

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7580569号
(P7580569)

(45)発行日 令和6年11月11日(2024.11.11)

(24)登録日 令和6年10月31日(2024.10.31)

(51)国際特許分類		F I	
G 0 6 N	20/00 (2019.01)	G 0 6 N	20/00
G 0 6 Q	10/10 (2023.01)	G 0 6 Q	10/10

請求項の数 30 (全26頁)

(21)出願番号	特願2023-503130(P2023-503130)	(73)特許権者	520262319 ユーアイパス、インコーポレイテッド UiPath, Inc. アメリカ合衆国、ニューヨーク 100 17、ニューヨーク、ヴァンダービルト アベニュー 1、シックスティース フロア 1 Vanderbilt Avenue , 60th Floor, New Yo rk, NY 10017, United States of America
(86)(22)出願日	令和2年9月24日(2020.9.24)	(74)代理人	110003487 弁理士法人東海特許事務所
(65)公表番号	特表2023-536063(P2023-536063 A)	(72)発明者	シング、ブラブディーブ アメリカ合衆国 10016 ニューヨー ク、ニューヨーク、パーク アベニュー 最終頁に続く
(43)公表日	令和5年8月23日(2023.8.23)		
(86)国際出願番号	PCT/US2020/052424		
(87)国際公開番号	WO2022/019934		
(87)国際公開日	令和4年1月27日(2022.1.27)		
審査請求日	令和5年9月14日(2023.9.14)		
(31)優先権主張番号	16/934,369		
(32)優先日	令和2年7月21日(2020.7.21)		
(33)優先権主張国・地域又は機関	米国(US)		

(54)【発明の名称】 ロボティックプロセスオートメーションのための人工知能/機械学習モデルドリフト検出および修正

(57)【特許請求の範囲】

【請求項1】

非一時的なコンピュータ読み取り可能な媒体上に格納されたロボティックプロセスオートメーション(RPA)のための人工知能(AI)/機械学習(ML)モデルドリフト検出および修正を実行するためのコンピュータプログラムであって、前記コンピュータプログラムは、少なくとも1つのプロセッサが、

AI/MLモデルのための入力データに係る情報を解析してデータドリフトが発生したか否かを判断し、前記AI/MLモデルの遂行からの結果に係る情報を解析してモデルドリフトが発生したか否かを判断し、またはその両方を行い、

前記情報の解析に基づき、変化条件が見出され、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、前記AI/MLモデルを再訓練するように構成され、

前記情報は、前記AI/MLモデルによって行われた予測の1または複数の統計分布を含み、前記コンピュータプログラムはさらに、前記少なくとも1つのプロセッサが、

前記1または複数の統計分布を、複数のRPAロボットによって取られたそれぞれのアクションにマッピングするように構成される、コンピュータプログラム。

【請求項2】

前記プログラムはさらに、前記少なくとも1つのプロセッサが、

前記変化条件、前記変化しきい値、またはその両方が、少なくとも所定の期間、満たされているまたはそれを越えているかを検証するように構成される、請求項1に記載のコン

10

20

コンピュータプログラム。

【請求項 3】

前記解析は、共変量ドリフトについて、元のデータセットと新しいデータセットの一次元分布間の 1 または複数の距離を測定することを含む、請求項 1 に記載のコンピュータプログラム。

【請求項 4】

前記解析は、統計モーメントが変化するか否かを判断することを含む、請求項 1 に記載のコンピュータプログラム。

【請求項 5】

前記統計モーメントは、平均、分散、歪度、尖度、共分散、またはそれらの組み合わせを含む、請求項 4 に記載のコンピュータプログラム。

10

【請求項 6】

前記情報は、前記 AI / ML モデルがどのような予測を行ったか、いくつの予測を前記 AI / ML モデルが行ったかおよび RPA ロボットによっていくつの予測が使用されたか、前記 AI / ML モデルによって行われた前記予測の 1 もしくは複数の統計分布、前記 AI / ML モデルに提供された入力データ、またはそれらの組み合わせを含む、請求項 1 に記載のコンピュータプログラム。

【請求項 7】

前記アクションは、それぞれの前記 RPA ロボットの RPA ワークフローのそれぞれのアクティビティで行われる、請求項 1 に記載のコンピュータプログラム。

20

【請求項 8】

前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、前記再訓練された AI / ML モデルが 1 または複数のパフォーマンスしきい値を満たしていることを検証し、

前記再訓練された AI / ML モデルが前記 1 または複数のパフォーマンスしきい値を満たした場合、前記再訓練された AI / ML モデルを展開するように構成される、請求項 1 に記載のコンピュータプログラム。

【請求項 9】

前記再訓練された AI / ML モデルは、以前のバージョンの前記 AI / ML モデルと共に展開され、前記プログラムはさらに、前記少なくとも 1 つのプロセッサが、

30

実データを使用して、前記再訓練された AI / ML モデルと以前のバージョンの前記 AI / ML モデルのモデルドリフトを測定し、

前記再訓練された AI / ML モデルが、測定された前記モデルドリフトに基づいて以前のバージョンの前記 AI / ML モデルよりも前記実データに対して優れたパフォーマンスを発揮する場合、以前のバージョンの前記 AI / ML モデルの代わりに前記再訓練された AI / ML モデルを使用するように構成される、請求項 8 に記載のコンピュータプログラム。

【請求項 10】

前記再訓練された AI / ML モデルが、測定された前記モデルドリフトに基づいて、以前のバージョンの前記 AI / ML モデルよりも前記実データに対して優れたパフォーマンスを発揮しない場合、前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、

40

前記再訓練された AI / ML モデルを破棄するか、またはさらに再訓練を行うように構成される、請求項 9 に記載のコンピュータプログラム。

【請求項 11】

前記再訓練された AI / ML モデルは、カナリアテスト技法またはチャンピオン / チャレンジャーテスト技法を用いて展開される、請求項 8 に記載のコンピュータプログラム。

【請求項 12】

前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、RPA ロボットから、前記 AI / ML モデルを遂行する要求と前記 AI / ML モデルに

50

関する入力情報とを受信し、

前記入力情報が前記 A I / M L モデルのためのフレームワークに適合していることを検証し、

前記入力情報が前記フレームワークに適合する場合、

前記入力情報を用いて前記 A I / M L モデルを遂行し、かつ

前記 A I / M L モデルの遂行の結果を前記 R P A ロボットに返すように構成される、請求項 1 に記載のコンピュータプログラム。

【請求項 1 3】

前記入力情報が前記フレームワークに適合しない場合、前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、

前記 R P A ロボットに、前記入力情報が前記フレームワークに適合していないことを示すエラーを提供するように構成される、請求項 1.2 に記載のコンピュータプログラム。

【請求項 1 4】

前記フレームワークは、複数の A I / M L モデルのために強制される、請求項 1.2 に記載のコンピュータプログラム。

【請求項 1 5】

前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、

前記情報に対して次元削減技術を実行して、前記情報の二次元表現を生成し、

前記情報の二次元表現を表示するように構成される、請求項 1 に記載のコンピュータプログラム。

【請求項 1 6】

前記情報の解析は、複数のそれぞれの R P A ロボットによって遂行される複数の R P A プロセスにわたって実行される、請求項 1 に記載のコンピュータプログラム。

【請求項 1 7】

前記コンピュータプログラムはさらに、前記少なくとも 1 つのプロセッサが、

前記入力データに基づいて、前記再訓練された A I / M L モデルまたは以前のバージョンの前記 A I / M L モデルを選択し、

選択されたバージョンの前記 A I / M L モデルを呼び出すように構成される、請求項 1 に記載のコンピュータプログラム。

【請求項 1 8】

前記選択は、階層的に行われる、請求項 1.7 に記載のコンピュータプログラム。

【請求項 1 9】

ロボティックプロセスオートメーション (R P A) のための人工知能 (A I) / 機械学習 (M L) モデルドリフト検出および修正を実行するためのコンピュータ実装方法であって、

A I / M L モデルのための入力データに係る情報を解析してデータドリフトが発生したか否かを判断し、前記 A I / M L モデルの遂行からの結果に係る情報を解析してモデルドリフトが発生したか否かを判断し、またはその両方を行い、

前記情報の解析に基づき、変化条件が見出され、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、

前記 A I / M L モデルを再訓練し、

前記再訓練された A I / M L モデルが 1 または複数のパフォーマンスしきい値を満たしていることを検証し、

前記再訓練された A I / M L モデルが前記 1 または複数のパフォーマンスしきい値を満たした場合、前記再訓練された A I / M L モデルを展開することを含み、

前記情報は、前記 A I / M L モデルによって行われた予測の 1 または複数の統計分布を含み、前記方法はさらに、

前記 1 または複数の統計分布を、複数の R P A ロボットによって取られたそれぞれのアクションにマッピングすることを含む、コンピュータ実装方法。

【請求項 2 0】

10

20

30

40

50

前記変化条件、前記変化しきい値、またはその両方が、少なくとも所定の期間、満たされているまたはそれを超えているかを検証することをさらに含む、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.1】

前記アクションは、それぞれの前記 RPA ロボットの RPA ワークフローのそれぞれのアクティビティで行われる、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.2】

前記再訓練された AI / ML モデルは、以前のバージョンの前記 AI / ML モデルと共に展開され、前記方法はさらに、

実データを使用して、前記再訓練された AI / ML モデルと以前のバージョンの前記 AI / ML モデルのモデルドリフトを測定し、

前記再訓練された AI / ML モデルが、測定された前記モデルドリフトに基づいて以前のバージョンの前記 AI / ML モデルよりも前記実データに対して優れたパフォーマンスを発揮する場合、以前のバージョンの前記 AI / ML モデルの代わりに前記再訓練された AI / ML モデルを使用することを含む、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.3】

RPA ロボットから、前記 AI / ML モデルを遂行する要求と前記 AI / ML モデルに関する入力情報とを受信し、

前記入力情報が前記 AI / ML モデルのためのフレームワークに適合していることを検証し、

前記入力情報が前記フレームワークに適合する場合、

前記入力情報を用いて前記 AI / ML モデルを遂行し、かつ

前記 AI / ML モデルの遂行の結果を前記 RPA ロボットに返すことをさらに含み、

前記フレームワークは、複数の AI / ML モデルのために強制される、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.4】

前記入力情報が前記フレームワークに適合していない場合、前記方法はさらに、

前記 RPA ロボットに、前記入力情報が前記フレームワークに適合していないことを示すエラーを提供することを含む、請求項 2.3 に記載のコンピュータ実装方法。

【請求項 2.5】

前記情報に対して次元削減技術を実行して、前記情報の二次元表現を生成し、

前記情報の二次元表現を表示することをさらに含む、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.6】

前記情報の解析は、複数のそれぞれの RPA ロボットによって遂行される複数の RPA プロセスにわたって実行される、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.7】

前記入力データに基づいて、前記再訓練された AI / ML モデルまたは以前のバージョンの前記 AI / ML モデルを選択し、

選択されたバージョンの前記 AI / ML モデルを呼び出すことをさらに含む、請求項 1.9 に記載のコンピュータ実装方法。

【請求項 2.8】

ロボティックプロセスオートメーション (RPA) のための人工知能 (AI) / 機械学習 (ML) モデルドリフト検出および修正を実行するためのコンピュータ実装方法であって、

