

AIと社会



AIと社会

・AIと法・倫理

AIを活用していく上で法の遵守・倫理の遵守も重要になってくる

→AIの責任について法律で定めていくことが大切になる

→AIが操作した機械が人を傷付けたとき誰が責任を取るのかなど

法律で定めておかないと被害者が泣き寝入りしてしまう

→責任のあるAIを開発していくことが求められている

| AIと社会

・データ量の増加

インターネット普及により大量のデータ（ビックデータ）を獲得することができるようになったことでAIも進歩した
→人間が行っていた仕事をAIが行うという時代になってきている

→定型的な仕事はAIが行った方が効率的であることが多い

AIと社会

・データ量の増加

AIがいきなり人間の仕事を行うことは難しい

→大量のデータがないとAIを学習させることができないため

人間が仕事しているアナログ空間のデータを蓄積する必要がある

→データ蓄積の方法として、RPAやIoTを駆使していく方法がある

AIと社会

- RPA

Robotic Process Automationの略で、

定型的な業務を、自動化する技術のこと

→毎月15日になつたら、請求書を作成して顧客に送信する など

→ルールの定まっていない業務は自動化することは難しい

| AIと社会

- IoT

Internet of Thingsの略で、

モノがインターネットに接続する技術のこと

→家電・車・住居など様々なモノがインターネットに接続することで、利便性の向上に繋がると同時に多くのデータを収集することができる

AIと社会

- ・デジタルデータの弱点

デジタルデータは書き換えが容易であるため

ブロックチェーンを活用して改ざんを防ぐことも必要である

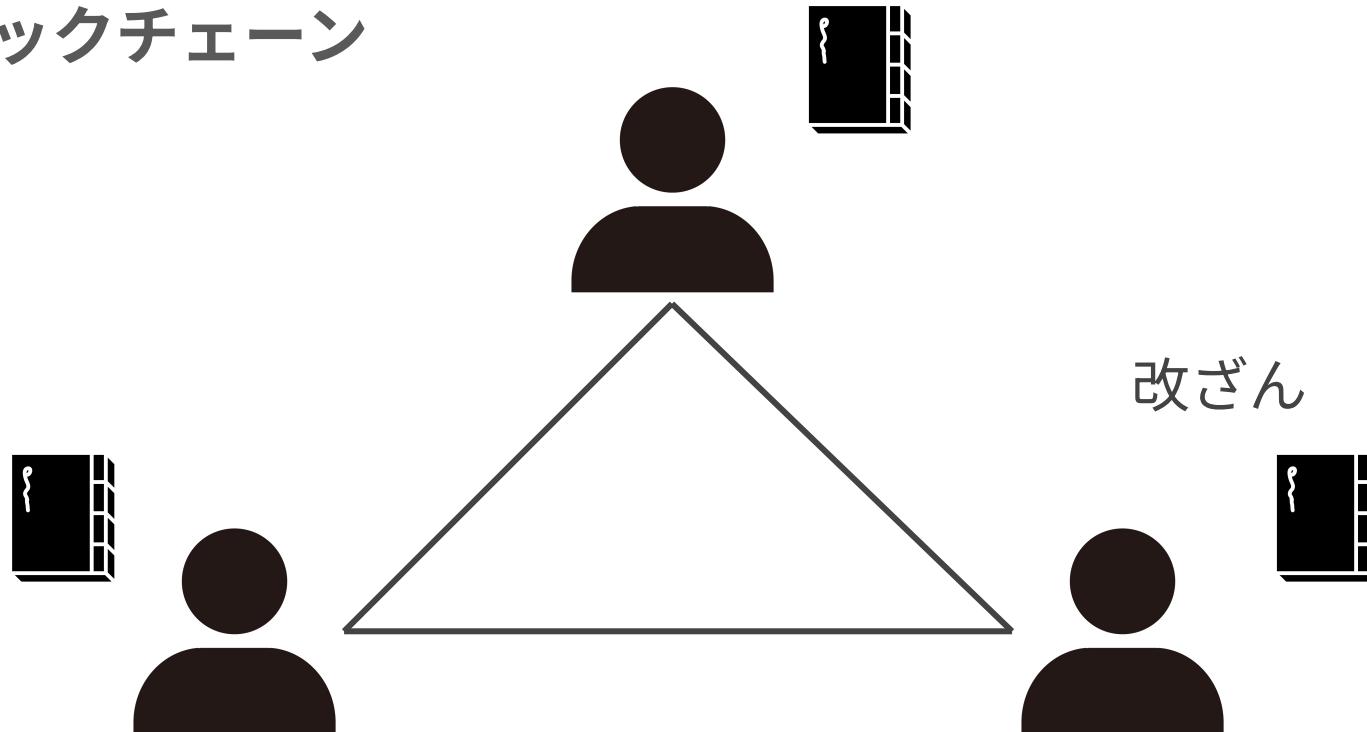
- ・ブロックチェーン（分散型台帳）

過去に行った全ての取引データ（台帳）を多くの人が

保有することでデータの改ざんを実質的に不可能にする

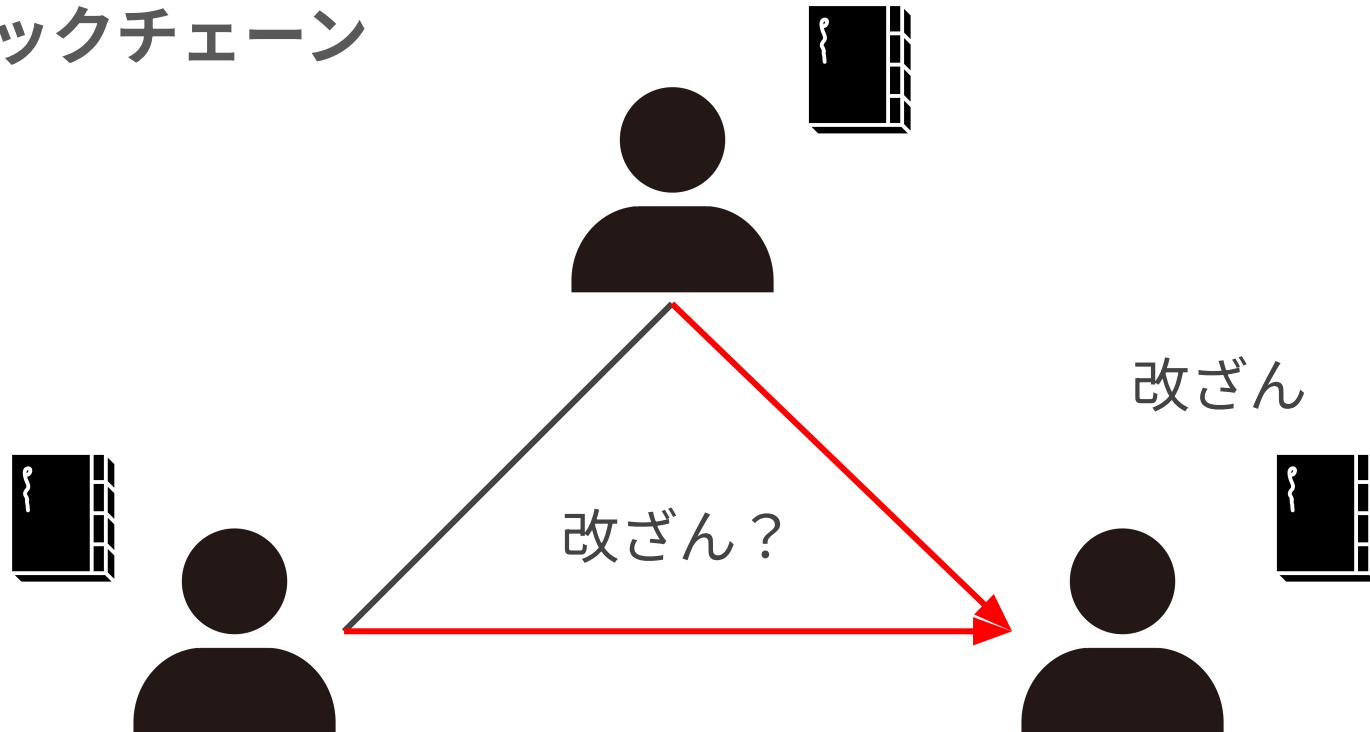
| AIと社会

- ・ブロックチェーン



| AIと社会

- ・ブロックチェーン





AIのビジネス活用

| AIのビジネス活用

・AIのビジネス活用

AI技術を普及させるためには利益の創出は重要なポイントになる
→利益が上がらない場合はAIの活用に対して消極的になる
利益が上がらないことに対して投資できる企業は少ない

→AIを上手く活用できればイノベーションが生まれやすくなる
今までない製品やサービスを創造しやすい

| AIのビジネス活用

・AIのビジネス活用

- 今まで1時間かかっていた作業が数十秒で終わるなど
→文章を要約するのに1時間かかっていたものが、
AIを活用することで数十秒で要約が終わる
- 1つの文章で、用途別に文章を作り変えることができる など
人が勝てない速さで作業を終わらすことが可能

| AIのビジネス活用

・AIのビジネス活用

- ビジネスを考えるときはAI以外の選択肢も考えることが大切
→ルールベースのシステムで問題や課題を解決できることも多い
- メリットが小さい場合は、AIを導入しない選択肢もある
- AIを活用したサービスを作る場合などは、
あらかじめプロジェクトの全体像を考えて行動する必要がある

