーション実行モデリングおよび対応するリソース割り当ての弾力性をサポートし、かつ、可能とするように機能する。ML実行モデル254からの出力254Aは、リソース・マネージャ258に転送される実行可能な指令の形態で、1または複数のモデル274Aおよび274Bによる実行をサポートするための計算リソースを指示また指定してもよい。

30

【0031】

AIプラットフォーム250を利用できる情報ハンドリング・システムのタイプとしては、ハンドヘルド・コンピュータ／携帯電話280のような小型のハンドヘルド・デバイスから、メインフレーム・コンピュータ282のような大型のメインフレーム・システムまで及ぶ。ハンドヘルド・コンピュータ280の例には、パーソナル・デジタル・アシスタント（PDA）、および、MP4プレーヤやポータブル・テレビやコンパクトディスク・プレーヤのようなパーソナル・エンターテインメント・デバイスが含まれる。情報ハンドリング・システムの他の例には、ペンまたはタブレット・コンピュータ284、ラップトップまたはノートブック・コンピュータ286、パーソナル・コンピュータ・システム288およびサーバ290が含まれる。図示されるように、種々の情報ハンドリング・システムは、コンピュータ・ネットワーク205を使用して一緒にネットワーク化されてもよい。種々の情報処理システムを相互接続するために使用することができるコンピュータ・ネットワーク205のタイプには、ローカル・エリア・ネットワーク（LAN）、無線ローカル・エリア・ネットワーク（WLAN）、インターネット、公衆交換電話ネットワーク（PSTN）、他の無線ネットワーク、その他、情報処理システムを相互接続するために使用することができる任意のネットワークトポロジーが含まれる。情報ハンドリング・システムの多くは、ハード・ドライブもしくは不揮発性メモリまたはその両方のような不揮発性データ・ストアを含む。情報ハンドリング・システムのいくつかは、別個の不揮発

40

50

性データ・ストア（例えば、サーバ290が不揮発性データ・ストア290Aを利用し、メインフレーム・コンピュータ282が不揮発性データ・ストア282Aを利用する）を使用してもよい。不揮発性データ・ストア282Aは、種々の情報ハンドリング・システムの外部のコンポーネントであってもよいし、情報ハンドリング・システムのうちの1つの内部のものであってもよい。

【0032】

AIプラットフォーム250をサポートするために使用される情報ハンドリング・システムは、多くの形態をとってもよく、そのいくつかが図2に示される。例えば、情報ハンドリング・システムは、デスクトップ、サーバ、ポータブル、ラップトップ、ノートブックまたは他のフォーム・ファクタのコンピュータまたはデータ処理システムの形態をとってもよい。加えて、情報ハンドリング・システムは、パーソナル・デジタル・アシスタント（PDA）、ゲーム・デバイス、ATMマシン、ポータブル電話デバイス、通信デバイスまたはプロセッサおよびメモリを含む他のデバイスなどの他のフォーム・ファクタをとってもよい。

【0033】

アプリケーション・プログラム・インタフェース（API）は、当該技術分野では、2以上のアプリケーション間のソフトウェア中間手段として理解される。図2に示され、説明されたAIプラットフォーム250に関して、ツール252～258の1つまたは複数および関連する機能性をサポートするために1または複数のAPIを利用することができる。図3を参照すると、ツール252～258およびそれらの関連付けられるAPIを示すブロック図300が提供される。図示されるように、APIプラットフォーム305内に、API₀₃₁₂に関連付けられるプロファイル・マネージャ352と、API₁₃₂₂に関連付けられるMLマネージャ354と、API₂₃₃₂に関連付けられるディレクタ356と、API₃₃₄₂に関連付けられるリソース・マネージャ358とを含んで、複数のツールが具現化されている。APIの各々は、1または複数の言語およびインタフェース仕様で実装されてもよい。API₀₃₁₂は、アプリケーション実行の詳細およびアプリケーション実行リソース使用の組み合わせとして結合プロファイルを生成するための機能サポートを提供し、API₁₃₂₂は、MLアプリケーション実行および計算リソース使用を特徴付け、また識別することを対象にML実行モデルを構築および維持するための機能サポートを提供し、API₂₃₃₂は、弾力的にアプリケーションおよびML実行モデルからの出力に基づいて計算リソースを要求するための機能サポートを提供し、API₃₃₄₂は、モデルおよびそれらの対応するニューラル・ネットワークのうちの1つまたは複数による実行をサポートするために計算リソースを指示および指定するための機能的サポートを提供する。

【0034】

図示されるように、API₃₁₂、322、332および342の各々は、APIオーケストレータ360（さもないならばオーケストレーション層として知られている）に動作可能に結合されており、これは、当該技術分野では、個別のAPIを透過的に一緒にスレッド化する抽象化層として機能するものと理解されている。一実施形態においては、別個のAPIの機能性は、結合または組み合わせられてもよい。それゆえ、本明細書に示されるAPIの構成は、限定するものと考えるべきではない。したがって、本明細書に示されるように、ツールの機能性は、それらのそれぞれのAPIによって具現化またはサポートされてもよい。別の実施形態においては、別個のAPIを分散計算リソースにおいて処理し、弾力性、スケーラビリティおよび可用性を達成することができる。

【0035】

図4を参照すると、MLモデリングのためのプロセスおよび計算リソース割り当ての弾力性を示すためにフロー図400が提供される。図示されるように、アプリケーションを実行するための要求は、本明細書では実行要求として知られており、アプリケーション特有およびシステム・プロファイリング特有のパラメータとともに受信される（402）。実行指令が、ステップ402からの受信された要求に基づいて、およびモデルのアプリケ

10

20

30

40

50

ーション・パラメータおよび履歴処理を必然的に伴うML実行モデラー450に基づいて、生成される(404)。一実施形態においては、履歴実行が未知であるか、利用可能ではない場合、ユーザ指定入力のみが利用されてもよい。可能性のある実行指令には、特に限定されるものではないが、メモリ使用の設定限界、および、使用のためのリソースの割り当てが含まれる。メモリ使用に関しては、1以上の以前の反復または1または複数の以前の実行からの履歴データに基づいて、アプリケーションのインスタンスに対して制限が設定されてもよい。リソースの利用に関しては、以前の履歴データには、トレーニングモデルの同一の調整パラメータであるが、異なるコンピューティング・プラットフォーム、例えば、異なるCPUの数量、CPUタイプ、異なるGPUの数量、GPUタイプおよびNVLink(有線ベースの通信プロトコル)接続性上でのリソース利用に関する情報を包含し得る。これらの指令は、例示の目的のためであり、限定であるとみなすべきではない。よって、指令は、ML実行モデルおよび実行要求に対応するパラメータに基づいて生成される。