前記 AI / ML モデルによって行われた予測の 1 もしくは複数の統計分布を含む情報であってデータドリフトが発生したかどうかを判断するための AI / ML モデルのための入力データに関連する情報、モデルドリフトが発生しているかどうかを判断するための前記 AI / ML モデルの遂行からの結果、またはその両方を解析し、

前記 1 または複数の統計分布を、複数の RPA ロボットによって取られたそれぞれのア

10

20

30

40

50

クシオンにマッピングし、

前記情報の解析に基づき、少なくとも所定の期間、変化条件が見出され、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、前記 A I / M L モデルを再訓練することを含み、

前記アクションは、それぞれの前記 R P A ロボットの R P A ワークフローのそれぞれのアクティビティで行われる、コンピュータ実装方法。

【請求項 29】

前記情報は、いくつかの予測を前記 A I / M L モデルが行ったかおよび前記複数の R P A ロボットによっていくつかの予測が使用されたかを含む、請求項 28 に記載のコンピュータ実装方法。

10

【請求項 30】

前記複数の R P A ロボットのうちの 1 つの R P A ロボットから、前記 A I / M L モデルを遂行する要求と前記 A I / M L モデルに関する入力情報とを受信し、

前記入力情報が前記 A I / M L モデルのためのフレームワークに適合していることを検証し、

前記入力情報が前記フレームワークに適合する場合、

前記入力情報を用いて前記 A I / M L モデルを遂行し、かつ

前記 A I / M L モデルの遂行の結果を前記 R P A ロボットに返すことをさらに含み、

前記フレームワークは、複数の A I / M L モデルのために強制される、請求項 28 に記載のコンピュータ実装方法。

20

【発明の詳細な説明】

【技術分野】

【0001】

(関連出願への相互参照)

本出願は、2020年7月21日に提出された米国非仮特許出願第16/934,369号の利益を主張する。先に提出された本出願の主題は、その全体を参照することにより、本明細書に組み込まれる。

【0002】

本発明は概して、ロボティックプロセスオートメーション(RPA)に関し、より具体的には、RPAのための人工知能(AI)/機械学習(ML)モデルドリフト検出および修正に関する。

30

【背景技術】

【0003】

訓練された A I / M L モデルが展開されて、画像認識および文書処理などのさまざまなタスクの確率的ソリューションを提供し得る。しかし、A I / M L モデルによる予測は、時間の経過とともに変化またはドリフトし得る。ソフトウェアロボットが、A I / M L モデルを利用してタスクを実行し得るコンテキストでは、ユーザーの知識の有無にかかわらず、またはユーザーがまったく関与していない可能性があるため、A I / M L モデルドリフトは特に問題になり得る。したがって、A I / M L モデルドリフトを検出および/または修正するための改善された技術は有益であり得る。

40

【発明の概要】

【0004】

本発明の特定の実施形態は、現在の R P A 技術によってまだ十分に特定されていない、評価されていない、または解決されていない本分野における問題およびニーズのソリューションを提供し得る。例えば、本発明のいくつかの実施形態は、R P A のための A I / M L モデルドリフトの検出および修正に関する。

【0005】

実施形態では、R P A のための A I / M L モデルドリフト検出および修正を実行するためのコンピュータプログラムは、非一時的なコンピュータ読み取り可能な媒体上に格納される。コンピュータプログラムは、少なくとも 1 つのプロセッサが、A I / M L モデルの

50

ための入力データに係る情報を解析してデータドリフトが発生したか否かを判断し、A I / M Lモデルの遂行からの結果に係る情報を解析してモデルドリフトが発生したか否かを判断し、またはその両方を行うように構成される。情報の解析に基づき、変化条件が見出され、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、コンピュータプログラムはさらに、少なくとも1つのプロセッサが、A I / M Lモデルを再訓練するように構成される。

【0006】

実施形態では、R P AのためのA I / M Lモデルドリフト検出および修正を行うためのコンピュータ実装方法は、A I / M Lモデルのための入力データに係る情報を解析してデータドリフトが発生したか否かを判断し、A I / M Lモデルの遂行からの結果に係る情報を解析してモデルドリフトが発生したか否かを判断し、またはその両方を行うことを含む。情報の解析に基づき、変化条件が見出され、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、方法は、A I / M Lモデルを再訓練することと、再訓練されたA I / M Lモデルが1または複数のパフォーマンスしきい値を満たすことを検証することと、再訓練されたA I / M Lモデルが1または複数のパフォーマンスしきい値を満たす場合、再訓練されたA I / M Lモデルを展開することを含む。

10

【0007】

さらに別の実施形態では、R P AのためのA I / M Lモデルドリフト検出および修正を行うためのコンピュータ実装方法は、A I / M Lモデルによって行われた予測の1または複数の統計分布を含む情報を解析することを含む。情報は、データドリフトが発生したかどうかを判断するためのA I / M Lモデルの入力データ、モデルドリフトが発生したかどうかを判断するためのA I / M Lモデルの遂行からの結果、またはその両方に関する。方法はまた、1または複数の統計分布を、複数のR P Aロボットによって取られたそれぞれのアクションにマッピングすることを含む。情報の解析に基づき、変化条件が見出され、少なくとも所定の期間、変化しきい値が満たされるもしくはそれを超えるか、またはその両方の場合、方法はさらに、A I / M Lモデルを再訓練することを含む。アクションは、それぞれの前記R P AロボットのR P Aワークフローのそれぞれのアクティビティで行われる。

20

【図面の簡単な説明】

【0008】

本発明の特定の実施形態の利点が容易に理解されるように、上記で簡単に説明した本発明のより特定の説明は、添付の図面に図示されている特定の実施形態を参照して描写される。これらの図面は、本発明の典型的な実施形態のみを描いており、したがって、その範囲を限定するものとは考えられないことが理解されるべきであるが、本発明は、以下の添付の図面を使用することにより、さらなる特定および詳細をもって描写され、説明されるであろう。

30

【0009】

【図1】本発明の実施形態による、ロボティックプロセスオートメーション(R P A)システムを示すアーキテクチャ図である。

【0010】

【図2】本発明の実施形態による、展開したR P Aシステムを示すアーキテクチャ図である。

40

【0011】

【図3】本発明の実施形態による、デザイナー、アクティビティ、およびドライバの間関係を示すアーキテクチャ図である。

【0012】

【図4】本発明の実施形態による、R P Aシステムを示すアーキテクチャ図である。

【0013】

【図5】本発明の実施形態による、R P AのためのA I / M Lモデルドリフト検出および修正を実行するように構成されたコンピューティングシステムを示すアーキテクチャ図で

50

ある。

【0014】

【図6】本発明の実施形態による、RPAのためのAI/MLモデルドリフト検出および修正を実行するように構成されたシステムを示すアーキテクチャ図である。

【0015】

【図7A】本発明の実施形態による、RPAのためのAI/MLモデルドリフト検出および修正を実行するための処理を示すフローチャートである。

【図7B】本発明の実施形態による、RPAのためのAI/MLモデルドリフト検出および修正を実行するための処理を示すフローチャートである。

【発明を実施するための形態】

10

【0016】

別段の記載がない限り、類似の参照文字は、添付の図面全体で一貫して対応する特徴を示す。

【0017】

(実施形態の詳細な説明)

いくつかの実施形態は、RPAのためのAI/MLモデルドリフト検出および修正に関するものである。いくつかの実施形態は、AI/MLモデルがいくつ予測を行ったかおよびいくつが実際に使用されたか、予測の1または複数の統計分布などの重要業績評価指標(KPI)を格納する。変化しきい値が満たされるまたはそれを超える場合、アラートまたは再訓練トリガが生成され得る。例えば、変化しきい値は、少なくとも一定量の、潜在的に少なくとも一定期間の、過去のパフォーマンスからの逸脱であり得る。AI/MLモデルは、その後、潜在的に1もしくは複数のユーザーおよび/またはロボットによって提供される新しい(より新しい)データを使用して、訓練され得、新しいバージョンのAI/MLモデルまたは全く新しいモデルが、より良い予測を提供し、ドリフトの問題を解決するように展開され得る。

20

【0018】

AI/MLモデルドリフトは、データドリフト、モデルドリフト、またはその両方が原因で発生し得る。データドリフトは、AI/MLモデルのための入力データの特性が、AI/MLモデルを学習するために使用されたデータから時間の経過とともに変化する場合である。これにより、AI/MLモデルによって行われた予測の精度が低下することがあり得る。モデルドリフトは、AI/MLモデルへの入力データとAI/MLモデルからの出力データとの間の関係における根本的な問題における経時的な変化である。ドリフトのこれらの形態の一方または両方が発生すると、AI/MLモデルのパフォーマンスが低下したり、AI/MLモデルが解決するために訓練された問題に対して不正確になったりする場合がある。本明細書では、データドリフトおよびモデルドリフトをまとめて「コンセプトドリフト」と呼ぶ。

30

【0019】

いくつかの実施形態では、統計分布は、RPAロボット(複数可)がそのワークフロー(複数可)で行っているアクションにマッピングされる。特定の実施形態では、統計分布は、ビジネス成果に一致させられ得る。例えば、特定のビジネス成果に、1~5のスケールが使用され、通常、結果が1~5スケールの1つの標準分布(例えば、2~4)に収まる場合を考えてみる。1および5がより一般的になり始めた場合(例えば、1標準偏差よりも頻繁に発生する場合)、例外が自動的にトリガされ得る。人間のレビューアは、AI/MLモデルのパフォーマンス特性を表示して、AI/MLモデルがどのようにドリフトしたかを判断し得る。モデルを改善し、そしてビジネスの成果を向上させるために、人による検証が実行され得る。ドリフトの根本的な原因は、データドリフトまたはモデルドリフトのいずれかであり得る。いくつかの実施形態では、各ビジネス成果についてこれらの統計分布をユーザーに表示するダッシュボードが提示され得る。

40

【0020】

RPAのコンテキストでは、ドリフトの検出が特に重要になる場合がある。例えば、R

50

P A ロボットが予測を行うために A I / M L モデルを呼び出している場合、現在のところ、これらのシナリオで人間の監督なしでモデルの品質をチェックする既存の方法はない。R P A ロボットは、A I / M L モデルの消費者であり、それらは、自律的に動作し得る。ユーザーは、これらの A I / M L モデルの予測に基づいてロボットが実行するアクションを制御できない場合があり、そのような R P A ロボットの自律性は、望ましくないまたは危険でさえあり得る結果につながる場合がある（例えば、ミッションクリティカルシステムの場合）。

【 0 0 2 1 】

いくつかの実施形態は、データドリフトおよびモデルドリフトを測定するために 1 または複数のアルゴリズムを使用する。例えば、データセットの次元を小さくする技法（例えば、主成分分析（P C A））、密度推定（すなわち、高次元空間における密度の連続性を評価する）、それらの組み合わせなどが用いられ得る。しかし、本発明の範囲から逸脱することなく、ドリフトを測定するために任意の適切な技法（複数可）が使用され得ることに留意されたい。