| AIのビジネス活用

- ・プロジェクトの全体像
- ・プロジェクトの企画・計画
- ・データの収集・加工・分析
- ・アセスメント・コンセプト検証
- ・学習・運用



詳しく解説



AIプロジェクトの進め方

| AIプロジェクトの進め方

・AIプロジェクトの全体像

AIプロジェクトにおいて**モデル開発**はプロセスの一部
→データを分析したり、システムの保守管理を行なったり
モデル開発以外でも行うべきことは数多く存在する

→AIプロジェクトを成功に導くために多くのフレームワークがある
データマイニングのフレームワークとして**CRISP-DM**がある

| AIプロジェクトの進め方

- CRISP-DM

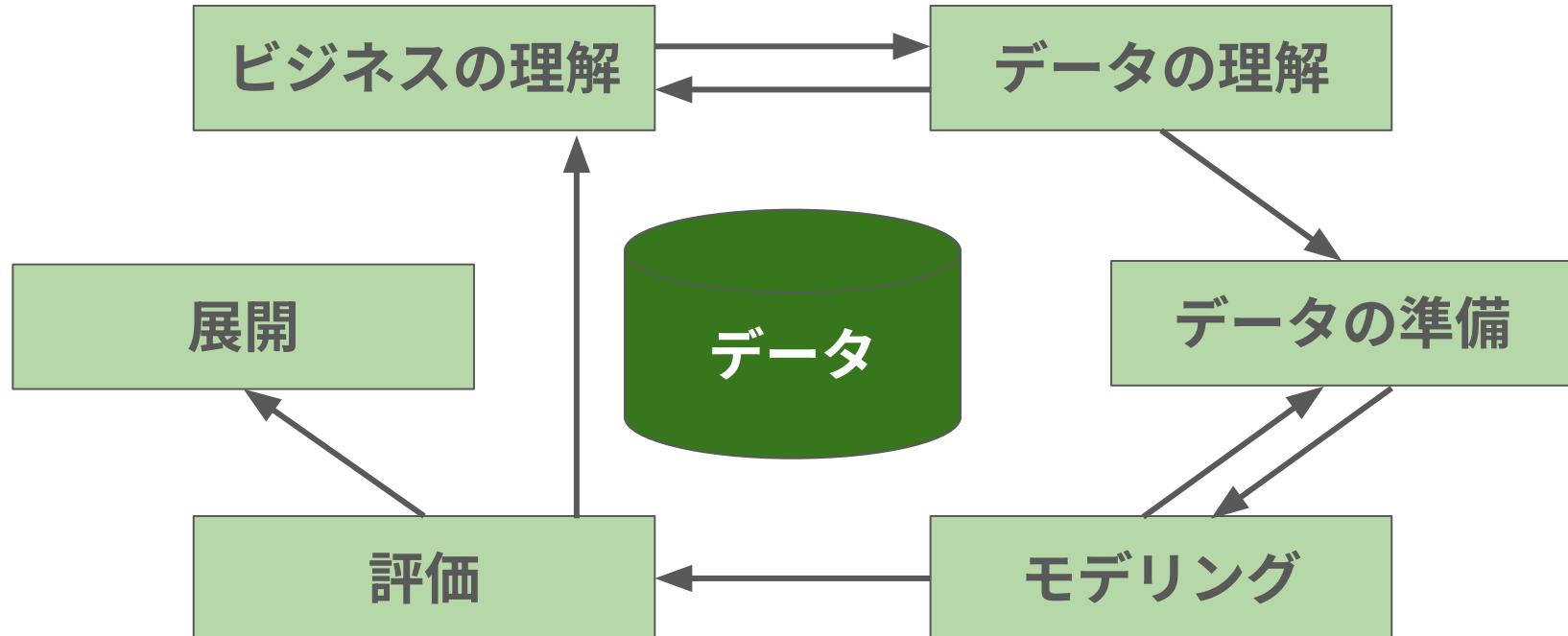
データマイニングのための指針となる手法・型のこと
→6つのステップから構成されている