【0036】

ステップ404で生成された指令が用いられて、リソース管理と相互作用して実行要求をサポートするための所望のリソースを取得し、また、要求パラメータの範囲内でリソース割り当てを実行する(406)。より具体的には、ステップ404で得られる指令は、計算リソースを割り当てるために使用される。例えば、一実施形態においては、割り当ては、CPUコアの数量、CPUメモリ、GPUタイプおよびGPUの数量に対する潜在的な動的拡張のための弾力的な実行のための指定されたサイズのコンテナに向けられてもよい。同様に、一実施形態においては、割り当ては、可能である場合、あるいは望ましい場合には、同一マシン上のコンテナをパッケージングすることに向けられてもよい。別の実施形態においては、割り当ては、GPU共有に向けられてもよい。リソースが割り当てられた後、1または複数のMLアプリケーションが実行され(408)、実行されたMLアプリケーションのプロファイル情報が収集される(410)。一実施形態においては、収集は、MLアプリケーションの詳細およびリソース使用情報に関するコールバック関数を使用する。収集されたMLアプリケーション実行データは、本明細書では、動的アプリケーション情報と参照され、ML実行モデラー450に通信される。ステップ410およびプロファイル情報収集に続いて、終了閾値に達したかどうか判定される(412)。ステップ412での判定に対する否定的な応答に続いて、ステップ404へ戻り、肯定的な応答は、プロファイル収集および展開処理を終了させる。

【0037】

ML実行モデラー450は、アプリケーション実行からの入力およびリソース管理との相互作用を受信する。ML実行モデラー450は、静的入力データ460および動的入力データ470の両方を受信する。静的入力データは、入力データ情報462、調整パラメータ範囲464およびCPU、GPUアーキテクチャ、メモリおよびストレージ466などの静的計算情報の形態である。動的入力データ470は、パラメータのセッティング472およびリソース使用情報474の形態である。したがって、ML実行モデラー450は、アプリケーション実行情報およびリソース情報を利用して、最適なリソースを識別し、MLアプリケーションをサポートするための要件の予測を構築する。

【0038】

図4に示されるように、コールバック関数が、実行された、または実行しているMLアプリケーションによって使用され、MLアプリケーション・データおよびMLアプリケーション・リソース使用を識別し、これによって、MLワークロードおよびリソース使用の結合プロファイルを生成する。一実施形態においては、結合プロファイルは、MLアプリケーション特有の特徴またはシグネチャおよびリソース使用を有する、タイムスタンプされたデータを含んでもよい。ML実行モデラーにおいて構築されたML実行モデルは、アプリケーション実行詳細を包含する、生成された結合プロファイルを利用する。より具体的には、ML実行モデルは、結合プロファイルから更新を受信し、適用するという点で動的である。ML実行モデルは、以前の実行からのデータを使用して構築されるか、または

、同一の実行における以前の反復から最近収集されたデータから構築される。結合プロファイルに含まれる特徴およびシグネチャ並びに結合プロファイル内で動的に適用される更新は、その後の反復のアプリケーション実行に対する指令を提供し、アプリケーション実行パフォーマンスを、適切な、一実施形態においては異種の、計算リソースと関連させる。一実施形態においては、実行されたまたは実行しているMLアプリケーションからのコールバック関数をサポートするために、APIのセットが採用される。APIのセットは、任意のMLフレームワークにおけるアプリケーションから呼び出し可能である。したがって、結合プロファイルは、ML実行モデラーにおけるML実行モデルを構築するための両方で使用され、一度結合ファイルが構築されると、ML実行モデラーによって適用されて、リソース管理コンポーネントから、種々の反復でMLアプリケーションを効率的に実行するためにリソースを有効に要求する。

10

【0039】

以下は、アプリケーションがどのように実行されているかの詳細を捉えるためのAPIからのトレース出力の一例である。

【数1】

```
2018-05-17 02:51:34.288301 Running on machine with 28 cores and architecture type: x86_64

2018-05-17 02:51:34.288305 Running on CPU cores - [3] for this process: psutil.Process(pid=4954, name='python',
started='22:51:26') CUDA_VISIBLE_DEVICES: [7]; Cuda V.: 384.81; Device [7] Type K80

2018-05-17 02:51:34.288301 batch size 1024

2018-05-17 02:51:34.288352 epoch 1 starts

2018-05-17 02:51:35.959277 CPU utilization:                68.4 % x 1 cores

2018-05-17 02:51:35.959277 CPU memory utilization:        1562587136(RSS) 567708688384(VMS)

2018-05-17 02:51:35.959277 GPU[7] total memory:          11995578368

2018-05-17 02:51:35.959277 GPU[7] used memory:           11485577216

2018-05-17 02:51:35.959277 GPU[7] utilization:           99 %

2018-05-17 02:51:35.959277 GPU[7] memory utilization:     27 %

2018-05-17 02:51:35.960202 Elapse time for epoch 1 -      0:00:01.671845; accuracy - 0.6077724358974359
```

20

30

図示されているように、APIは、バッチサイズ、入力情報、データ・タイプ、メモリ利用およびMLアプリケーションの反復実行のために費やした時間を含む、アプリケーションがどのように実行されるかの詳細を捕捉する。APIは、時間駆動またはロジック駆動であってもよい。したがって、この例で示されるAPIは、MLアプリケーション実行データおよびリソース使用を捕捉し、後続のMLアプリケーション反復のためのMLアプリケーション指令のために、キャプチャされたデータをML実行モデラーに通信する。

【0040】

機能ツール252～258の側面およびそれらの関連する機能性は、単一の位置におけるコンピュータ・システム／サーバ内で具現化されてもよく、または、一実施形態においては、計算リソースを共有するクラウド・ベースのシステムで構成されてもよい。図5を参照すると、ブロック図500は、図1～図4に関連して上述した処理を実装するための、クラウド・ベースのサポート・システムと通信する、ホスト502と参照されるコンピュータ・システム／サーバ502の一例を示すものである。ホスト502は、多数の他の汎用的なまたは特定用途のコンピューティング・システム環境または構成とともに動作可能である。ホスト502と共に使用するのに適した周知のコンピューティング・システム、環境もしくは構成またはその組み合わせの例には、これらに限定されるものではないが、パーソナル・コンピュータ・システム、サーバ・コンピュータ・システム、シン・クラ