10

【 0 0 2 2 】

特定の実施形態では、A I / M L モデルドリフトを検出および扱う方法について、自動化プラットフォームによってフレームワークが提供および強制される。例えば、A I / M L モデルが本番環境に展開された場合、そのモデルは、自動化プラットフォームに展開された他の A I / M L モデルと共通の入力スキーマを持ち得る。スキーマ定義が強制される程度は、いくつかの実施形態では、A I / M L モデル自体および A I / M L モデルパイプラインに依存し得る。入力スキーマは、A I / M L モデルが入力として受信するものを記述する。いくつかの実施形態では、これは、事前定義された列および特定のデータタイプを有するカンマ区切り値（C S V）ファイルであり得る。これにより、特定のフィールド要件および/または値が強制され得る。

20

【 0 0 2 3 】

例えば、電子メール本文がシリアル化された J a v a S c r i p t O b j e c t N o t a t i o n（J S O N）で A I / M L モデルへの入力として渡される場合、電子メール本文は入力として扱われ得る。しかし、数値データを含む C S V ファイルは、A I / M L モデルがデータの解釈方法を認識できない場合があるため、受信されない場合がある。したがって、スキーマは、望ましくないまたは許可されていないデータ形式が A I / M L モデルに供給されるのを防ぐのに役立ち得る。

30

【 0 0 2 4 】

展開された A I / M L モデルが実行され、解析が実行されると、入力データの表現および A I / M L モデル予測が徐々に構築され得る。例えば、元のデータセットと新しいデータセットの次元分布間の距離は、共変量ドリフトについて測定され得る。いくつかの実施形態では、解析は、統計モーメントが変化するかどうかを判断することを含む。特定の実施形態では、統計モーメントは、平均、分散、歪度、尖度、共分散、またはそれらの組み合わせを含む。また、残留ドリフト間の距離が 2 つのモデルのために確認され得る。例えば、A I / M L モデルの 2 つのバリエーション（例えば、新しいバージョンのモデルと以前のバージョンのモデル）が、異なるデータセットまたはデータセットの異なる部分に対して訓練される場合、モデルの分布間の距離を使用して、なぜ一方のモデルでドリフトが起こり、他方では起こらない（または少なくとも少ない）可能性があるかを判断しようとすることができる。

40

【 0 0 2 5 】

入力データが 1 ~ 5 の範囲で該当する 5 つのクラスを A I / M L モデルが予測する、上記の例をもう一度考えてみる。A I / M L モデルは 6 ヶ月間展開され、予測の 9 5 % がクラス 2、3、または 4 であった。しかし、A I / M L モデルはその後、クラス 1 およびクラス 5 の予測をより多く行うようになる。そして、入力データの特徴（例えば、分布）が変化し、それが予測の変化を引き起こしたのか、それとも、入力データに大きな変化がなくても A I / M L モデルの予測の変化が生じたのかが判断され得る。

50

【 0 0 2 6 】

いくつかの実施形態では、RPAワークフロー内のアクティビティは、ドリフトが発生しているかどうか（および場合によっては、どのようなタイプか）を検出し得る。例えば、開発者がデータまたはモデルドリフトを考慮することを可能にするアクティビティの収集が、RPAデザイナーアプリケーションにおいて利用可能であってもよい。これらのアクティビティは、ドリフトが発生する特定の割合、標準偏差を超える、一般的でない予測が著しく一般的になる、などに基づいて構成され得る。

【 0 0 2 7 】

ドリフトが発生していると判断された場合、トリガが生成されてもよい。例えば、トリガは、AI/MLモデルおよび/または入力データによってなされた予測を、人間のレビューアが分析するために送信および格納することを引き起こしてもよい。アクションセンターキューは、人間がモデルの新バージョンを検証し、自動的に再訓練することを可能にし得る。人間のレビューアは、次に、AI/MLモデルが入力データに基づいて予測したものをレビューし、モデルの精度に問題があるかどうかを判断し得る。再訓練されたAI/MLモデルが特定のしきい値（例えば、信頼しきい値、ある種のランダムサンプリング、他のヒューリスティックなど）を満たす場合、それは既存のAI/MLモデルと一緒に自動的に展開され、モデルのドリフトは、新しいバージョンのAI/MLモデルが実データ上でより良く機能しているかどうかを見るために共存して（side-by-side）測定され得る。モデルがそのしきい値を下回っている場合、しきい値が満たされていない理由を判断するために人間のレビューが実行され得、そして、それがしきい値を満たすまたはそれを超えるようにAI/MLモデルを訓練することが試行され得る。

【 0 0 2 8 】

いくつかの実施形態では、AI/MLモデルが訓練される場合、訓練データの一部は、後でテストに使用するために分割される。モデルを評価するための1つの高レベルのメトリック（metric）を作成するために、メトリックの収集が使用され得る。一例は、文書理解モデルのF1スコアである。これは精度および呼び戻し（recall）の派生物であり、両者の間にはトレードオフが存在する。交差検証（cross-validation）は、データの別の部分を分割してそれを混ぜ合わせ、混ぜ合わされたデータを使ってモデルをテストするために実行される。しかし、分割されたデータの一部が実世界データを代表していない場合、モデルが訓練データに過剰に適合してしまい、実世界アプリケーションでうまく機能しない場合がある。

【 0 0 2 9 】

いくつかの実施形態では、このプロセスは、自動化され得る。F1スコアが例えば90%超である場合、新しいバージョンのAI/MLモデルが展開され得る。この新しいバージョンは、その後、人間によってドリフトについてチェックされおよび/または検証され得る。

【 0 0 3 0 】

特定の実施形態は、ロボティックプロセスオートメーション（RPA）に採用されてもよい。図1は、本発明の実施形態による、RPAシステム100を示すアーキテクチャ図である。RPAシステム100は、開発者がワークフローを設計して実装することを可能にするデザイナー110を含む。デザイナー110は、アプリケーション統合のためのソリューションを提供するとともに、サードパーティアプリケーション、管理情報技術（IT）タスク、およびビジネスITプロセスを自動化する。デザイナー110は、ビジネスプロセスのグラフィック表現である自動化プロジェクトの開発を容易にし得る。簡単に言えば、デザイナー110は、ワークフローおよびロボットの開発および展開を容易にする。

【 0 0 3 1 】

自動化プロジェクトは、本明細書で「アクティビティ」と定義されるワークフローで開発されたステップのカスタムセット間の遂行順序および関係の制御を開発者に与えることにより、ルールベースのプロセスの自動化を可能にする。デザイナー110の実施形態の商業的な一例は、UiPath Studio（商標）である。各アクティビティは、ボタ

10

20

30

40

50

ンをクリックする、ファイルを読む、ログパネルに書き込むなどのアクションを含み得る。いくつかの実施形態では、ワークフローは入れ子になっているか、または埋め込まれ得る。

【0032】

ワークフローのタイプには、シーケンス、フローチャート、FSM、および/またはグローバル例外ハンドラなどを含み得るが、これらに限定されない。シーケンスは、ワークフローを乱雑にすることなく、あるアクティビティから別のアクティビティへのフローを可能にする、線形プロセスに特に適し得る。フローチャートは、特により複雑なビジネスロジックに適し得、複数の分岐ロジックオペレータを介して、より多様な方法で意思決定の統合およびアクティビティの接続を可能にする。FSMは、大規模なワークフローに特に適し得る。FSMは、条件（すなわち、遷移）またはアクティビティによりトリガされる有限の数の状態をそれらの遂行中に使用し得る。グローバル例外ハンドラは、遂行エラーに遭遇したときのワークフローの挙動を決定したり、プロセスをデバッグしたりするのに特に適し得る。

10

【0033】

ワークフローがデザイナー110内で開発されると、ビジネスプロセスの遂行は、コンダクタ120によって調整され、デザイナー110内で開発されたワークフローを遂行する1または複数のロボット130を調整する。コンダクタ120の実施形態の商業的な一例は、UiPath Orchestrator（商標）である。コンダクタ120は、環境におけるリソースの生成、監視、および展開の管理を容易にする。コンダクタ120は、

20

【0034】

コンダクタ120は、全てのロボット130を管理し得、ロボット130を集中ポイントから接続して遂行する。管理され得るロボット130のタイプには、アテンディッドロボット132、アンアテンディッドロボット134、開発ロボット（アンアテンディッドロボット134と同様であるが、開発およびテストの目的で使用される）、および非生産ロボット（アテンディッドロボット132と同様であるが、開発およびテストの目的で使用される）が含まれるが、これらに限定されない。アテンディッドロボット132は、ユーザーイベントによってトリガされ、同じコンピューティングシステム上で人と並んで動作する。アテンディッドロボット132は、集中プロセスデプロイメントおよびロギング媒体のためのコンダクタ120とともに使用され得る。アテンディッドロボット132は、人のユーザーが様々なタスクを達成するのを支援してもよく、ユーザーイベントによってトリガされてもよい。いくつかの実施形態では、プロセスは、このタイプのロボット上でコンダクタ120から開始することができず、および/またはそれらはロックされた画面の下で実行することができない。特定の実施形態では、アテンディッドロボット132は、ロボットトレイからまたはコマンドプロンプトからのみ起動され得る。アテンディッドロボット132は、いくつかの実施形態では、人の監督下で動作することが好ましい。

30

【0035】

アンアテンディッドロボット134は、仮想環境で無人で動作し、多くのプロセスを自動化し得る。アンアテンディッドロボット134は、リモート遂行、監視、スケジューリング、および作業キューのサポートの提供を担当し得る。全てのロボットタイプのためのデバッグは、いくつかの実施形態では、デザイナー110で実行され得る。アテンディッドロボットおよびアンアテンディッドロボットの両方は、メインフレーム、ウェブアプリケーション、VM、エンタープライズアプリケーション（例えば、SAP（登録商標）、Sales Force（登録商標）、Oracle（登録商標）などにより生成されたもの）、およびコンピューティングシステムアプリケーション（例えば、デスクトップおよびラップトップアプリケーション、モバイルデバイスアプリケーション、ウェアラブルコンピュータアプリケーションなど）を含むが、これらに限定されない様々なシステムおよびアプリケーションを自動化し得る。

40

50

【0036】

コンダクタ120は、プロビジョニング、展開、バージョンング、構成、キューイング、監視、ロギング、および/または相互接続性の提供を含むがこれらに限定されない様々な能力を有し得る。プロビジョニングは、ロボット130とコンダクタ120（例えば、ウェブアプリケーション）との間の接続を作成し、維持することを含み得る。展開は、遂行のために割り当てられたロボット130へのパッケージバージョンの正しい配信を保証することを含み得る。バージョンングは、いくつかの実施形態では、いくつかのプロセスまたは構成の固有のインスタンスの管理を含んでもよい。構成は、ロボット環境およびプロセス構成の維持および配信を含み得る。キューイングは、キューおよびキュー項目の管理を提供することを含み得る。監視は、ロボットの特定データを追跡し、ユーザーの権限を維持することを含み得る。ロギングは、データベース（例えば、SQLデータベース）および/または別のストレージメカニズム（例えば、大規模なデータセットを格納し、迅速にクエリを実行する能力を提供するElastic Search（登録商標））へのログの保存およびインデックス作成を含み得る。コンダクタ120は、サードパーティのソリューションおよび/またはアプリケーションのための通信の集中点として操作することにより、相互接続性を提供し得る。