ビジネスの理解、データの理解、
データの準備、モデリング、評価、展開

→それぞれのステップを行き来しながらデータ分析を行う

| AIプロジェクトの進め方

- CRISP-DM



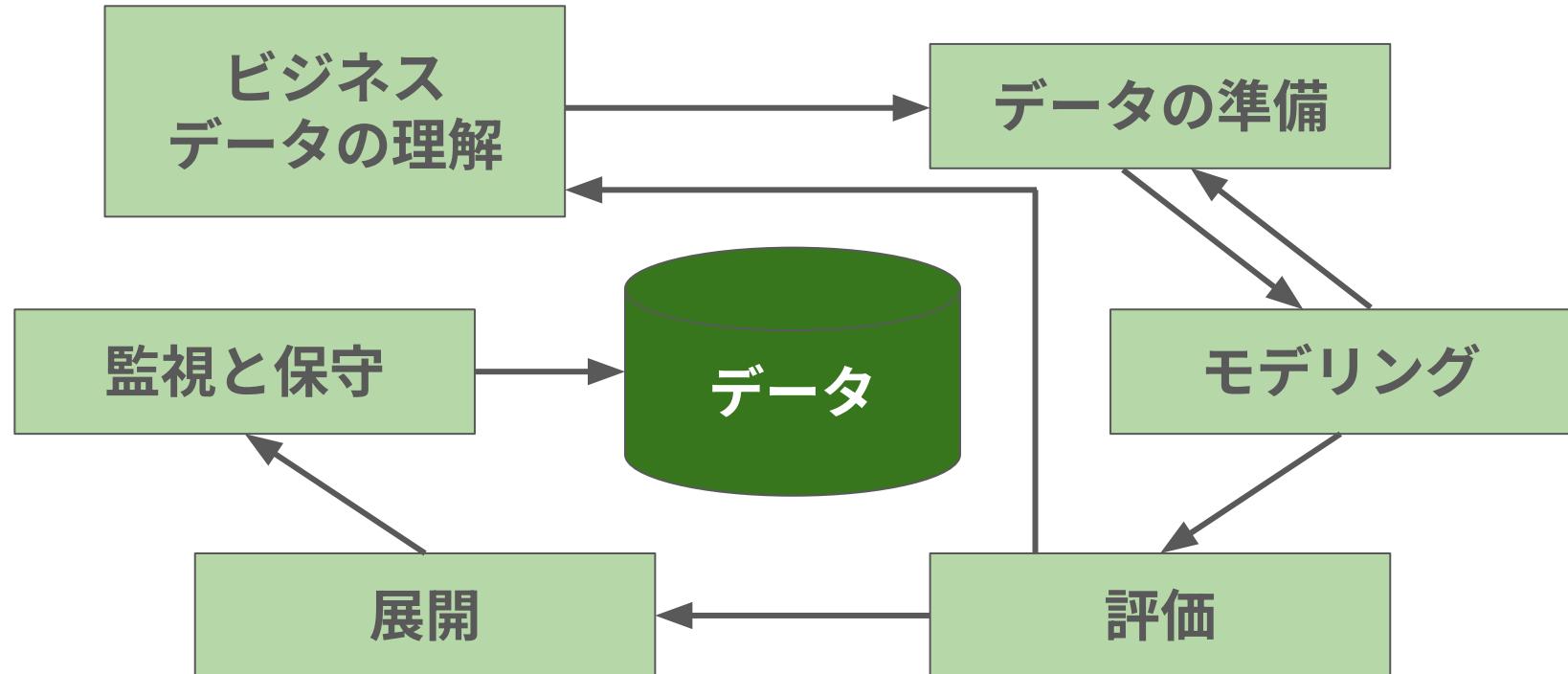
| AIプロジェクトの進め方

- CRISP-ML

CRISP-DMを発展させたデータマイニングに関する手法・型

- ビジネスの理解とデータの理解が統合
- 監視と保守が追加

| AIプロジェクトの進め方



| AIプロジェクトの進め方

- AIプロジェクトの進め方

- AIの導入検討

プロジェクトによってはコスト・利益の面から

AIを使わぬ方が良い場合がある

→AIを開発するためにデータを収集することができるのかも大切

→AIを導入することが全てではない、最適な選択肢を採用する

| AIプロジェクトの進め方

- AIプロジェクトの進め方

- データの蓄積と改善

最初から精度の高いAIを開発するのは困難である

→AIを改善しながら、精度の高いAIを提供していく

AIの精度を高めるためには、データを蓄積する必要がある

→データをどのように蓄積していくのかを事前に考える

| AIプロジェクトの進め方

- AIプロジェクトの進め方

- AI活用の土壤

社内でAIを活用していくためには、

AIを活用できる体制を作ることが大切になる

→ビジネスプロセスを再設計することを

BPR (Business Process Re-engineering) という

| AIプロジェクトの進め方

- **BPR (Business Process Re-engineering)**

AIを活用するためにビジネスプロセスを再設計することも大切
→人に特化したビジネスプロセスをAIを活用することを
前提にしたビジネスプロセスに変更していく

→人しか行えない作業とAIが行える作業が同じ作業内にある場合
人とAIの作業を分割することでAIを活用することができる

| AIプロジェクトの進め方

・AIサービスの提供方法

- AIの性質上、冷蔵庫やテレビのように提供して終わりよりも
サービスとして長期的に提供し続ける方が現実的である
→初期のAIは、荒削りで予測などが大雑把であることが多い
時間をかけてAIの精度を高めていくことが求められる
- AIサービスをどのように提供するのかを考えることも大切である