40

50

イアント、シック・クライアント、ハンドヘルドまたはラップトップ・デバイス、マルチプロセッサ・システム、マイクロプロセッサ・ベースのシステム、セット・トップ・ボックス、プログラマブル・コンシューマ・エレクトロニクス、ネットワークPC、ミニコンピュータ・システム、メインフレーム・コンピュータ・システム、および、上記のシステム、デバイス、およびそれらの等価物のいずれかを含むファイル・システム（例えば、分散型ストレージ環境および分散型クラウド・コンピューティング環境）を含む。

【0041】

ホスト502は、コンピュータ・システムによって実行されるプログラム・モジュールのようなコンピュータ・システム実行可能な命令の一般的な文脈において説明され得る。概して、プログラム・モジュールは、ルーチン、特定のタスクを実行するか、または、特定の抽象データ・タイプを実装する、プログラム、オブジェクト、コンポーネント、ロジック、データ構造などを含む。ホスト502は、通信ネットワークを介してリンクされた遠隔の処理デバイスによってタスクが実行される、分散型クラウド・コンピューティング環境580において実現してもよい。分散型クラウド・コンピューティング環境では、プログラム・モジュールは、メモリ・ストレージ・デバイスを含む、ローカルおよびリモートのコンピュータ・システム・ストレージ媒体の両方に配置されてもよい。

【0042】

図5に示すように、ホスト502は、汎用コンピューティング・デバイスの形態で示される。ホスト502のコンポーネントは、これらに限定されるものではないが、例えばハードウェア・プロセッサなどのプロセッサまたは処理ユニット504と、システム・メモリ506と、システム・メモリ506を含む種々のシステム・コンポーネントをプロセッサ504に結合するバス508とを含んでもよい。バス508は、メモリバスまたはメモリコントローラ、周辺バス、アクセラレーテッド・グラフィックス・ポート、種々のバス・アーキテクチャの任意のものを使用するプロセッサまたはローカル・バスを含むいくつかのタイプのバス構造のうちの1以上を表す。例として、このようなアーキテクチャには、限定されるものではないが、インダストリ・スタンダード・アーキテクチャ（ISA）バス、マイクロ・チャンネル・アーキテクチャ（MCA）バス、拡張ISA（EISA）バス、ビデオ・エレクトロニクス・スタンダード・アソシエーション（VESA）ローカル・バスおよびペリフェラル・コンポーネント・インターコネクト（PCI）バスが含まれる。ホスト502は、典型的には、種々のコンピュータ・システム可読媒体を含む。このような媒体は、ホスト502によってアクセス可能な任意の利用可能な媒体であってよく、それは、揮発性媒体と不揮発性媒体の両方、リムーバブル媒体と非リムーバブル媒体の両方を含む。

【0043】

メモリ506は、ランダム・アクセス・メモリ（RAM）530もしくはキャッシュ・メモリ532またはその両方などの揮発性メモリの形式のコンピュータ・システム可読媒体を含むことができる。単に例として、ストレージ・システム534が、非リムーバブル、不揮発性の磁気媒体（図示しないが、通常「ハード・ドライブ」と呼ばれる）に対する読み取りと書き込みを行うために提供されてもよい。図示しないが、リムーバブル、不揮発性の磁気ディスク（たとえば、「フロッピー（登録商標）ディスク」）に対する読み取りと書き込みを行うための磁気ディスク・ドライブおよびCD-ROM、DVD-ROMまたは他の光媒体などのリムーバブル、不揮発性の光ディスクに対する読み取りと書き込みを行うための光ディスク・ドライブが提供されてもよい。そのような例において、各々は、1以上のデータ媒体インタフェースによってバス508に接続されてもよい。

【0044】

プログラム・モジュール542のセット（少なくとも1つ）を有するプログラム／ユーティリティ540が、限定ではなく例として、メモリ506に記憶されてもよく、オペレーティング・システム、1以上のアプリケーション・プログラム、他のプログラム・モジュールおよびプログラム・データも同様である。オペレーティング・システム、1以上のアプリケーション・プログラム、他のプログラム・モジュールおよびプログラム・データ

10

20

30

40

50

の各々またはこれらの任意の組み合わせは、ネットワーク環境の実装を含んでよい。プログラム・モジュール542は、概して、動的なコミュニケーション評価の質問識別と処理（dynamically communication evaluation interrogatory identification and processing）の実施形態の機能もしくは方法論またはその両方を実行する。例えば、プログラム・モジュール542のセットは、図1に説明されるツール252～258を含んでもよい。

【0045】

ホスト502は、キーボード、ポインティング・デバイスなどの外部デバイス514；ディスプレイ524；ユーザがホスト502と対話できるようにする1以上のデバイス；もしくはホスト502が1以上の他のコンピューティング・デバイスと通信することを可能にする任意のデバイス（たとえば、ネットワーク・カード、モデムなど）またはその組み合わせと通信してもよい。このような通信は、入力／出力（I／O）インタフェース522を介して行うことができる。さらに、ホスト502は、ネットワーク・アダプタ520を介して、ローカル・エリア・ネットワーク（LAN）、一般的なワイド・エリア・ネットワーク（WAN）もしくはパブリック・ネットワーク（たとえば、インターネット）またはその組み合わせなどの1以上のネットワークと通信することができる。図示されるように、ネットワーク・アダプタ520は、バス508を介してホスト502の他のコンポーネントと通信する。一実施形態においては、分散ファイルシステム（図示せず）の複数のノードが、ホスト502とI／Oインタフェース522を介して、または、ネットワーク・アダプタ520を介して通信可能とされる。図示されていないが、他のハードウェア・コンポーネントもしくはソフトウェア・コンポーネントまたはその両方がホスト502とともに使用されてもよいことを理解されたい。具体例には、マイクロコード、デバイス・ドライバ、冗長処理ユニット、外部ディスク・ドライブ・アレイ、RAIDシステム、テープ・ドライブおよびデータ・アーカイブ・ストレージ・システムなどが含まれるが、これらに限定されない。

【0046】

この文書においては、「コンピュータ・プログラム媒体」、「コンピュータ使用可能媒体」、「コンピュータ可読媒体」という用語は、概して、RAM530を含むメイン・メモリ506、キャッシュ532、および、リムーバブル・ストレージ・ドライブおよびハードディスク・ドライブにインストールされたハード・ディスクなどのストレージ・システム534のような媒体を参照して使用される。