10

【0037】

ロボット130は、デザイナ110で構築されたワークフローを実行する遂行エージェントである。ロボット（複数可）130のいくつかの実施形態の1つの商業的な例は、UiPath Robots（商標）である。いくつかの実施形態では、ロボット130は、デフォルトで、Microsoft Windows（登録商標）Service Control Manager（SCM）管理サービスをインストールする。その結果、このようなロボット130は、ローカルシステムアカウントの下でインタラクティブなWindows（登録商標）セッションを開くことができ、Windows（登録商標）サービスの権利を有し得る。

20

【0038】

いくつかの実施形態では、ロボット130は、ユーザーモードで設置され得る。このようなロボット130については、所定のロボット130が設置されているユーザーと同じ権利を有することを意味する。この特色はまた、各マシンを最大限に全活用することを保証する高密度（HD）ロボットにも利用可能であり得る。いくつかの実施形態では、いずれかのタイプのロボット130は、HD環境で構成され得る。

30

【0039】

いくつかの実施形態におけるロボット130は、複数のコンポーネントに分割され、それぞれが特定の自動化タスクに特化されている。いくつかの実施形態におけるロボットコンポーネントは、SCM管理ロボットサービス、ユーザーモードロボットサービス、エグゼキュータ、エージェント、およびコマンドラインを含むが、これらに限定されない。SCM管理ロボットサービスは、Windows（登録商標）セッションを管理および監視し、コンダクタ120と遂行ホスト（すなわち、ロボット130が遂行されるコンピューティングシステム）との間のプロキシとして動作する。これらのサービスは、ロボット130の資格情報を任されて管理する。コンソールアプリケーションは、ローカルシステム下のSCMにより起動される。

40

【0040】

いくつかの実施形態におけるユーザーモードロボットサービスは、Windows（登録商標）セッションを管理および監視し、コンダクタ120と遂行ホストとの間のプロキシとして動作する。ユーザーモードロボットサービスは、ロボット130の資格情報を任されて管理し得る。SCM管理ロボットサービスがインストールされていない場合、Windows（登録商標）アプリケーションが自動的に起動され得る。

【0041】

エグゼキュータは、Windows（登録商標）セッションの下で与えられたジョブを遂行し得る（つまり、ワークフローを遂行し得る。エグゼキュータは、モニタ毎のドット

50

パーインチ (D P I) 設定を意識し得る。エージェントは、システムトレイウィンドウに利用可能なジョブを表示する Windows (登録商標) P r e s e n t a t i o n F o u n d a t i o n (W P F) アプリケーションであり得る。エージェントはサービスのクライアントであり得る。エージェントは、ジョブの開始または停止、設定の変更を依頼し得る。コマンドラインはサービスのクライアントである。コマンドラインは、ジョブの開始を要求し、その出力を待つことができるコンソールアプリケーションである。

【 0 0 4 2 】

上で説明したように、ロボット 1 3 0 のコンポーネントが分割されていることは、開発者、サポートユーザー、およびコンピューティングシステムが、各コンポーネントが何を遂行しているかをより容易に実行し、特定し、および追跡するのに役立つ。この方法では、エグゼキュータとサービスに異なるファイアウォールルールを設定するなど、コンポーネントごとに特別な挙動を構成し得る。エグゼキュータは常に、いくつかの実施形態では、モニタごとの D P I 設定を認識し得る。その結果、ワークフローは、ワークフローが作成されたコンピューティングシステムの構成に関係なく、いずれかの D P I で遂行し得る。また、いくつかの実施形態では、デザイナー 1 1 0 からのプロジェクトは、ブラウザのズームレベルに依存しないようにし得る。D P I を認識していないまたは意図的に認識していないとマークされているアプリケーションの場合、いくつかの実施形態では D P I を無効にし得る。

【 0 0 4 3 】

図 2 は、本発明の実施形態による、展開した R P A システム 2 0 0 を示すアーキテクチャ図である。いくつかの実施形態では、R P A システム 2 0 0 は、図 1 の R P A システム 1 0 0 であってもよく、またはその一部であってもよい。クライアント側、サーバー側、またはその両方が、本発明の範囲から逸脱することなく、いずれかの所望の数のコンピューティングシステムを含むことができることに留意すべきである。クライアント側では、ロボットアプリケーション 2 1 0 は、エグゼキュータ 2 1 2、エージェント 2 1 4、およびデザイナー 2 1 6 を含む。しかし、いくつかの実施形態では、デザイナー 2 1 6 は、コンピューティングシステム 2 1 0 上で実行されていなくてもよい。エグゼキュータ 2 1 2 はプロセスを実行している。図 2 に示すように、複数のビジネスプロジェクトが同時に実行され得る。エージェント 2 1 4 (例えば、Windows (登録商標) サービス) は、本実施形態では、全てのエグゼキュータ 2 1 2 のための単一の接続ポイントである。この実施形態における全てのメッセージは、コンダクタ 2 3 0 に記録され、それは、データベースサーバー 2 4 0、インデクササーバー 2 5 0、またはその両方を介して、それらをさらに処理する。図 1 に関して上述したように、エグゼキュータ 2 1 2 は、ロボットコンポーネントであり得る。

【 0 0 4 4 】

いくつかの実施形態では、ロボットは、マシン名とユーザー名との間の関連付けを表す。ロボットは、複数のエグゼキュータを同時に管理し得る。同時に実行される複数のインタラクティブセッションをサポートするコンピューティングシステム (Windows (登録商標) S e r v e r 2 0 1 2 など) では、複数のロボットが同時に実行され得、それぞれが一意のユーザー名を使用して別々の Windows (登録商標) セッションで実行され得る。これは、上記の H D ロボットと呼ばれる。

【 0 0 4 5 】

エージェント 2 1 4 はまた、ロボットの状態を送信し (例えば、ロボットがまだ機能していることを示す「ハートビート」メッセージを定期的送信する)、遂行されるパッケージの要求されるバージョンをダウンロードすることにも責任を負う。エージェント 2 1 4 とコンダクタ 2 3 0 との間の通信は、いくつかの実施形態では、常にエージェント 2 1 4 によって開始される。通知シナリオでは、エージェント 2 1 4 は、後にコンダクタ 2 3 0 によってロボットにコマンド (例えば、開始、停止など) を送信するために使用される Web S o c k e t チャンネルを開いてもよい。

【 0 0 4 6 】

10

20

30

40

50

サーバー側には、プレゼンテーション層（ウェブアプリケーション 232、オープンデータプロトコル（ODa t a）代表状態転送（R E S T）アプリケーションプログラミングインターフェース（A P I）エンドポイント 234、通知および監視 236）、サービス層（A P I 実装 / ビジネスロジック 238）、永続層（データベースサーバー 240、インデクササーバー 250）が含まれる。コンダクタ 230 は、ウェブアプリケーション 232、O D a t a R E S T A P I エンドポイント 234、通知および監視 236、ならびに A P I 実装 / ビジネスロジック 238 を含む。いくつかの実施形態では、ユーザーがコンダクタ 230 のインターフェース（例えば、ブラウザ 220 を介して）で実行するほとんどのアクションは、様々な A P I を呼び出すことによって実行される。このような動作は、本発明の範囲を逸脱することなく、ロボット上でのジョブの起動、キュー内のデータの追加 / 削除、無人で実行するジョブのスケジューリングなどを含み得るが、これらに限定されない。ウェブアプリケーション 232 は、サーバープラットフォームのビジュアル層である。この実施形態では、ウェブアプリケーション 232 は、ハイパーテキストマークアップ言語（H T M L）およびジャバスクリプト（J S）を使用する。しかし、本発明の範囲を逸脱することなく、いずれかの所望のマークアップ言語、スクリプト言語、または他のいずれかのフォーマットを使用し得る。ユーザーは、コンダクタ 230 を制御するための様々な動作を実行するために、本実施形態ではブラウザ 220 を介してウェブアプリケーション 232 からのウェブページとインタラクションする。例えば、ユーザーは、ロボットグループを作成し、ロボットへのパッケージの割り当てを行い、ロボット毎および / またはプロセス毎のログを解析し、ロボットを起動しおよび停止などし得る。

10

20

【0047】

ウェブアプリケーション 232 に加えて、コンダクタ 230 は、O D a t a R E S T A P I エンドポイント 234 を公開するサービス層も含む。しかし、本発明の範囲を逸脱することなく、他のエンドポイントが含まれていてもよい。R E S T A P I は、ウェブアプリケーション 232 とエージェント 214 の両方により消費される。エージェント 214 は、本実施形態では、クライアントコンピュータ上の 1 または複数のロボットのスーパーバイザである。

【0048】

本実施形態の R E S T A P I は、構成、ロギング、監視、およびキューイング機能をカバーする。構成エンドポイントは、いくつかの実施形態では、アプリケーションのユーザー、権限、ロボット、アセット、リリース、および環境を定義し、構成するために使用されてもよい。ロギング R E S T エンドポイントは、例えば、エラー、ロボットによって送信された明示的なメッセージ、およびその他の環境固有の情報など、様々な情報をログに記録するために使用され得る。デブロイメント R E S T エンドポイントは、コンダクタ 230 においてジョブ開始コマンドが使用された場合に遂行されるべきパッケージのバージョンを問い合わせるためにロボットにより使用されてもよい。キューイング R E S T エンドポイントは、キューへのデータの追加、キューからのトランザクションの取得、トランザクションのステータスの設定など、キューおよびキューアイテムの管理を担ってもよい。

30

【0049】

R E S T エンドポイントの監視は、ウェブアプリケーション 232 およびエージェント 214 を監視してもよい。通知および監視 A P I 236 は、エージェント 214 の登録、エージェント 214 への構成設定の配信、ならびにサーバーおよびエージェント 214 からの通知の送受信に使用される R E S T エンドポイントであってもよい。通知および監視 A P I 236 は、いくつかの実施形態では、W e b S o c k e t 通信を使用してもよい。

40

【0050】

永続層は、本実施形態では、一对のサーバー - データベースサーバー 240（例えば、S Q L サーバー）およびインデクササーバー 250 を含む。本実施形態におけるデータベースサーバー 240 は、ロボット、ロボットグループ、関連するプロセス、ユーザー、役割、スケジュールなどの構成を格納する。この情報は、いくつかの実施形態では、ウェブ

50

アプリケーション 232 を介して管理される。データベースサーバー 240 は、キューおよびキューアイテムを管理してもよい。いくつかの実施形態では、データベースサーバー 240 は、ロボットにより記録されたメッセージを（インデクササーバー 250 に加えて、またはそれに代えて）格納してもよい。