| AIプロジェクトの進め方

・AIサービスの提供方法

AIサービスを提供する方法として以下のような方法がある

- ・クラウドサービスとして提供する
 - ・Web APIとして提供する
 - ・エッジデバイスとして提供する
- などなど

| AIプロジェクトの進め方

・クラウドサービス

クラウド上でAIを活用したWebサービスを定額などで提供

→AIの更新を定期的に行い、Webサービスの価値を高めていく

→最初から完成度の高いWebサービスを提供することは

現実的に不可能なので、更新を繰り返して完成度を高める

→YouTube、SNS、LINEなどのように繰り返し更新していく

| AIプロジェクトの進め方

- **Web API**

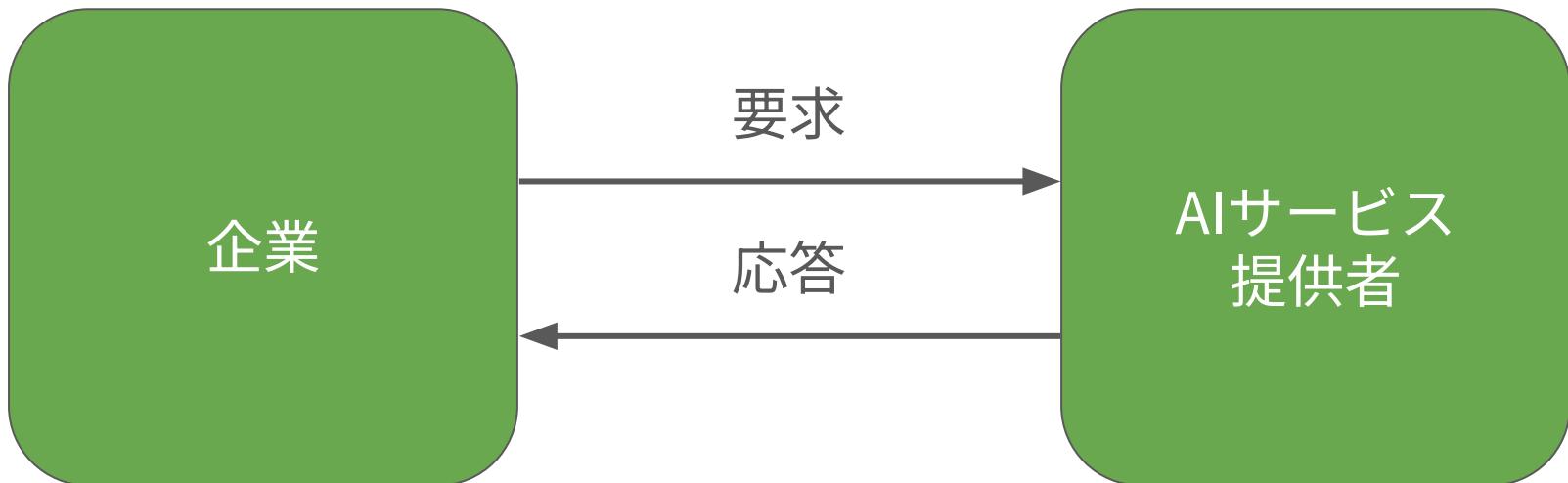
AIシステムを簡単に使える**Web API**を提供し、
使用回数などに応じて報酬をもらう
→定期的にAIを更新し、精度を高めていく

→**Web API**とは**HTTP・HTTPS通信**によってやりとりする**API**
APIとは外部機能などを使うための仕組みのこと

| AIプロジェクトの進め方

- **Web API**

インターネットを活用して、外部機能を使う仕組みのこと



| AIプロジェクトの進め方

- エッジデバイス

AIモデルをダウンロードしたエッジデバイスを提供し、
提供後もモデルを定期的に更新し続ける

→テスラ社の自動車はAIモデルが導入されている

AIモデルが定期的に更新され、高精度になっていく

→AIサービスなどは継続的に提供する方式が現実的である

| AIプロジェクトの進め方

- デプロイ

コードが書かれたファイルをサーバ上に配置すること



| AIプロジェクトの進め方

- AIサービスの提供方法

システムを開発しながら運用していく開発手法や考え方を
DevOps (DevelopmentとOperationsの造語) という

→特にAIを開発しながら運用していく開発手法や考え方を
MLOPs (Machine LearningとOperationsの造語) という

プロジェクト体制の構築



| プロジェクト体制の構築

・プロジェクト体制の構築

AIシステムを構築する上で様々なスキルをもった人が必要になる
→マネージャー、プログラマー、エンジニア、広報、経営企画、
データサイエンティスト、デザイナー など