【0047】

コンピュータ・プログラム（コンピュータ制御ロジックと参照される）は、メモリ506に格納される。コンピュータ・プログラムは、また、ネットワーク・アダプタ520のような通信インタフェースを介して受信されてもよい。このようなコンピュータ・プログラムは、実行された場合、コンピュータ・システムが、本明細書で説明されるような本実施形態の機能を実行することを可能にする。特に、コンピュータ・プログラムは、実行された場合に、処理ユニット504が、コンピュータ・システムの機能を実行することを可能にする。したがって、このようなコンピュータ・プログラムは、コンピュータ・システムのコントローラを表す。

【0048】

コンピュータ可読ストレージ媒体は、命令実行デバイスによって使用するための命令を保持し格納する有形のデバイスであってよい。コンピュータ可読ストレージ媒体は、例えば、これに限定されるものではないが、電子的ストレージ・デバイス、磁気ストレージ・デバイス、光学ストレージ・デバイス、電磁気ストレージ・デバイス、半導体ストレージ・デバイスまたは上記の任意の適切な組み合わせであってよい。コンピュータ可読ストレージ媒体のより具体的な例示の非網羅的リストとしては、ポータブルなコンピュータ・ディスクケット、ハード・ディスク、ダイナミックまたはスタティック・ランダム・アクセス・メモリ（RAM）、リード・オンリー・メモリ（ROM）、消去可能プログラマブル・リード・オンリー・メモリ（EPROMまたはフラッシュ・メモリ）、磁気ストレージ・

デバイス、ポータブルなコンパクト・ディスク・リード・オンリー・メモリ（ＣＤ－ＲＯＭ）、デジタル・バーサタイル・ディスク（ＤＶＤ）、メモリースティック、フロッピー（登録商標）ディスク、パンチカードまたは記録された命令を有する溝内の隆起構造のような機械的エンコードされたデバイス、および上記の任意の適切な組み合わせが含まれる。コンピュータ可読ストレージ媒体は、本明細書で使用されるように、電波、自由伝搬する電磁波、導波路または他の伝送媒体を伝搬する電磁波（たとえば、ファイバ光ケーブルを通過する光パルス）または、ワイヤを通して伝送される電気信号のような、それ自体が一時的な信号として解釈されるものではない。

【００４９】

本明細書で説明されるコンピュータ可読プログラム命令は、コンピュータ可読ストレージ媒体からそれぞれのコンピュータ／処理デバイスに、または、外部コンピュータまたは外部ストレージ・デバイスに、例えばインターネット、ローカル・エリア・ネットワーク、ワイド・エリア・ネットワークもしくは無線ネットワークまたはその組み合わせといったネットワークを介してダウンロードすることができる。ネットワークは、銅伝送ケーブル、光伝送ファイバ、無線伝送、ルータ、ファイアウォール、スイッチ、ゲートウェイ・コンピュータもしくはエッジサーバまたはその組み合わせを含んでもよい。各コンピュータ／処理デバイスにおけるネットワーク・アダプタ・カードまたはネットワーク・インタフェースは、ネットワークからコンピュータ可読プログラム命令を受信し、コンピュータ可読プログラム命令を、それぞれのコンピューティング／処理デバイス内のコンピュータ可読ストレージ媒体に格納するために転送する。

【００５０】

本実施形態の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セットアーキテクチャ（ＩＳＡ）命令、機械語命令、機械依存命令、マイクロコード、ファームウェア命令、状態設定データ、または、１以上のプログラミング言語の任意の組み合わせで書かれたソース・コードあるいはオブジェクト・コードであってよく、１以上のプログラミング言語は、Ｊａｖａ（登録商標）、Ｓｍａｌｌｔａｌｋ（登録商標）、Ｃ＋＋またはこれらに類するものなどのオブジェクト指向言語、Ｃプログラミング言語または類似のプログラミング言語などの従来型の手続型言語を含む。コンピュータ可読プログラム命令は、スタンド・アローンのソフトウェア・パッケージとして、全体としてユーザのコンピュータ上で、部分的にユーザのコンピュータ上で、部分的にユーザのコンピュータ上かつ部分的に遠隔のコンピュータ上で、または、完全に遠隔のコンピュータまたはサーバ上、またはサーバのクラスタ上で実行されてもよい。後者のシナリオでは、遠隔のコンピュータは、ユーザのコンピュータに、ローカル・エリア・ネットワーク（ＬＡＮ）またはワイド・エリア・ネットワーク（ＷＡＮ）を含む任意のタイプのネットワークを通じて接続されてもよく、あるいは接続は、（例えば、インターネット・サービス・プロバイダを用いてインターネットを通じて）外部コンピュータになされてもよい。いくつかの実施形態においては、電気的回路は、例えば、プログラマブル・ロジック回路、フィールド・プログラマブル・ゲート・アレイ（ＦＰＧＡ）、またはプログラマブル・ロジック・アレイ（ＰＬＡ）を含むが、本実施形態の側面を実行するために、コンピュータ可読プログラム命令の状態情報を利用して、電気的回路を個別化することによって、コンピュータ可読プログラム命令を実行してもよい。

【００５１】

一実施形態においては、ホスト５０２は、クラウド・コンピューティング環境のノードである。当該技術分野において知られるように、クラウド・コンピューティングは、最小の管理労力またはサービス・プロバイダとの対話で迅速にプロビジョニングおよびリリースされ得る、構成可能な計算リソース（例えば、ネットワーク、ネットワーク帯域、サーバ、処理、メモリ、ストレージ、アプリケーション、仮想マシンおよびサービス）の共有プールへの便利なオンデマンドのネットワーク・アクセスを可能とする、サービス配布のモデルである。このクラウド・モデルは、少なくとも５つの特性、少なくとも３つのサービス・モデルおよび少なくとも４つのデプロイメント・モデルを含んでもよい。これらの

10

20

30

40

50

特性の例は、以下の通りである。

オンデマンド・セルフ・サービス：クラウド・コンシューマは、サービス・プロバイダとの人的な対話を必要とせず自動的に必要なだけ、サーバ時間およびネットワーク・ストレージなどのコンピュータ能力を一方的にプロビジョニングすることができる。

広帯域ネットワーク・アクセス：能力は、ネットワーク越しに利用可能であり、異種シン・クライアントまたはシック・クライアント・プラットフォーム（例えば、モバイルフォン、ラップトップ、PDA）による使用を促進する標準的なメカニズムを介して、アクセスされる。