【0051】

いくつかの実施形態では任意であるが、インデクササーバー 250 は、ロボットにより記録された情報を保存し、インデックスを作成する。特定の実施形態では、インデクササーバー 250 は、構成設定を介して無効化されてもよい。いくつかの実施形態では、インデクササーバー 250 は、オープンソースプロジェクトの全文検索エンジンである *Elastic Search*（登録商標）を使用する。ロボットにより記録されたメッセージ（例えば、ログメッセージまたはライン書き込みのようなアクティビティを使用して）は、ロギング REST エンドポイント（複数可）を介してインデクササーバー 250 に送信されてもよく、そこで将来の利用のためにそれらはインデックス化される。

10

【0052】

図 3 は、本発明の実施形態による、デザイナー 310、アクティビティ 320、330、およびドライバ 340 の間の関係 300 を示すアーキテクチャ図である。以上のように、開発者は、デザイナー 310 を用いて、ロボットによって遂行されるワークフローを開発する。ワークフローは、ユーザー定義のアクティビティ 320 および UI オートメーションアクティビティ 330 を含んでもよい。いくつかの実施形態では、画像中の非テキストの視覚的コンポーネントを特定することができ、これは、本明細書ではコンピュータビジョン（CV）と呼ばれる。このようなコンポーネントに関連するいくつかの CV アクティビティは、クリック、タイプ、テキストを取得、ホバー、要素の有無を検出、スコープの更新、ハイライトなどを含み得るが、これらに限定されない。いくつかの実施形態では、クリックは、例えば、CV、光学的文字認識（OCR）、ファジーテキストマッチング、およびマルチアンカーを使用して要素を特定し、それをクリックする。タイプは、上記および要素内のタイプを用いて要素を特定してもよい。テキストの取得は、特定のテキストの場所を特定し、OCR を使用してそれをスキャンし得る。ホバーは、要素を特定し、その上にホバーし得る。要素の有無の検出は、上述した技法を用いて、画面上に要素の有無を検出するかどうかを確認し得る。いくつかの実施形態では、デザイナー 310 に実装され得る数百または数千でさえもアクティビティが存在してもよい。しかし、本発明の範囲を逸脱することなく、いずれかの数および/またはアクティビティのタイプを利用することができる。

20

30

【0053】

UI オートメーションアクティビティ 330 は、低レベルのコード（例えば、CV アクティビティ）で記述され、画面とのインタラクションを促進する特別な低レベルのアクティビティのサブセットである。UI オートメーションアクティビティ 330 は、ロボットが所望のソフトウェアとインタラクションすることを可能にするドライバ 340 を介して、これらのインタラクションを促進する。例えば、ドライバ 340 は、OS ドライバ 342、ブラウザドライバ 344、VM ドライバ 346、エンタープライズアプリケーションドライバ 348 などを含んでもよい。

40

【0054】

ドライバ 340 は、フックを探したり、キーを監視したりするなど、低レベルで OS とインタラクションしてもよい。それらは、*Chrome*（登録商標）、*IE*（登録商標）、*Citrix*（登録商標）、*SAP*（登録商標）などとの統合を促進してもよい。例えば、「クリック」アクティビティは、ドライバ 340 を介して、これらの異なるアプリケーションにおいて同じ役割を果たす。

【0055】

図 4 は、本発明の実施形態による、RPA システム 400 を示すアーキテクチャ図である。いくつかの実施形態では、RPA システム 400 は、図 1 および/または図 2 の RPA システム 100 および/または 200 であってもよいし、それを含んでもよい。RPA

50

システム400は、ロボットを実行する複数のクライアントコンピューティングシステム410を含む。コンピューティングシステム410は、その上で実行されるウェブアプリケーションを介してコンダクタコンピューティングシステム420と通信することができる。コンダクタコンピューティングシステム420は、順番に、データベースサーバー430および任意のインデクササーバー440と通信することができる。

【0056】

図1および図3に関して、これらの実施形態ではウェブアプリケーションが使用されているが、本発明の範囲から逸脱することなく、任意の適切なクライアント/サーバーソフトウェアを使用することができることに留意すべきである。例えば、コンダクタは、クライアントコンピューティングシステム上で、非ウェブベースのクライアントソフトウェアアプリケーションと通信するサーバーサイドアプリケーションを実行してもよい。

10

【0057】

図5は、本発明の実施形態による、RPAのためのAI/MLモデルドリフト検出および修正を実行するように構成されたコンピューティングシステム500を示すアーキテクチャ図である。いくつかの実施形態では、コンピューティングシステム500は、本明細書に描かれたおよび/または記載された1または複数のコンピューティングシステムであってもよい。コンピューティングシステム500は、情報を通信するためのバス505または他の通信機構と、情報を処理するためのバス505に結合されたプロセッサ(複数可)510とを含む。プロセッサ(複数可)510は、中央処理ユニット(CPU)、特定用途集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、グラフィックスプロセッシングユニット(GPU)、それらの複数のインスタンス、および/またはそれらのいずれかの組み合わせを含む、いずれかのタイプの一般的または特定用途向けプロセッサであり得る。プロセッサ(複数可)510はまた、複数の処理コアを有してもよく、コアの少なくとも一部は、特定の機能を実行するように構成され得る。いくつかの実施形態では、複数並列処理が使用されてもよい。特定の実施形態では、少なくとも1つのプロセッサ(複数可)510は、生物学的ニューロンを模倣する処理要素を含むニューロモーフィック回路であり得る。いくつかの実施形態では、ニューロモーフィック回路は、フォンノイマンコンピューティングアーキテクチャの典型的なコンポーネントを必要としない場合がある。

20

【0058】

コンピューティングシステム500は、プロセッサ(複数可)510によって遂行される情報および命令を格納するためのメモリ515をさらに含む。メモリ515は、ランダムアクセスメモリ(RAM)、読み出し専用メモリ(ROM)、フラッシュメモリ、キャッシュ、磁気ディスクもしくは光ディスクなどの静的記憶装置、または他のタイプの非一時的なコンピュータ読み取り可能な媒体、あるいはそれらのいずれかの組み合わせで構成され得る。非一時的なコンピュータ読み取り可能な媒体は、プロセッサ(複数可)510によりアクセス可能ないずれかの利用可能な媒体であってもよく、揮発性媒体、不揮発性媒体または両方などを含み得る。また、媒体は、取り外し可能なもの、取り外し不可能なもの、または両方であってもよい。

30

【0059】

さらに、コンピューティングシステム500は、無線および/または有線接続を介して通信ネットワークへのアクセスを提供するために、トランシーバなどの通信デバイス520を含む。いくつかの実施形態では、通信デバイス520は、周波数分割多元接続(FDMA)、シングルキャリアFDMA(SC-FDMA)、時分割多元接続(TDMA)、符号分割多元接続(CDMA)、直交周波数分割多重方式(OFDM)、直交周波数分割多元接続(OFDMA)、移動体用グローバルシステム(GSM: Global System for Mobile)通信、汎用パケット無線サービス(GPRS: General Packet Radio Service)、ユニバーサル移動体通信システム(UMTS: Universal Mobile Telecommunications System)、cdma2000、広帯域CDMA(W-CDMA: Wideband

40

50

CDMA)、高速ダウンリンクパケットアクセス(HSDPA:High-Speed Downlink Packet Access)、高速アップリンクパケットアクセス(HSUPA:High-Speed Uplink Packet Access)、高速パケットアクセス(HSPA:High-Speed Packet Access)、ロングタームエボリューション(LTE:Long Term Evolution)、LTEアドバンス(LTE-A:LTE Advanced)、802.11x、Wi-Fi、Zigbee、超広帯域無線(UWB:Ultra-WideBand)、802.16x、802.15、Home Node-B(HnB)、Bluetooth、無線IDタグ(RFID:Radio Frequency Identification)、IrDA(Infrared Data Association)、近距離無線通信(NFC:Near-Field Communications)、第5世代(5G)、New Radio(NR)、それらのいずれかの組み合わせ、および/または本発明の範囲から逸脱することなく、いずれかの他の現在存在するまたは将来実装される通信標準および/またはプロトコルを使用するように構成され得る。いくつかの実施形態では、通信デバイス520は、本発明の範囲から逸脱することなく、単数のアンテナ、アレイ状のアンテナ、フェーズドアンテナ、スイッチドアンテナ、ビームフォーミングアンテナ、ビームステアリングアンテナ、それらの組み合わせ、および/またはいずれかの他のアンテナ構成である1または複数のアンテナを含み得る。

【0060】

プロセッサ(複数可)510は、バス505を介して、プラズマディスプレイ、液晶ディスプレイ(LCD)、発光ダイオード(LED)ディスプレイ、電界放出ディスプレイ(FED)、有機発光ダイオード(OLED)ディスプレイ、フレキシブルOLEDディスプレイ、フレキシブル基板ディスプレイ、プロジェクションディスプレイ、4Kディスプレイ、高精細ディスプレイ、Retina(登録商標)ディスプレイ、IPS(In-Plane Switching)ディスプレイ、またはユーザーに情報を表示するためのいずれかの他の適切なディスプレイなどのディスプレイ525にさらに結合されている。ディスプレイ525は、抵抗方式、静電容量方式、表面弾性波(SAW)静電容量方式、赤外線方式、光学イメージング方式、分散信号方式、音響パルス認識方式、フラストレート全内部反射方式などを用いて、タッチ(ハプティック)ディスプレイ、3次元(3D)タッチディスプレイ、マルチ入力タッチディスプレイ、マルチタッチディスプレイなどとして構成されていてもよい。本発明の範囲を逸脱することなく、いずれかの好適な表示デバイスおよびハプティックI/Oを使用することができる。

【0061】

コンピュータマウス、タッチパッドなどのようなキーボード530およびカーソル制御デバイス535は、ユーザーがコンピューティングシステム500とインターフェースすることを可能にするために、バス505にさらに結合されている。しかしながら、特定の実施形態では、物理的なキーボードおよびマウスが存在しない場合があり、ユーザーは、ディスプレイ525および/またはタッチパッド(図示せず)のみを介してデバイスとインタラクションすることができる。任意の入力デバイスの種類および組み合わせは、設計の選択の問題として使用され得る。特定の実施形態では、物理的な入力デバイスおよび/またはディスプレイは存在しない。例えば、ユーザーは、コンピューティングシステム500と通信している別のコンピューティングシステムを介してリモートでそれとインタラクションしてもよいし、コンピューティングシステム500は自律的に動作してもよい。

【0062】

メモリ515は、プロセッサ(複数可)510により遂行されたときに機能を提供するソフトウェアモジュールを格納する。モジュールは、コンピューティングシステム500のためのオペレーティングシステム540を含む。モジュールはさらに、本明細書に記載された処理の全てもしくは一部、またはその派生物処理を実行するように構成されたAI/MLモデルドリフト検出および修正モジュール545を含む。コンピューティングシステム500は、付加的な機能を含む1または複数の付加的な機能モジュール550を含み