・データサイエンティスト

データ分析やモデル構築などの仕事を行う人のこと

| プロジェクト体制の構築

- プロジェクト体制の構築

- デザイナー

UIやUXを意識したデザインを行うことが求められる

→精度の高いAIを使用したサービスでも

UIやUXが悪いと、利用されない可能性が高い

→AIを導入したサービスは、使いやすさなどを意識する必要

| プロジェクト体制の構築

- **UI (User Interface)**

ユーザがソフトなどを操作するために接する部分のこと

→Webサイト、アプリ画面、機械の操作画面 など

- **UX (User Experience)**

商品を通して得られるユーザー体験のこと

| プロジェクト体制の構築

顧客体験（CX : Customer Experience）の高いサービスを提供することが大切でありCXに責任を持つ人物を組織に入れる
→CXが低いサービスは継続利用されにくい特徴がある

- CX (Customer Experience)

顧客体験のことで、商品の認知からアフターサポートまでの一連の中で得られる総合的な体験のこと

| プロジェクト体制の構築

AI開発では様々なステークホルダーを巻き込むことを意識する
→多くの人が協力してくれる体制を作り込んでいく
→実際にAIを使用するステークホルダーのニーズを汲み取り
利用してもらえるAIを開発していく

- ステークホルダー（利害関係者）
顧客、取引先、従業員、経営者、株主など

| プロジェクト体制の構築

- AIは開発したら終わりではなく、利用してもらうことが大切
→新しいシステムなどを導入してもなかなか使用してくれない
- 開発したシステムを利用してもらうためには、
インセンティブ（報酬）設計も大切になる
- システムを利用したくなる動機づけなどを行っていく