リソース・プーリング：プロバイダの計算リソースは、マルチ・テナント・モデルを用いて複数のコンシューマに提供するためにプールされ、種々の物理的および仮想的リソースが需要に従って動的に割り当てられ、また、再割り当てられる。コンシューマは、一般的に、提供されるリソースの正確な場所を管理したり、知識を有したりせず、しかし、より高度な抽象レベル（例えば国、州、またはデータセンタ）にて場所を指定することが可能であるという意味で、場所の独立感がある。

迅速な弾力性：能力は、迅速かつ柔軟に、いくつかの場合では自動的に、プロビジョニングされて素早くスケール・アウトすることができ、また、迅速にリリースされて素早くスケール・インすることができる。コンシューマにとって、プロビジョニング利用可能な能力は、しばしば外面的には無制限のように見え、任意の時間に任意の量を購入することができる。

測量されたサービス：クラウドシステムは、サービスのタイプにとって適切なある抽象レベル（例えば、ストレージ、処理、帯域幅、アクティブ・ユーザ数）での計量能力を利用することによって、自動的にリソース使用を制御し、また最適化する。リソース使用量は、監視され、制御されおよび報告されて、利用サービスのプロバイダおよびコンシューマの双方に対する透明性を提供する。

【0052】

サービス・モデルは、以下の通りである。

ソフトウェア・アズ・ア・サービス（SaaS）：コンシューマに提供される能力は、クラウド・インフラストラクチャ上で稼働するプロバイダのアプリケーションを使用することである。アプリケーションは、ウェブ・ブラウザ（例えばウェブベースの電子メール）などのシン・クライアント・インタフェースを介して種々のクライアント・デバイスからアクセス可能である。コンシューマは、ネットワーク、サーバ、オペレーティング・システム、ストレージ、または、限定されたユーザ固有のアプリケーション構成設定の潜在的な例外を除いて個々のアプリケーション能力すらも含む下層のインフラストラクチャを管理または制御しない。

プラットフォーム・アズ・ア・サービス（PaaS）：コンシューマに提供される能力は、プロバイダによってサポートされるプログラミング言語およびツールを用いて作成された、コンシューマ作成または獲得のアプリケーションをクラウド・インフラストラクチャ上にデプロイすることである。コンシューマは、ネットワーク、サーバ、オペレーティング・システムまたはストレージを含む下層のクラウド・インフラストラクチャを管理または制御しないが、デプロイされたアプリケーションおよび場合によってはアプリケーション・ホスティング環境の構成への制御を有する。

インフラストラクチャ・アズ・ア・サービス（IaaS）：コンシューマに提供される能力は、処理、ストレージ、ネットワーク、および、コンシューマが、オペレーティング・システムおよびアプリケーションを含み得る任意のソフトウェアをデプロイし、稼働させることができる他の基本的な計算リソースを提供することである。コンシューマは、下層のクラウド・インフラストラクチャを管理または制御しないが、オペレーティング・システム、ストレージ、デプロイされたアプリケーションに対する制御、および、場合によっては、選択したネットワーキング・コンポーネント（例えば、ホストファイアウォール）の限定された制御を有する。

【0053】

デプロイメント・モデルは、以下の通りである。

プライベート・クラウド：クラウド・インフラストラクチャは、1つの組織のためだけに使用される。これは、組織または第三者によって管理されてもよく、オンプレミスまたはオフプレミスが存在し得る。

コミュニティ・クラウド：クラウド・インフラストラクチャは、いくつかの組織により共有され、共通の懸念（例えば、ミッション、セキュリティ要件、ポリシーおよびコンプライアンスに関する考慮事項）を有する特定のコミュニティをサポートする。これは、組織または第三者によって管理されてもよく、オンプレミスまたはオフプレミスが存在し得る。

パブリック・クラウド：クラウド・インフラストラクチャは、一般公衆、または、大きな業界団体が利用可能であり、クラウド・サービスを販売する組織によって所有される。

ハイブリッド・クラウド：クラウド・インフラストラクチャは、2以上のクラウド（プライベート、コミュニティまたはパブリック）の混成であり、これらのクラウドは、固有のエンティティのままであるが、しかし、データおよびアプリケーションのポータビリティを可能とする標準化されたまたは独自の技術（例えばクラウド間の負荷分散のためのクラウド・バースティング）によって結合される。

クラウド・コンピューティング環境は、ステートレス性、低結合、モジュール性および意味論的な相互運用性に重点を置いたサービス指向である。クラウド・コンピューティングの核心は、相互接続された複数のノードのネットワークを含むインフラストラクチャである。

【0054】

ここで、図6を参照すると、例示的なクラウド・コンピューティング・ネットワーク600が示されている。図示されるように、クラウド・コンピューティング・ネットワーク600は、1以上のクラウド・コンピューティング・ノード610を有するクラウド・コンピューティング環境650を含み、1以上のクラウド・コンピューティング・ノード610と、例えば、クラウド・コンシューマによって使用されるローカル・コンピューティング・デバイスが通信してもよい。これらのローカル・コンピューティング・デバイスの例には、これらに限定されないが、パーソナル・デジタル・アシスタント（PDA）または携帯電話654A、デスクトップ・コンピュータ654B、ラップトップ・コンピュータ654Cもしくは自動車コンピュータ・システム654Nまたはその組み合わせが含まれる。ノード610内の個々のノードは、さらに、互いに通信してもよい。これらは、プライベート、コミュニティ、パブリックもしくはハイブリッド・クラウドなど上述したような、またはその組み合わせなどの1以上のネットワークにおいて、物理的にまたは仮想的にグループ化（図示しない）されてもよい。これは、クラウド・コンピューティング環境600が、インフラストラクチャ、プラットフォームもしくはソフトウェアまたはその組み合わせをサービスとして提供することを可能とし、これらについては、クラウド・コンシューマは、リソースをローカル・コンピューティング・デバイス上で維持する必要がない。図6に示されるコンピューティング・デバイス654A～654Nのタイプは、説明する目的のみであり、クラウド・コンピューティング環境650が、任意のタイプのネットワーク、ネットワークアドレス可能な接続（例えば、ウェブ・ブラウザを使用して）またはその両方を介して、任意のタイプのコンピュータ化されたデバイスと通信することができることが理解される。

【0055】

ここで、図7を参照すると、図6のクラウド・コンピューティング・ネットワークによって提供される機能抽象層のセット700が示される。図7に示すコンポーネント、層および機能が、説明する目的のみであり、本実施形態は、これらに限定されないことを事前に理解されるべきである。図示するように、ハードウェアおよびソフトウェア層710、仮想化層720、管理層730およびワークロード層740の層および対応する機能が提供される。