得る。

【0063】

当業者であれば、「システム」は、本発明の範囲から逸脱することなく、サーバー、組み込みコンピューティングシステム、パーソナルコンピュータ、コンソール、パーソナルデジタルアシスタント(PDA)、携帯電話、タブレットコンピューティングデバイス、量子コンピューティングシステム、または他のいずれかの適切なコンピューティングデバイス、またはデバイスの組み合わせとして具現化され得ることを理解するであろう。上述した機能を「システム」により実行されるものとして提示することは、何ら本発明の範囲を限定することを意図するものではなく、本発明の多くの実施形態の一例を提供することを意図するものである。実際、本明細書に開示された方法、システム、および装置は、クラウドコンピューティングシステムを含むコンピューティング技法と整合性のあるローカライズされた形態および分配された形態で実装されてもよい。

10

【0064】

本明細書で説明するシステム特色のいくつかは、実装の独立性をより強調するために、モジュールとして提示されていることに留意すべきである。例えば、モジュールは、カスタムの非常に大規模な集積(VLSI)回路またはゲートアレイ、ロジックチップ、トランジスタ、または他の個別部品のような既製の半導体を含むハードウェア回路として実装され得る。また、モジュールは、フィールドプログラマブルゲートアレイ、プログラマブルアレイロジック、プログラマブルロジックデバイス、グラフィックス処理ユニットなどのプログラマブルハードウェアデバイスに実装され得る。

20

【0065】

モジュールはまた、様々なタイプのプロセッサにより遂行されるためのソフトウェアに少なくとも部分的に実装され得る。例えば、遂行可能コードの特定された単位は、例えば、オブジェクト、プロシージャ、または関数として編成されていてもよいコンピュータ命令の1または複数の物理的または論理的なブロックを含み得る。それにもかかわらず、遂行可能な特定されたモジュールは、物理的に一緒に配置されている必要はなく、論理的に結合されたときにモジュールを含み、モジュールのために述べられた目的を達成するために、異なる場所に格納された別々の命令を含んでいてもよい。さらに、モジュールは、例えば、ハードディスクドライブ、フラッシュデバイス、RAM、テープのような非一時的なコンピュータ読み取り可能な媒体、および/または本発明の範囲から逸脱することなくデータを格納するために使用される他のいずれかの非一時的なコンピュータ読み取り可能な媒体に格納されていてもよい。

30

【0066】

実際、遂行可能コードのモジュールは、単一の命令であってもよいし、多数の命令であってもよいし、さらには、複数の異なるコードセグメント、異なるプログラム間、および複数のメモリデバイス間に分散されていてもよい。同様に、動作データは、モジュール内で特定され、ここで示されてもよく、いずれかの適切なタイプのデータ構造体内でいずれかの適切な形態で具現化され、組織化され得る。動作データは、単一のデータセットとして収集されてもよいし、または異なる記憶デバイスにわたり異なる場所に分散されていてもよく、少なくとも部分的には、単にシステムまたはネットワーク上の電子信号として存在していてもよい。

40

【0067】

いくつかの実施形態では、AI/MLモデルのパフォーマンスに関する情報が取得される。この情報は、そのユーザーまたはコンピューティングシステムに特注されたローカルなAI/MLモデルについては単一のコンピューティングシステムから、より一般的に適用されることが意図されている(例えば、グループレベル、会社レベル、グローバルレベルなどでソリューションを提供する)AI/MLモデルについては複数または多数のコンピューティングシステムから、AI/MLモデル自体から、AI/MLモデルを監視するアプリケーションもしくは他のプロセスから、それらの任意の組み合わせなどから得られ得る。この情報は、AI/MLモデルが時間とともにどの程度うまく機能しているかを判

50

断するのに役立つ。例えば、情報は、限定されないが、どのような予測がなされたか（例えば、AI/MLモデルからの出力）、AI/MLモデルがいくつの予測を行い、いくつが実際に使用されたか、予測の1または複数の統計分布（例えば、正規分布、二項分布、または任意の他の統計分布）、AI/MLモデルに提供された入力データ、これらの任意の組み合わせなどを含み得る。いくつかの実施形態では、統計分布は、RPAロボットによってそのワークフローで取られるアクションにマッピングされる（例えば、アクティビティにマッピングされる）。

【0068】

この情報が収集された後、情報が解析されて、データドリフト、モデルドリフト、またはその両方が発生したかどうかを判断する。変化条件が見出される（例えば、予測値または入力データが過去の範囲から外れる）および/または変化しきい値が満たされるもしくはそれを越える（例えば、モデルの統計的パフォーマンスが過去のパフォーマンスから少なくとも一定量逸脱する）と、アラートまたは再訓練のトリガが生成される。いくつかの実施形態では、AI/MLモデルパフォーマンスがドリフトしたとみなされる前に、変化条件または変化しきい値が少なくともある期間またはインスタンスの数満たされなければならない。特定の実施形態では、元のデータセットと新しいデータセットの次元分布間の距離は、共変量ドリフトについて測定され得る。また、残留ドリフト間の距離は、AI/MLモデルが既に存在する場合は、そのバージョンで確認され得る。

10

【0069】

AI/MLモデルは、その後、AI/MLモデルパフォーマンスを改善することを試みるために、収集された情報および/または他の情報を使用して再訓練され得る。いくつかの実施形態では、再訓練されたAI/MLモデルが一定のしきい値を満たした場合、それは既存のAI/MLモデルと一緒に自動的に展開され、モデルのドリフトは、新しいバージョンのAI/MLモデルが実データ上でより良く機能しているかどうかを見るために共存して（side-by-side）測定され得る。特定の実施形態では、新しいバージョンのAI/MLモデルは、それがこれらのしきい値を満たさない限り、展開されない。

20

【0070】

いくつかの実施形態では、カナリアテストが使用される。例えば、新しいバージョンのAI/MLモデルは、AI/MLモデルの変化を知らない可能性がある比較的少数のユーザー（例えば、地理的または他の何らかのフィルタによる一度に10%のユーザー）によって使用され得る。新しいバージョンのAI/MLモデルが以前のバージョンと同様にうまく機能していない場合、新しいバージョンはロールバックされ、再訓練が実行され得る。

30

【0071】

特定の実施形態では、チャンピオン/チャレンジャー技法が使用されてもよく、ここで、入力データは、新しいバージョンのAI/MLモデルと以前のバージョンのAI/MLモデルの両方に送られるが、出力の1つのセットのみが使用される。これは、新しいバージョンのAI/MLモデルと以前のバージョンのAI/MLモデルがどのように機能するかを、いくつかのグラウンドトゥルース（ground truth）に基づいて同一条件の下で解析するのに役立つ。この技法は、どのAI/MLモデルバージョンが適切であるかまたはより適切に近いのかをより頻繁に判断するのに役立つ。また、この技法は、以前のバージョンのAI/MLモデルが特定の動作を示していた理由に関するトラブルシューティングに役立つ情報を提供し得る。

40

【0072】

いくつかの実施形態では、どの情報がワークフローアクティビティからRPAロボットによって呼び出されおよび/またはワークフローアクティビティ内でRPAロボットによって遂行されるAI/MLモデルに供給されるかに関して、自動化プラットフォームによって共通のフレームワークが提供され、強制される。このフレームワークは、RPAプラットフォームで使用されるAI/MLモデルが意図したデータを確実に消費するのに役立つ。このフレームワークは、特定のフィールド要件および/または値を強制するために、AI/MLモデルが適切な形式で入力を受信することを確実にするのに役立つ。

50

【0073】

図6は、本発明の実施形態による、RPAのためのAI/MLモデルドリフト検出および修正を実行するように構成されたシステム600を示すアーキテクチャ図である。システム600は、デスクトップコンピュータ602、603、タブレット604、スマートフォン606などのユーザーコンピューティングシステムを含む。しかしながら、スマートウォッチ、ラップトップコンピュータ、インターネットオブシングス(IoT)デバイス、車両コンピューティングシステムなどを含むが、これらに限定されない、本発明の範囲から逸脱しない、いずれかの所望のコンピューティングシステムが使用され得る。また、3つのユーザーコンピューティングシステムが図6に示されているが、本発明の範囲から逸脱することなく、いずれかの好適な数のコンピューティングシステムが使用され得る。例えば、いくつかの実施形態では、数十、数百、数千、または数百万のコンピューティングシステムが使用されてもよい。

10

【0074】

各コンピューティングシステム602、604、606は、MLモデル(例えば、サーバー630のAI/MLモデル632のうちの一つ、コンピューティングシステム602、604、606上のローカルMLモデル、および任意の他の場所に位置するMLモデル等)を呼び出すおよび/または遂行するRPAロボット610をその上で動作させている。RPAロボット610は、いくつかの実施形態では、RPAデザイナーアプリケーションを介して生成されたロボットであり得る。この実施形態では、RPAロボット610は、AI/MLモデル632に要求、および潜在的に入力データを送信する。しかし、いくつかの実施形態では、AI/MLモデル632は、RPAロボット610からの入力データに加えまたはそれに代えて、1または複数の他のソース(例えば、データベース640、別のコンピューティングシステムなど)から入力データを取得し得る。

20

【0075】

RPAロボット610は、ネットワーク620(例えば、ローカルエリアネットワーク(LAN)、移動通信ネットワーク、衛星通信ネットワーク、インターネット、それらの任意の組み合わせなど)を介して、サーバー630のAI/MLモデル632にAI/MLモデル要求を送信し、サーバー630のAI/MLモデル632からAI/MLモデル遂行結果を受信する。サーバー630は、AI/MLモデル632のためのモデルドリフトに関連する情報を収集し、データベース640に格納する。この情報は、データドリフト、モデルドリフト、またはその両方に関連し得る。

30

【0076】

AI/MLモデル632の一つについて、変化条件が見出される(例えば、予測または入力データが過去の範囲から外れる)および/または変化しきい値が満たされるもしくはそれを越える(例えば、モデル統計的パフォーマンスが少なくとも一定量だけ過去のパフォーマンスから逸脱する)場合、いくつかの実施形態では、サーバー630は、レビューアコンピューティングシステム650上で実行しているアプリケーション652に(例えば、レビューおよびラベリングのためのキューの一部として)アラートまたは再訓練トリガを送信する。いくつかの実施形態では、アプリケーション652は、レビューアがAI/MLモデルドリフトに関する情報を閲覧し、結果を修正し(例えば、モデルドリフトについて)、および/または入力データが本質的に変化したこと(例えば、データドリフト)を認識できるようにし得る。いくつかの実施形態では、アプリケーション652は、本番環境中のMLモデルを「所有」するML操作担当者が、モデルドリフトを二次元グラフで見ることが可能にし得る。例えば、PCAを使用して次元削減が実行され、特定の異常を視覚的に示し得る。これらの異常のうちの一つが発生した場合にフラグを立てるために、特定のアラートも作成され得る。

40

【0077】

いくつかの実施形態では、データが、ビジネスユーザーがデータの意味を理解できるような方法でビジネスユーザーに提示される(例えば、ビジネスデータの解析がどのように変化したかをビジネスユーザーに示す2Dグラフ)アプリケーション652における抽象