| プロジェクト体制の構築

新しい技術は難しそうという思い込みから
AIサービスを利用してもらえないという可能性がある
→誰でも使えるサービスを提供することが求められる

→同じ機能でも、デザインによって定着度は異なる
使いやすいデザイン、親しみやすいデザインのサービスは
定着度が高く、利用頻度も高い傾向がある

開発手法



| 開発手法

・開発手法

ソフトウェアの開発手法には、**アジャイル開発**や
ウォーターフォール開発などがある

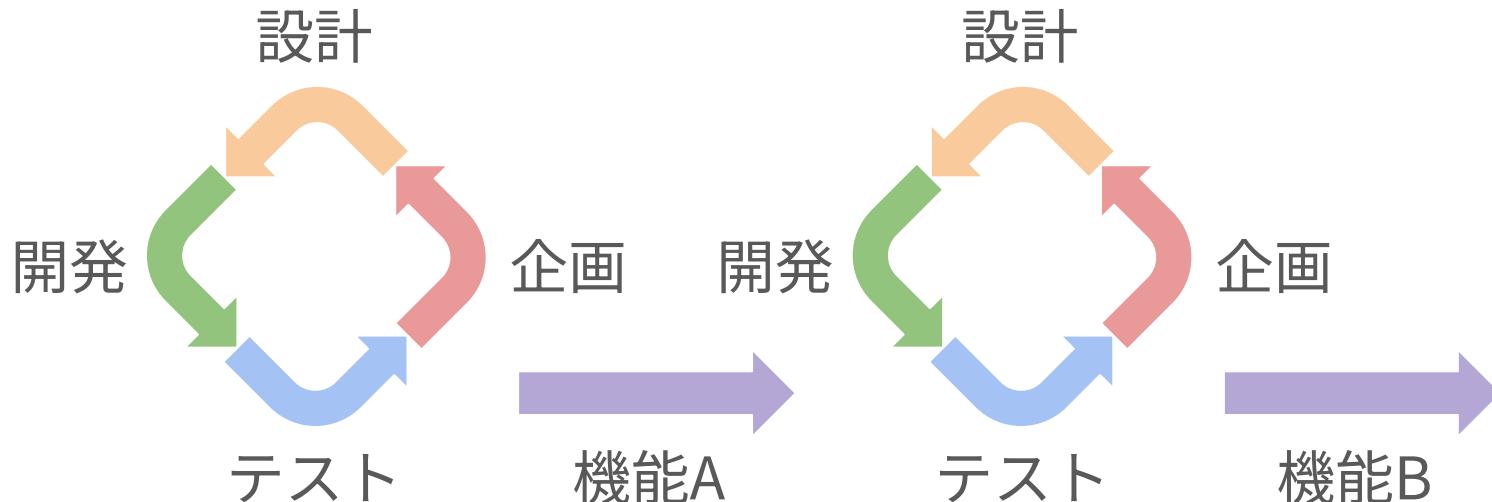
→目的に合わせて開発手法を選択する

AI開発では、速い技術の変化に対応するため、
アジャイル開発などの柔軟な開発手法が採用されること多い

開発手法

- ・アジャイル開発

短い開発サイクルを繰り返しながら、開発を行っていく手法



開発手法

- ・アジャイル開発の特徴

数週間から数ヶ月でユーザーの声などをもとに機能を開発していくため、リリースまでの期間が短い

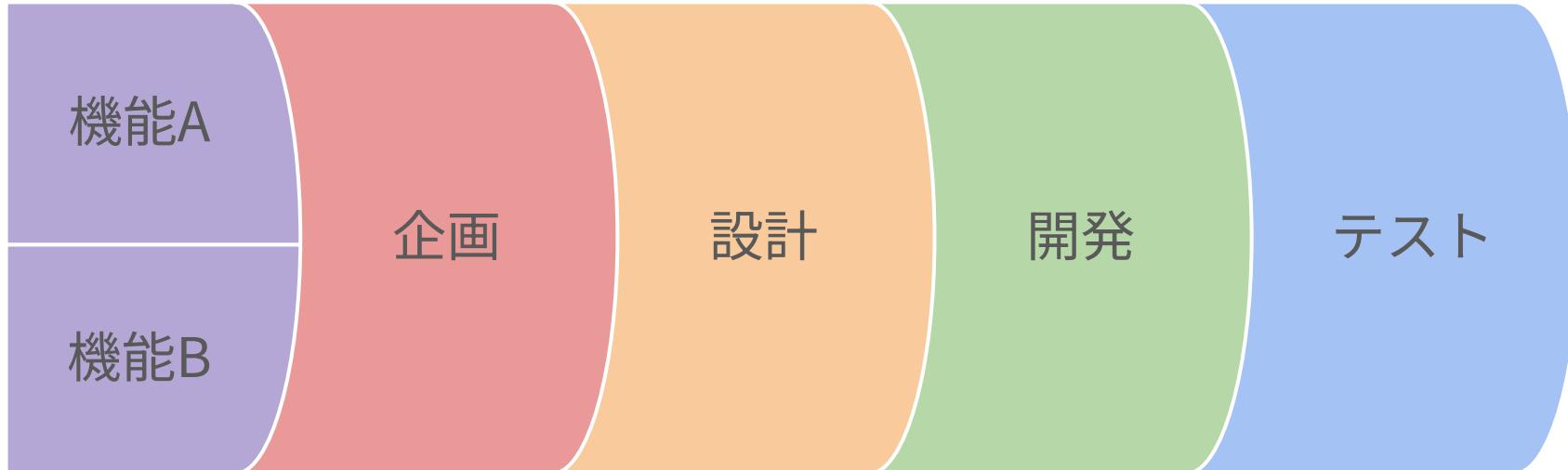
完成図が見えづらく、全体のスケジュールを予測することが困難

複数の機能の開発・改善が行われるのでマネジメントが難しい

開発手法

・ウォーターフォール開発

水の流れのように上流から下流へ工程を順に完成させていく手法



| 開発手法

・ウォーターフォール開発の特徴

リリースまで数ヶ月から数年であり開発期間が長い

全体像を固めてから開発をしていくため、マネジメントしやすい

丁寧にドキュメントを作成することが多いため、

ドキュメント作成に時間はかかるが、引き継ぎなどがしやすい

開発手法

・特徴

素早く開発し、リリースしたい場合や
ユーザーの声をもとに素早く改善していきたい場合は
アジャイル開発を選択することが多い

機能などを追加・改善することが少ない場合は
ウォーターフォール開発を選択することが多い

データ収集



| データ収集

AIを開発するためには質の高いデータが大量に必要

- **大量のデータを収集する**

データが網羅的でないと偏った学習になってしまう

- **質の高いデータを収集する**

ノイズ等が入っているデータなどは使えない

| データ収集

- データの収集方法

自社だけでデータを収集することが難しい場合は
オープンデータセットなどを購入する

- オープンデータセット

企業などがネット上などで提供しているデータセットのこと

| データ収集

- オープンデータセット
 - 画像関係
ImageNet、MNIST、CIFAR など
 - 自然言語関係
WordNet、SQuAD など

| データ収集

- **ImageNet**

数千万枚以上ある大規模な画像データベース

→ラベルが設定されていますが、中には間違ったラベルも存在

- **MNIST**

0~9の手書き数字のモノクロ画像を集めたデータセット

→6万の訓練画像と1万の評価画像が含まれている

| データ収集

- CIFAR

飛行機や自転車など様々な種類の画像が含まれている
→数万種類の32×32のカラー画像が含まれている

→CIFAR-10やCIFAR-100などがあり、
10や100は画像データのクラス数を表している

| データ収集

- **WordNet**

英語の語彙データベースであり、語彙の類義語や
上位概念・下位概念などの情報がまとめられている

- **SQuAD**

文章と質問と回答がペアになったデータセット
→自然言語の読解のベンチマークでよく使用される

| データ収集

- ・ オープンデータセット

利用条件などによりオープンデータセットが使えない場合は
自分たちでデータを収集する必要がある

→コスト面を意識しながらどのようなデータを収集するのか考える

→カメラなどセンサを活用してデータを収集していく

サンサにはX線、温度計など様々な種類のものが存在する

| データ収集

- ・マルチスペクトラム画像

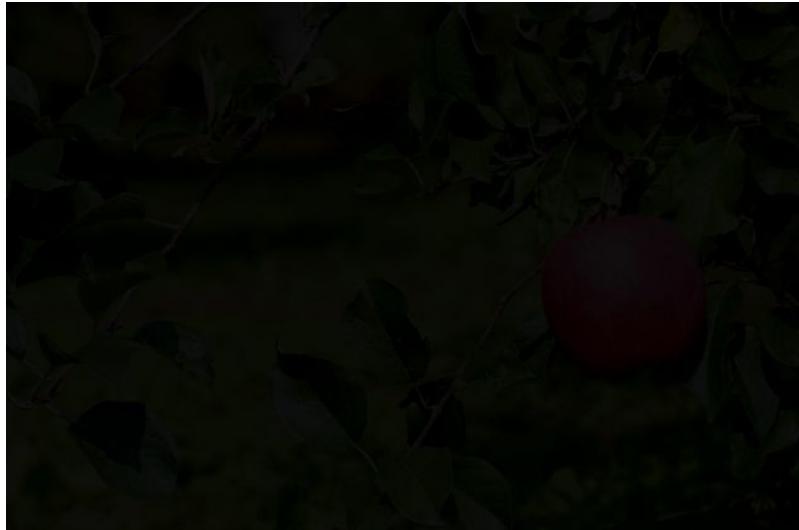
人が目で捉えることができる可視光線だけでなく、
紫外線、赤外線など複数の波長帯の電磁波を記録した画像のこと
→マルチスペクトラム画像を使用することでより詳細な解析可能