【0056】

10

20

30

40

50

ハードウェアおよびソフトウェア層 710 は、ハードウェアおよびソフトウェア・コンポーネントを含む。ハードウェア・コンポーネントの例には、メインフレーム、一例においては、IBM（登録商標）zSeries（登録商標）System、RISC（縮約命令セットコンピュータ）アーキテクチャに基づくサーバ、一例においては IBM（登録商標）pSeries（登録商標）System、IBM（登録商標）xSeries（登録商標）System、IBM（登録商標）BladeCenter（登録商標）System、ストレージ・デバイスおよびネットワークおよびネットワークング・コンポーネントを含む。ソフトウェア・コンポーネントの例には、ネットワーク・アプリケーション・サーバ・ソフトウェア、一例においては IBM（登録商標）WebSphere（登録商標）アプリケーション・サーバ・ソフトウェア、および、データベース・ソフトウェア、一例においては、IBM（登録商標）DB2（登録商標）データベース・ソフトウェアを含む（IBM、zSeries、pSeries、xSeries、BladeCenter、WebSphere および DB2 は、世界の多くの管轄区域において登録された国際的な・ビジネス・マシナイズ・コーポレーションの登録商標である）。

【0057】

10

仮想化層 720 は、そこから仮想化サーバ、仮想化ストレージ、バーチャル・プライベート・ネットワークを含む仮想化ネットワーク、仮想化アプリケーションおよびオペレーティング・システム、および仮想クライアントなどの仮想化エンティティの例が提供される、抽象化層を提供する。

【0058】

20

一例においては、管理層 730 は、リソース・プロビジョニング、メータリングおよびプライシング、ユーザ・ポータル、サービス・レベル・マネジメント、SLA の計画と履行の機能を提供してもよい。リソース・プロビジョニングは、クラウド・コンピューティング環境内でタスクを実行するために利用される計算リソースおよび他のリソースの動的な調達を提供する。メータリングおよびプライシングは、リソースがクラウド・コンピューティング環境内で利用されるコストの追跡およびこれらのリソースの消費に対する請求またはインボイスの送付を提供する。一例においては、これらのリソースは、アプリケーション・ソフトウェアのライセンスを含んでもよい。セキュリティは、クラウド・コンシューマおよびタスクについての本人確認、並びに、データおよび他のリソースに対する保護を提供する。ユーザ・ポータルは、コンシューマおよびシステム管理者に対しクラウド・コンピューティング環境へのアクセスを提供する。サービス・レベル・マネジメントは、要求されるサービス・レイヤを満たすようにクラウド・計算リソースの割り当ておよび管理を提供する。サービス・レベル合意（SLA）の計画と履行は、SLA に従って、将来の要求が予測されるクラウド・コンピューティング・リソースの事前配置および調達を提供する。

30

【0059】

ワークロード層 740 は、クラウド・コンピューティング環境が利用される機能性の例を提供する。この層から提供されるワークロードおよび機能の例には、これに限定されるものではないが、マッピングおよびナビゲーション、ソフトウェア開発およびライフサイクル管理、仮想クラスルーム教育配信、データ・アナリティクス処理、トランザクション処理、および、ML 実行モデリングが含まれる。

40

【0060】

本明細書では、自然言語入力を評価し、対応する通信における質問を検出し、検出された質問を応答もしくはサポート内容またはその両方で解決するためのシステム、方法、装置およびコンピュータ・プログラム製品が開示されることが理解される。

【0061】

本実施形態の特定の実施形態が図示され、説明されたが、当業者には、本明細書の教示に基づいて、実施形態およびそのより広い側面から逸脱することなく、変更および修正を行うことができることが明らかであろう。したがって、添付の特許請求の範囲は、本実施形態の真の精神および範囲内にあるような全てのそのような変更および修正をその範囲内

50

に包含するものとする。さらに、これらの実施形態は、特許請求の範囲によってのみ定義されることが理解されるべきである。特定の数の導入された請求項の要素が意図される場合には、そのような意図は、請求項において明示的に記述され、そのような記載が欠如する場合、そのような限定がないということが当業者には理解されるであろう。非限定的な例について、理解を助けるために、以下の特許請求の範囲は、請求項の要素を導入するために、「少なくとも1つ」および「1または複数の」というフレーズの使用を含む。しかしながら、そのようなフレーズの使用は、「1または複数の」または「少なくとも1つ」というフレーズおよび「a」または「an」といった不定詞を含む同一クレームでさえも、不定詞「a」または「an」による請求項の要素の導入が、そのような要素を1つだけを含む実施形態に限定することを暗に示すものと解釈されるべきではなく、特許請求の範囲中の定冠詞の使用についても当てはまる。

10

【0062】

本実施形態は、システム、方法もしくはコンピュータ・プログラム製品またはその組み合わせであってもよい。加えて、本実施形態の選択された側面は、全体としてハードウェアの実施形態、全体としてソフトウェアの実施形態（ファームウェア、常駐ソフトウェア、マイクロコードなどを含む）、または本明細書では全て一般的に「回路」、「モジュール」または「システム」と参照されるソフトウェアもしくはハードウェアまたはその両方の態様を組み合わせた実施形態の形式をとることができる。さらに、本実施形態の側面は、プロセッサに本実施形態の側面を実行させるようにコンピュータ可読プログラム命令をその上に有するコンピュータ可読ストレージ媒体（または複数の媒体）に具現化されたコンピュータ・プログラム製品の形態をとることができる。したがって、開示されたシステム、方法もしくはコンピュータ・プログラム製品またはその組み合わせは、効率的で効果的なMLアプリケーションのパフォーマンスおよびリソース管理を含む、MLアプリケーション実行およびリソース使用をモデル化するための人工知能プラットフォームの機能および動作を改善するように動作する。

20

【0063】

コンピュータ可読ストレージ媒体は、命令実行デバイスによって使用するための命令を保持し格納する有形のデバイスであってもよい。コンピュータ可読ストレージ媒体は、例えば、これに限定されるものではないが、電子的ストレージ・デバイス、磁気ストレージ・デバイス、光学ストレージ・デバイス、電磁気ストレージ・デバイス、半導体ストレージ・デバイスまたは上記の任意の適切な組み合わせであってもよい。コンピュータ可読ストレージ媒体のより具体的な例示の非網羅的リストとしては、ポータブルなコンピュータ・ディスクレット、ハード・ディスク、ダイナミックまたはスタティック・ランダム・アクセス・メモリ（RAM）、リード・オンリー・メモリ（ROM）、消去可能プログラマブル・リード・オンリー・メモリ（EPROMまたはフラッシュ・メモリ）、磁気ストレージ・デバイス、ポータブルなコンパクト・ディスク・リード・オンリー・メモリ（CD-ROM）、デジタル・バーサタイル・ディスク（DVD）、メモリースティック、フロッピー（登録商標）ディスク、パンチカードまたは記録された命令を有する溝内の隆起構造のような機械的エンコードされたデバイス、および上記の任意の適切な組み合わせが含まれる。コンピュータ可読ストレージ媒体は、本明細書で使用されるように、電波、自由伝搬する電磁波、導波路または他の伝送媒体を伝搬する電磁波（たとえば、ファイバ光ケーブルを通過する光パルス）または、ワイヤを通して伝送される電気信号のような、それ自体が一時的な信号として解釈されるものではない。