50

化の層はより多く存在し得る。これは、より技術的に精通したML操作担当者向けの表示に加えまたはそれに代えて提示され得る。例えば、ワードクラウド(word cloud)のようなメカニズムを使用して、より頻繁に発生する結果をより大きなフォントで表示し、人間がどの結果が最も広く認められているかをより簡単に確認できるようにし得る。

【0078】

特定の実施形態では、重要な変化が強調されることがある。例えば、ビジネスユーザーは、ローンを承認するためのAI/MLモデルが、1日に承認された5つのローンから1日に承認された20のローンに跳ね上がったことを見ることができる場合がある。その後、ビジネスユーザーは、問題が解決しない場合に再訓練が行われるまでずっと、モデルの動作を修正するためのルールを定義し得る。ビジネスユーザーは、AI/MLモデルのパフォーマンスが原因ではない(例えば、ローン申請の数が増加した、より多くの適格な申請者が申請しているなど)異常があったかどうかを判断することもでき得る。人間は、アプリケーション652を使用して、自動化ドリフト検出メカニズムによって検出されなかった逸脱を引き起こしている変化を自分自身で判断することもできる。いくつかの実施形態では、1つは技術に精通したユーザー用と他方はビジネスユーザー用の2つのレイヤーが存在し得る。

10

【0079】

特定の実施形態では、コンピューティングシステム650およびアプリケーション652を含む複数または多数のコンピューティングシステムおよびレビューアプリケーションが、結果をまとめてレビューおよび修正するために使用され得る。いくつかの実施形態では、これらの結果は、共通のキューから引き出される。次いで、これらの修正は、サーバー630に戻って提供され、ドリフトAI/MLモデル632を再訓練するためにデータベース640に格納され得る。しかし、特定の実施形態では、サーバー630は、入力データの変更および/またはAI/MLモデル予測における変更によるAI/MLモデルドリフトが発生していることを自動的に検出し得る。

20

【0080】

AI/MLデータおよび/またはモデルドリフトがサーバー630によって検出された後(ならびにいくつかの実施形態では人為的な修正後)、ドリフトAI/MLモデル632は再訓練される。いくつかの実施形態では、再訓練は、サーバー630によって実行される。しかし、特定の実施形態では、再訓練は、サーバー630に加えてまたはサーバー630の代わりに、1または複数の他のサーバーによって実行され得る。ドリフトAI/MLモデル632は、その後、AI/MLモデルパフォーマンスを改善することを試みるために、収集された情報および/または他の情報を使用して再訓練され得る。

30

【0081】

新しいバージョンのAI/MLモデル632が適切に訓練された後(例えば、再訓練されたAI/MLモデル632が特定のしきい値を満たす場合)、サーバー630は、新しいバージョンのAI/MLモデル632をデータベース640に保存し、以前のバージョンのAI/MLモデルと置き換えることが可能である。しかし、特定の実施形態では、AI/MLモデル632の複数のバージョンが格納され得る。次いで、サーバー630は、RPAロボット610によって呼び出されるように再訓練されたAI/MLモデル632を展開する。しかし、いくつかの実施形態では、再訓練されたAI/MLモデル632は、AI/MLモデル632の1または複数の以前のバージョンと一緒に展開される。AI/MLモデル632の各バージョンのドリフトは、その後、新しいバージョンのAI/MLモデル632が実データ上でより良いパフォーマンスをしているかどうかを見るために、サイドバイサイドで測定される。これが経時的にそのようなケースであると判断される場合、新しいバージョンのAI/MLモデル632は、以前のバージョンのAI/MLモデル632と置き換え得る。しかし、そうでない場合、新しいバージョンのAI/MLモデルは廃棄されてもよく、またはさらなる再訓練が実行され得る。

40

【0082】

図7Aおよび図7Bは、本発明の実施形態による、RPAのためのAI/MLモデルド

50

リフト検出および修正を実行するための処理700を示すフローチャートである。プロセスは、705において、RPAロボットからAI/MLモデルを遂行する要求およびAI/MLモデルのための入力情報を受信することから始まる。入力情報がAI/MLモデルのフレームワークに適合していることを検証することが710で実行される。715において、入力情報がフレームワークに適合していない場合、720で、入力情報がフレームワークに適合していないことを示すエラーがRPAロボットに提供される。715において、入力情報がフレームワークに適合する場合、AI/MLモデルは入力情報を使用して遂行され、AI/MLモデルの遂行の結果は725でRPAロボットに返される。いくつかの実施形態では、フレームワークは、複数のAI/MLモデルのために強制される。

【0083】

データドリフトが発生したか否かを判断するためのAI/MLモデルの入力データに関する情報、モデルドリフトが発生したか否かを判断するためのAI/MLモデルの実行の結果に関する情報、またはその両方が730で解析される。いくつかの実勢形態では、解析は、共変量ドリフトについて、元のデータセットと新しいデータセットの次元分布間の1または複数の距離を測定することを含む。特定の実施形態では、解析は、統計モーメントが変化するかどうかを判断することを含む。いくつかの実施形態では、統計モーメントは、平均、分散、歪度、尖度、共分散、またはそれらの組み合わせを含む。特定の実施形態では、情報は、AI/MLモデルがどのような予測を行ったか、いくつかの予測をAI/MLモデルが行ったかおよびRPAロボットによっていくつかの予測が使用されたか、AI/MLモデルによって行われた予測の1もしくは複数の統計分布、AI/MLモデルに提供された入力データ、またはそれらの組み合わせを含む。いくつかの実施形態では、情報の解析は、複数のそれぞれのRPAロボットによって遂行される複数のRPAプロセスにわたって実行される(例えば、連合学習を介して)。

【0084】

いくつかの実施形態では、情報は、AI/MLモデルによって行われた予測の1または複数の統計分布を含む。そのような実施形態では、1または複数の統計分布は、735で複数のRPAロボットによって取られたそれぞれのアクションにマッピングされ得る。いくつかの実施形態では、アクションは、それぞれのRPAロボットのRPAワークフローのそれぞれのアクティビティで行われる。

【0085】

情報の二次元表現を生成するために、情報に対して次元削減技術が実行され、情報の二次元表現が740で表示される。745での情報の解析に基づいて、変化条件が見出され、変化しきい値が満たされているもしくはそれを超えているか、またはその両方であるかどうかの検証が実行される。いくつかの実施形態では、検証は、変化条件、変化しきい値、またはその両方が、少なくとも所定の期間、満たされているまたはそれを超えているかを検証することを含む。

【0086】

750での情報の解析に基づいて、変化条件が見出され、変化しきい値が満たされているもしくはそれを超えているか、またはその両方である場合、755でAI/MLモデルが再訓練される。次に、760で再訓練されたAI/MLモデルが1または複数パフォーマンスしきい値を満たしていることが検証される。765で1または複数のパフォーマンスしきい値が満たされない場合、プロセスは755でAI/MLモデルの再訓練に戻る。しかし、765で再訓練されたAI/MLモデルが1または複数のパフォーマンスしきい値を満たした場合、770で再訓練されたAI/MLモデルが展開される。いくつかの実施形態では、AI/MLモデルを展開することは、1または複数のRPAロボットの1または複数のアクティビティを変更して新しいAI/MLモデルを呼び出すこと、以前のバージョンのAI/MLモデルを再訓練されたAI/MLモデルに置き換えることを含み得る。特定の実施形態では、AI/MLモデルは、カナリアテスト技法またはチャンピオン/チャレンジャーテスト技法を用いて展開される。

【0087】

10

20

30

40

50

再訓練されたAI/MLモデルおよび以前のバージョンのAI/MLモデルのモデルドリフトは、実データを使用して775で測定される。再訓練されたAI/MLモデルが、780において測定されたモデルドリフトに基づいて、以前のバージョンのAI/MLモデルよりも実データに対して良好に実行しない場合、再訓練されたAI/MLモデルは、785において破棄され、および/またはプロセスは、さらなる再訓練のためのステップ755に戻る。しかし、再訓練されたAI/MLモデルが、780において測定されたモデルドリフトに基づく以前のバージョンのAI/MLモデルよりも実データに対して良好に実行する場合、再訓練されたAI/MLモデルは、790において以前のバージョンのAI/MLモデルに加えてまたはその代わりに使用される。入力データおよび/またはユースケースに基づいて、再訓練されたAI/MLモデルまたは以前のバージョンのAI/MLモデルが、795で選択され、呼び出される。いくつかの実施形態では、選択は、階層的な方法で行われる。

10

【0088】

いくつかの実施形態では、モデルドリフト検出は、プロセスごとまたはプロセスのグループごとに実行され得る。例えば、あるタイプの処理のために、AI/MLモデルのあるバージョンを使用することが良い場合がある。AI/MLモデルがある特性を有する入力データに基づいて訓練されると、AI/MLモデルは、別のタイプの入力データに対して正確でなくなることが観察され得る。例えば、欧州の日付形式（例えば、17 November 2019または17/11/2019）を使用して訓練された請求書の処理に使用されるAI/MLモデルがある場合について考える。訓練中に、米国の日付形式を持つ請求書がより多く使用された場合（例えば、November 17, 2019または11/17/2019）、AI/MLモデルは、欧州の日付形式を持つ請求書に対して失敗し始める場合がある。

20

【0089】

したがって、いくつかの実施形態では、AI/MLモデルの異なるバージョンが、異なる状況に対応するように訓練されてもよく、ドリフト検出が、与えられたデータに対して最もよく機能するAI/MLモデルバージョンを選択するために使用されてもよい。特定の実施形態では、AI/MLモデルバージョンは、RPAワークフローレベルでのドリフト検出に基づいて、階層的な方法で適用されてもよい。入力のタイプに基づいてどのAI/MLモデルバージョンを使用するかを判断するためにロジックが開発されてもよく、または複数のAI/MLモデルが階層的な方法で試され、特定の基準が満たされた場合に結果が選択されてもよい。これはまた、複数のAI/MLモデルのバージョンでどのようなドリフトが発生しているかの見通しを提供し得る。これは、AI/MLモデルが、あるユースケースには悪いが、他のユースケースにはそうではないかどうかを判断するのに役立つ。特定の実施形態では、AI/MLモデルチェーンプロセスは、決定論的論理が階層で使用されるAI/MLモデルバージョンを支配するように構築され得る。いくつかの実施形態では、別のAI/MLモデルが、どのAI/MLモデルバージョンを使用するかを判断し得る。特定の実施形態では、複数のコンピューティングシステムが、ローカルデバイス上に訓練データを保持しながら、共有された予測モデルを共同して学習する、連合学習（federated learning）が使用され得る。これは、RPAのガバナンス要件（例えば、EU一般データ保護規則（the E.U. General Data Protection Regulation）（GDPR）、米国医療保険の相互運用性と説明責任に関する法律（the U.S. Health Insurance Portability and Accountability Act）（HIPAA）、サードパーティの利用規約など）への準拠を保証するために特に有益であり得る。

30

40

【0090】

図7Aおよび7Bで実行されるプロセスステップは、本発明の実施形態に従って、図7Aおよび7Bで説明したプロセス（複数可）の少なくとも一部を実行するようにプロセッサ（複数可）への命令をエンコードするコンピュータプログラムによって実行されてもよい。コンピュータプログラムは、非一時的なコンピュータ読み取り可能な媒体に格納され