→暗い場合、可視光カメラでは物体が映らない時でも、
赤外線カメラなら物体を映すことができる

| データ収集

・マルチスペクトラム画像

可視光カメラ



赤外線カメラ



データの収集と外部連携



| データの収集と外部連携

・ 外部との協力

AIを開発する場合、様々な専門知識が必要になるため
外部と連携していくことが多い

- 企業と大学が連携してAIを開発したり、
企業・大学・行政が連携してAIを開発したりすることもある
- 企業・大学（研究機関）が連携することを**产学連携**、
企業・大学（研究機関）・行政が連携することを**产学官連携**

| データの収集と外部連携

・ 外部との協力

自社のノウハウやアイデアなどだけでなく、社外のノウハウやアイデアなどを組み合わせてAIを開発することも多い

→イノベーションを促進・達成するために組織内外のアイデア、ノウハウ、データなどを組み合わることをオープン・イノベーションという（AI分野でも期待されている）

| データの収集と外部連携

・ 注意点

- 多くの人がプロジェクトに関わることで、
責任や役割が分かりにくくなるデメリットも存在する
→責任や役割を明確にしていくことが大切になる
企業文化などによりトラブルが生じてしまうケースもある
- 企業間同士で話し合い、契約を交わすことが重要になる

データの加工



| データの加工

・データの加工

収集したデータをそのまま使うことが難しい場合は
データをAIが学習しやすいように加工する必要がある

- ・データクレンジング
- ・エンコーディング
- ・アノテーション

| データの加工

- データクレンジング

データの重複や誤記などの不備、欠損しているデータや無関係なデータを特定して修正、変更、削除などを行うこと

- エンコーディング

一定の規則に基づいて、目的に応じたデータに変換すること
→カテゴリ変数を直接処理のは難しいので、数値に置き換えて処理

| データの加工

- ・エンコーディングの代表的な手法
 - ・ラベルエンコーディング
 - ・カウントエンコーディング
 - ・ターゲットエンコーディング
 - ・ワンホットエンコーディング　などがある

| データの加工

- ラベルエンコーディング

各カテゴリに対して一意の数値を割り振る手法のこと

→パソコン、スマホ、タブレットというカテゴリがあったら、
重複しないように数値を割り振っていく

→パソコンは1、スマホは2、タブレットは3

というふうに数値を割り振っていく

| データの加工

・カウントエンコーディング

データにカテゴリが登場した回数を
カテゴリの数値として割り振ること

→A型の人が10人、B型の人が8人、O型の人が12人ならば
A型は10、B型は8、O型は15というふうに数値を割り振っていく

| データの加工

・ターゲットエンコーディング

カテゴリの目的変数の平均を、そのカテゴリの数値として割り振る手法のこと

→顧客グループAの平均購入単価が50,323円、
顧客グループBの平均購入単価が34,824円の場合、
それぞれの値が各グループの数値になる

| データの加工

- ・ワンホットエンコーディング

カテゴリ変数をワンホット表現に変換する手法のこと

→ワンホット表現とは、1つの値が1、他の値を0で表す表現方法

- ・赤色 => [1, 0 ,0]
- ・青色 => [0, 1 ,0]
- ・白色 => [0, 0, 1]

| データの加工

・アノテーション

テキストや画像などのデータに対して、
タグやメタデータなどを設定する作業のこと

→教師あり学習を行う場合、大量のデータに対して

1つ1つラベルを設定していく必要がある

→作業者によってばらつきがないようにマニュアルなどを作成



プライバシー保護に関する技術

| プライバシー保護に関する技術

- プライバシー保護に関する技術
 - 差分プライバシー
 - 連合学習
 - 暗号計算
 - k-匿名化

| プライバシー保護に関する技術

・差分プライバシー

プライバシーを保護するために、統計的に影響が出ないように、データにノイズを付加すること

- A、B、Cの平均点が50点、Aが40点、Cが60点の場合、Bの点数は50点だと推測することができてしまう
- 各点数にノイズを追加することで推論しにくくする

| プライバシー保護に関する技術

・差分プライバシー

Aの点数：40点 → 38点 (-2点)

Bの点数：50点 → 53点 (+3点)

Cの点数：60点 → 57点 (-3点)