30

40

【0064】

本明細書で説明されるコンピュータ可読プログラム命令は、コンピュータ可読ストレージ媒体からそれぞれのコンピュータ／処理デバイスに、または、外部コンピュータまたは外部ストレージ・デバイスに、例えばインターネット、ローカル・エリア・ネットワーク、ワイド・エリア・ネットワークもしくは無線ネットワークまたはその組み合わせといったネットワークを介してダウンロードすることができる。ネットワークは、銅伝送ケーブル、光伝送ファイバ、無線伝送、ルータ、ファイアウォール、スイッチ、ゲートウェイ・

50

コンピュータもしくはエッジサーバまたはその組み合わせを含んでもよい。各コンピュータ／処理デバイスにおけるネットワーク・アダプタ・カードまたはネットワーク・インタフェースは、ネットワークからコンピュータ可読プログラム命令を受信し、コンピュータ可読プログラム命令を、それぞれのコンピューティング／処理デバイス内のコンピュータ可読ストレージ媒体に格納するために転送する。

【0065】

本実施形態の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セットアーキテクチャ（ISA）命令、機械語命令、機械依存命令、マイクロコード、ファームウェア命令、状態設定データ、または、1以上のプログラミング言語の任意の組み合わせで書かれたソース・コードあるいはオブジェクト・コードであってよく、1以上のプログラミング言語は、Java（登録商標）、Smalltalk（登録商標）、C++またはこれらに類するものなどのオブジェクト指向言語、Cプログラミング言語または、Haskell, Lispなどの機能プログラミング言語、類似のプログラミング言語などの従来型の手続型言語を含む。コンピュータ可読プログラム命令は、スタンド・アローンのソフトウェア・パッケージとして、全体としてユーザのコンピュータ上で、部分的にユーザのコンピュータ上で、部分的にユーザのコンピュータ上かつ部分的に遠隔のコンピュータ上で、または、完全に遠隔のコンピュータまたはサーバ上、またはサーバのクラスタ上で実行されてもよい。後者のシナリオでは、遠隔のコンピュータは、ユーザのコンピュータに、ローカル・エリア・ネットワーク（LAN）またはワイド・エリア・ネットワーク（WAN）を含む任意のタイプのネットワークを通じて接続されてもよく、あるいは接続は、（例えば、インターネット・サービス・プロバイダを用いてインターネットを通じて）外部コンピュータになされてもよい。いくつかの実施形態においては、電氣的回路は、例えば、プログラマブル・ロジック回路、フィールド・プログラマブル・ゲート・アレイ（FPGA）、またはプログラマブル・ロジック・アレイ（PLA）を含むが、本実施形態の側面を実行するために、コンピュータ可読プログラム命令の状態情報を利用して、電氣的回路を個別化することによって、コンピュータ可読プログラム命令を実行してもよい。

【0066】

本実施形態の側面は、実施形態による方法、装置（システム）およびコンピュータ・プログラム製品のフローチャートもしくはブロック図またはその両方を参照しながら本明細書において説明される。フローチャート図もしくはブロック図またはその両方の各ブロック、および、フローチャート図もしくはブロック図またはその両方における複数のブロックの組み合わせは、コンピュータ可読プログラム命令によって実装されてもよいことが理解されよう。

【0067】

これらのコンピュータ可読プログラム命令は、汎用コンピュータ、特定用途コンピュータのプロセッサまたは他のプログラマブル・データ処理装置に提供され、コンピュータのプロセッサまたは他のプログラマブル・データ処理装置を介して実行される命令が、フローチャート図もしくはブロック図またはその両方のブロックまたは複数のブロックにおいて特定される機能／作用を実装するための手段を作成するように、マシンを生成する。これらのコンピュータ可読プログラム命令は、また、コンピュータ、プログラマブル・データ処理装置もしくは他のデバイスまたはその組み合わせに特定のやり方で機能するよう指示できるコンピュータ可読ストレージ媒体に格納され、それに格納された命令を有するコンピュータ可読ストレージ媒体に、フローチャート図もしくはブロック図またはその両方ブロックまたは複数のブロックで特定される機能／作用の側面を実装する命令を含む製品が含まれるようにする。

【0068】

コンピュータ可読プログラム命令は、また、コンピュータ、他のプログラマブル・データ処理装置、または他のデバイスにロードされ、コンピュータ、他のプログラマブル・データ処理装置または他のデバイス上で一連の動作ステップを実行させて、コンピュータ、

10

20

30

40

50

他のプログラマブル・データ処理装置または他のデバイス上で実行される命令が、フローチャート図もしくはブロック図またはその両方のブロックまたは複数のブロックで特定される機能／作用の側面を実装するように、コンピュータ実装処理を生成することもできる。

【0069】

図面におけるフローチャート図およびブロック図は、本実施形態の種々の実施形態に従ったシステム、方法およびコンピュータ・プログラム製品の可能な実装のアーキテクチャ、機能性および動作を示す。この点に関して、フローチャート図またはブロック図の各ブロックは、特定の論理機能を実装するための1以上の実行可能な命令を含む、モジュール、セグメントまたは命令の部分を表してもよい。いくつかの代替の実装では、ブロックにおいて言及された機能は、図面に示された順序から外れて生じる可能性がある。例えば、連続して示される2つのブロックは、実際には、同時に、実質的に同時に実行されてもよく、あるいは、複数のブロックは、関与する機能性に応じて逆の順序で実行されてもよい。ブロック図もしくはフローチャート図またはその両方の各ブロックおよびブロック図もしくはフローチャート図またはその両方の複数のブロックの組み合わせが、特定の機能または作用を実行し、または、特別な目的のハードウェアおよびコンピュータ命令の組み合わせを実施する、特定目的ハードウェアベースのシステムによって実装されてもよいことに留意されたい。