50

ていてもよい。コンピュータ読み取り可能な媒体は、ハードディスクドライブ、フラッシュデバイス、RAM、テープ、および/またはデータを格納するために使用される他のそのような媒体または媒体の組み合わせであってもよいが、これらに限定されるものではない。コンピュータプログラムは、図7Aおよび7Bに記載されたプロセスステップの全部または一部を実装するために、コンピューティングシステム（例えば、図5のコンピューティングシステム500のプロセッサ（複数可）510）のプロセッサ（複数可）を制御するためのコード化された命令を含んでもよく、これはまた、コンピュータ読み取り可能な媒体に格納されてもよい。

【0091】

コンピュータプログラムは、ハードウェア、ソフトウェア、またはハイブリッド実装で実装され得る。コンピュータプログラムは、互いに動作可能な通信を行うモジュールで構成され得、情報または指示をディスプレイに送るように設計されている。コンピュータプログラムは、汎用コンピュータ、ASIC、またはいずれかの他の好適なデバイスで動作するように構成され得る。

10

【0092】

本発明の様々な実施形態のコンポーネントは、本明細書に一般的に記載され、図示されているように、様々な異なる構成で配置され、設計されてもよいことが、容易に理解されるであろう。したがって、添付の図に表されるような本発明の実施形態の詳細な説明は、特許請求されるような本発明の範囲を限定することを意図するものではなく、本発明の選択された実施形態を代表するものにすぎない。

20

【0093】

本明細書を通して記載された本発明の特色、構造、または特徴は、1または複数の実施形態では、いずれかの好適な方法で組み合わせられ得る。例えば、本明細書全体を通して「特定の実施形態」、「いくつかの実施形態」、または類似の言語を参照することは、実施形態に関連して記載された特定の特色、構造、または特徴が、本発明の少なくとも1つの実施形態に含まれていることを意味する。したがって、本明細書全体を通して「特定の実施形態では」、「いくつかの実施形態では」、「他の実施形態では」、または類似の言語の出現は、必ずしも全ての実施形態の同じグループを指すものではなく、記載された特色、構造、または特徴は、1または複数の実施形態ではいずれかの好適な方法で組み合わせられ得る。

30

【0094】

本明細書全体を通して特色、利点、または類似の言語への参照は、本発明で実現され得る特色および利点の全てが、本発明のいずれかの単一の実施形態にあるべきであること、または本発明のいずれかの実施形態であることを意味するものではないことに留意すべきである。むしろ、特色および利点に言及する言語は、実施形態に関連して記載された特定の特色、利点、または特徴が、本発明の少なくとも1つの実施形態に含まれることを意味すると理解される。したがって、本明細書全体での特色および利点の議論、ならびに類似の言語は、同じ実施形態を参照することができるが、必ずしもその必要性はない。

【0095】

さらに、本発明の記載された特色、利点、および特徴は、1または複数の実施形態では、いずれかの好適な方法で組み合わせることができる。関連する技術の当業者は、本発明が、1または複数の特定の実施形態の特定の特徵または利点なしに実施され得ることを認識するであろう。他の例では、追加の特徵および利点は、本発明の全ての実施形態には存在しないかもしれないが特定の実施形態では認識され得る。

40

【0096】

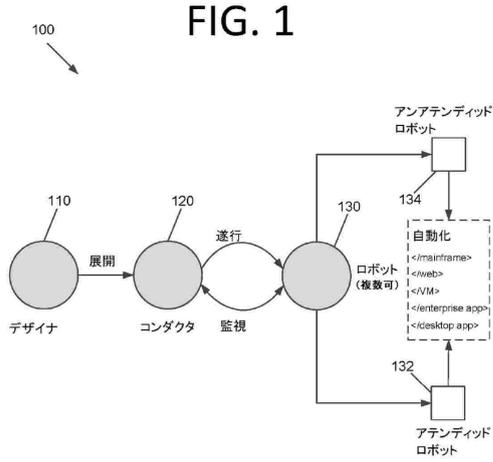
本分野における通常の技術を有する者は、上述したような本発明を、異なる順序でのステップを用いて、および/または開示されているものとは異なる構成のハードウェア要素を用いて実施することができることを容易に理解するであろう。したがって、本発明は、これらの好ましい実施形態に基づいて説明されてきたが、本発明の精神および範囲内にとどまりながら、特定の変更、変形、および代替的な構成が明らかになることは、当業者に

50

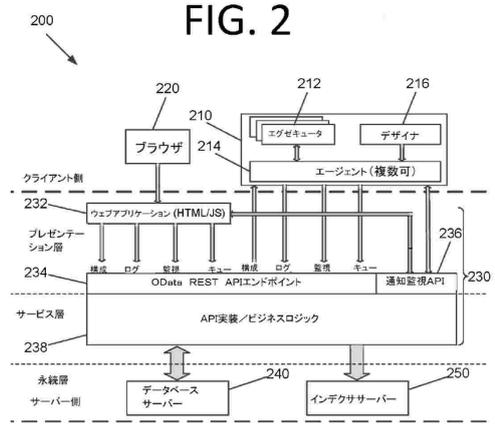
は明らかであろう。したがって、本発明の範囲を決定するためには、添付の特許請求の範囲を参照すべきである。

【図面】

【図 1】



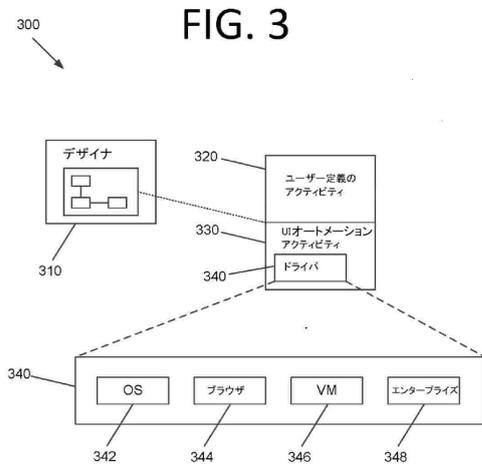
【図 2】



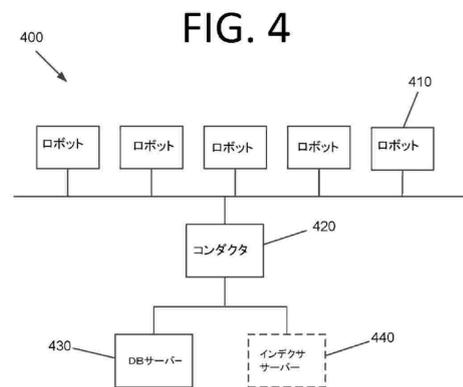
10

20

【図 3】



【図 4】

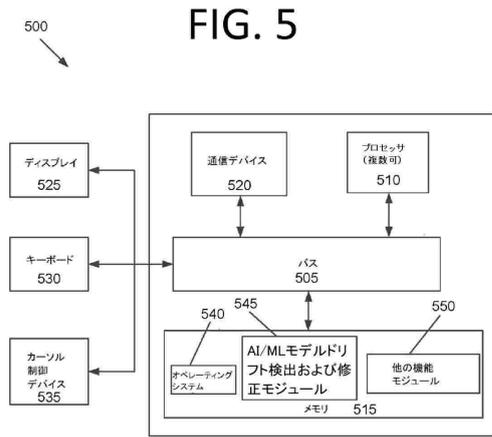


30

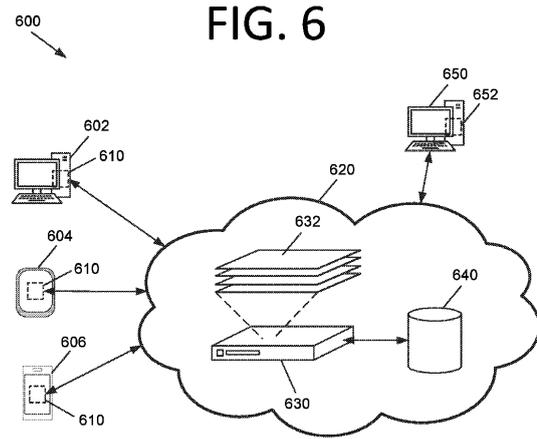
40

50

【図5】



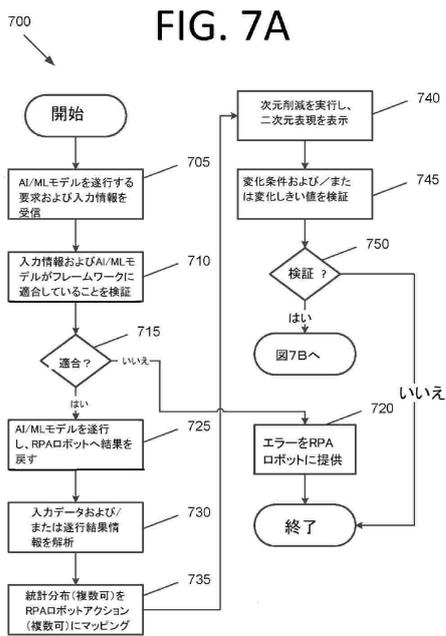
【図6】



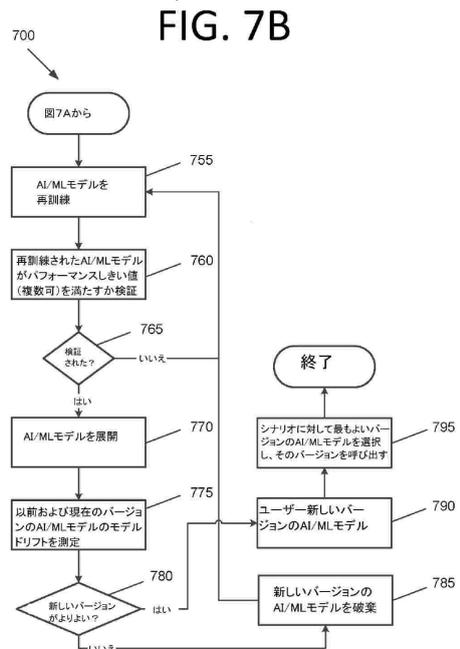
10

20

【図7A】



【図7B】



30

40

50

フロントページの続き

- 90, トウェンテース フロア
(72)発明者 マクゴネル, アントン
アメリカ合衆国 10016 ニューヨーク, ニューヨーク, パーク アベニュー 90, トウェン
テース フロア
- (72)発明者 ヒダルゴ, マルコ, アルバン
アメリカ合衆国 10016 ニューヨーク, ニューヨーク, パーク アベニュー 90, トウェン
テース フロア
- 審査官 多賀 実
- (56)参考文献 米国特許出願公開第2017/0330109(US, A1)
米国特許出願公開第2019/0180746(US, A1)
- (58)調査した分野 (Int.Cl., DB名)
- G06N 3/00 - 99/00
G06F 18/00 - 18/40
G06F 8/00 - 8/77
G06F 9/44 - 9/455
G06Q 10/00 - 10/30