【0070】

本明細書において説明のために特定の実施形態が説明されたが、実施形態の精神および範囲から逸脱することなく種々の変形が可能であることが理解されるであろう。したがって、実施形態の保護の範囲は、以下の特許請求の範囲およびそれらの均等物によってのみ限定される。

10

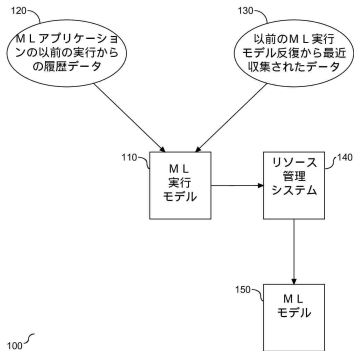
20

30

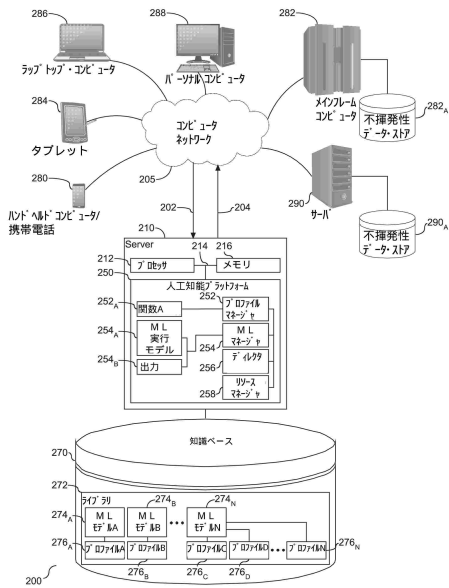
40

50

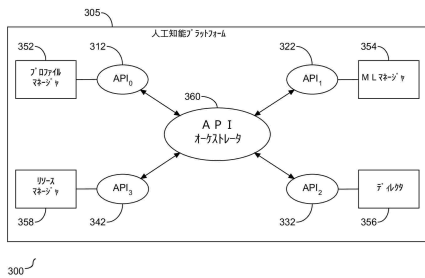
【図面】
【図 1】



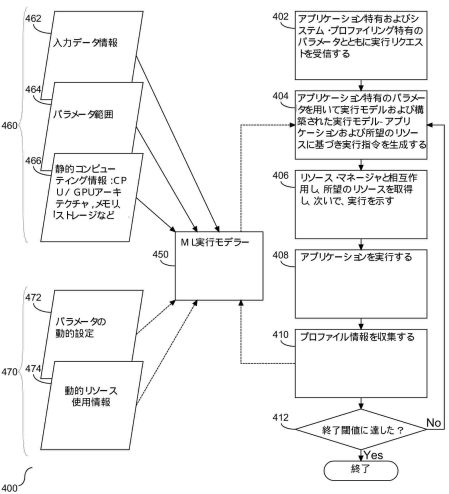
【図 2】



【図 3】



【図 4】



10

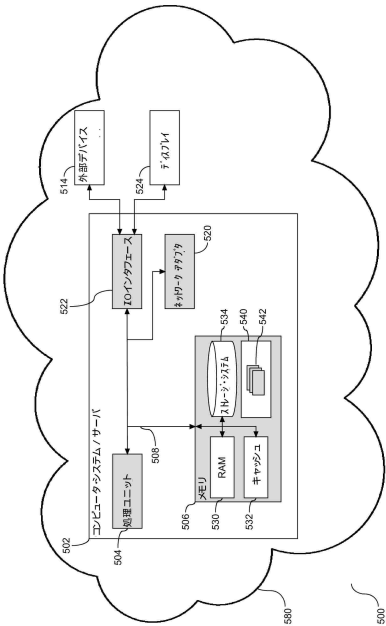
20

30

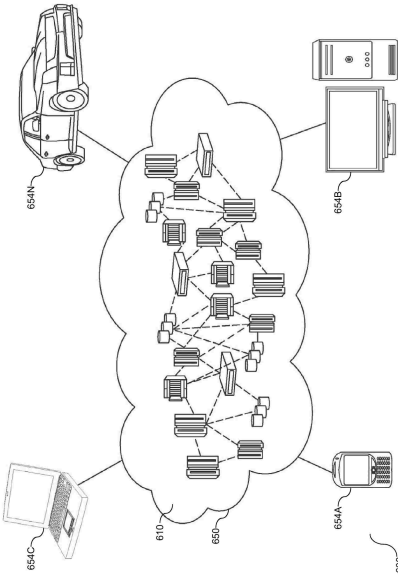
40

50

【図 5】



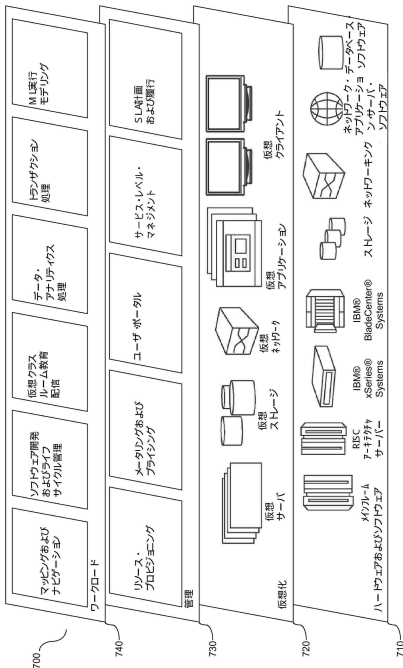
【図 6】



10

20

【図 7】



30

40

50

フロントページの続き

- (74)代理人 100120710
弁理士 片岡 忠彦
- (72)発明者 フォン, リアーナ
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- (72)発明者 シーラム, シーザラミ
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- (72)発明者 ヴェンカタラマン, ガネッシュ
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ ルート134 キ
ッチャワン・ロード1101
- (72)発明者 サハ, デバシッシュ
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- (72)発明者 ウム, ポウンレーク
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- (72)発明者 ヴェルマ, アーチット
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- (72)発明者 レディ, プラブハット, マディクンタ
アメリカ合衆国 ニューヨーク州 10598-0218 ヨークタウン・ハイツ キッチャワン・
ロード1101
- 審査官 三坂 敏夫
- (56)参考文献 米国特許出願公開第2019/0325307 (US, A1)
特開2019-192207 (JP, A)
特開2017-182114 (JP, A)
特表2010-533924 (JP, A)
特開2012-022558 (JP, A)
- (58)調査した分野 (Int.Cl., DB名)
G06F 9/455-9/54
G06N 20/00