→平均点は約49.3点になり、AとCの点数が分かっても
正しくBの点数を予測することはできない

| プライバシー保護に関する技術

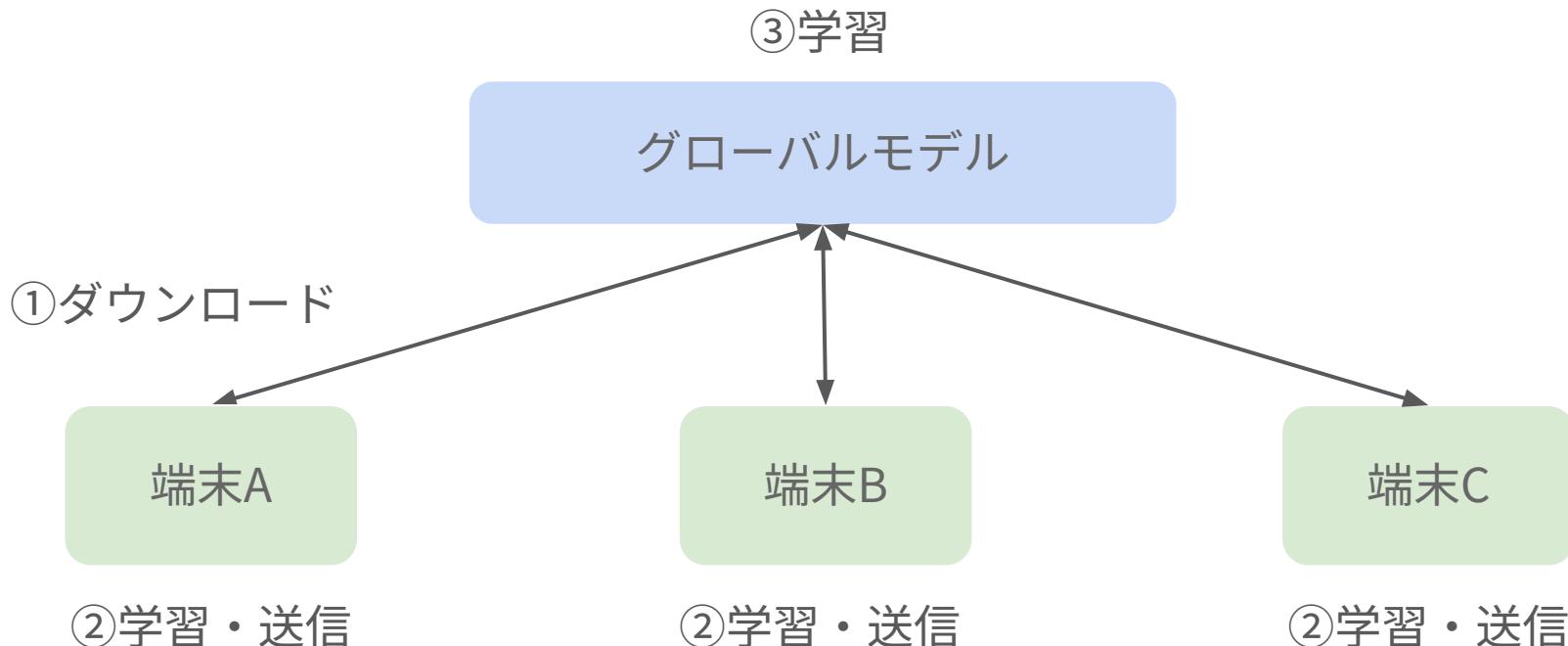
・連合学習

データを中央に集めずに、各端末（スマホなど）に分散した状態でモデルの学習を行う手法のこと

1. 各端末がモデル（グローバルモデル）をダウンロード
2. 各端末でデータを使用して学習を行い、差分を中央に送信
3. 差分を利用してグローバルモデルを更新を繰り返す

| プライバシー保護に関する技術

- ・連合学習



| プライバシー保護に関する技術

・暗号計算

データを暗号化したまま計算する技術のこと

→プライバシーの含まれたデータを複数事業間でやりとりすると
漏洩リスクが高くなるため、暗号化したデータでやりとりする



| プライバシー保護に関する技術

- k-匿名化

同じ属性（**準識別子**）を持つデータがk個以上存在するように、データを加工する方法で、プライバシーを保護する技術の1つ
→**準識別子**：年齢や郵便番号など個人の特定が困難な属性のこと

→情報が流出してしまったとしても個人の特定が困難になる
同じ属性を持つデータが100個あった場合、個人の特定は困難

| プライバシー保護に関する技術

- k-匿名化



開発環境



開発環境

・開発環境

AIを開発するときにPythonと呼ばれる

プログラミング言語がよく使用されている

→従来はプログラミング言語をダウンロードし、

パソコンで使用できるように環境を整える必要があった

→開発環境を構築することは初心者にとって難しい

開発環境

- 開発環境

Jupyter NotebookやGoogle Colaboratoryを使用することで
Pythonなどをダウンロードせずにコードを書くことができる
→何か思いついたらすぐにコードを書くことができる

→現在では一般的な開発ツールを1つにまとめた環境である
統合開発環境（IDE）を使って開発を行うことも増えてきた

開発環境

- **Jupyter Notebook**

Webブラウザ上でPythonなどを記述・実行できる開発環境
→ツールを利用することでデータ分析作業が楽になる

- **Google Colaboratory**

Google社が提供しているWebブラウザ上でPythonなどを
記述・実行できる開発環境

開発環境

- 開発環境

現在は、ディープラーニングの研究が進み、
数多くのライブラリやフレームワークが提供されている

- ライブラリ

様々な機能を簡単に使えるようにひとまとめにしたもの
→複雑な機能を短いコードで実装（実現）することができる

開発環境

- 開発環境

- フレームワーク

実装に必要となる一般的な機能や定型コードを
ひな形としてひとまとめにしたもの

→フレームワークを使用することで定型的なコードなどを
記述する必要がないため、開発コストを低くすることが可能

開発環境

- 代表的なライブラリ・フレームワーク

- **Numpy** : 配列や行列などの演算が行える
- **Pandas** : グラフ化、データ分析など行える
- **TensorFlow** : ディープラーニングのフレームワーク
- **PyTorch** : ディープラーニングのフレームワーク
- **Caffe** : ディープラーニングのフレームワーク
- **Chainer** : ディープラーニングのフレームワーク

開発環境

- **TensorFlow**

Googleが開発したディープラーニング用のフレームワーク
→TensorFlowには**Keras**というライブラリがある

- **Keras**

比較的短いコードで、簡単に深層学習を実装できる
→TensorFlowなどの上で動き、初心者でも使いやすい特徴

開発環境

- **PyTorch**

Facebookが開発したディープラーニングのフレームワーク

- **Caffe**

カリフォルニア大学バークレー校で開発された
ディープラーニングのフレームワーク

開発環境

- Chainer

日本のPreferred Networksが開発したディープラーニングのフレームワーク（開発は終了している）

→様々なフレームワークが無料で提供されている
バージョンも更新され、使いやすく、高機能になっている

開発環境

- ディープラーニングのフレームワークの種類

ディープラーニングのフレームワークは大きく分けて2種類ある

→ 「**define-and-run**」 と 「**define-by-run**」 である

→ディープラーニングのフレームワークの主流は、

「**define-and-run**」 から 「**define-by-run**」 に変わってきている

開発環境

- **define-and-run**

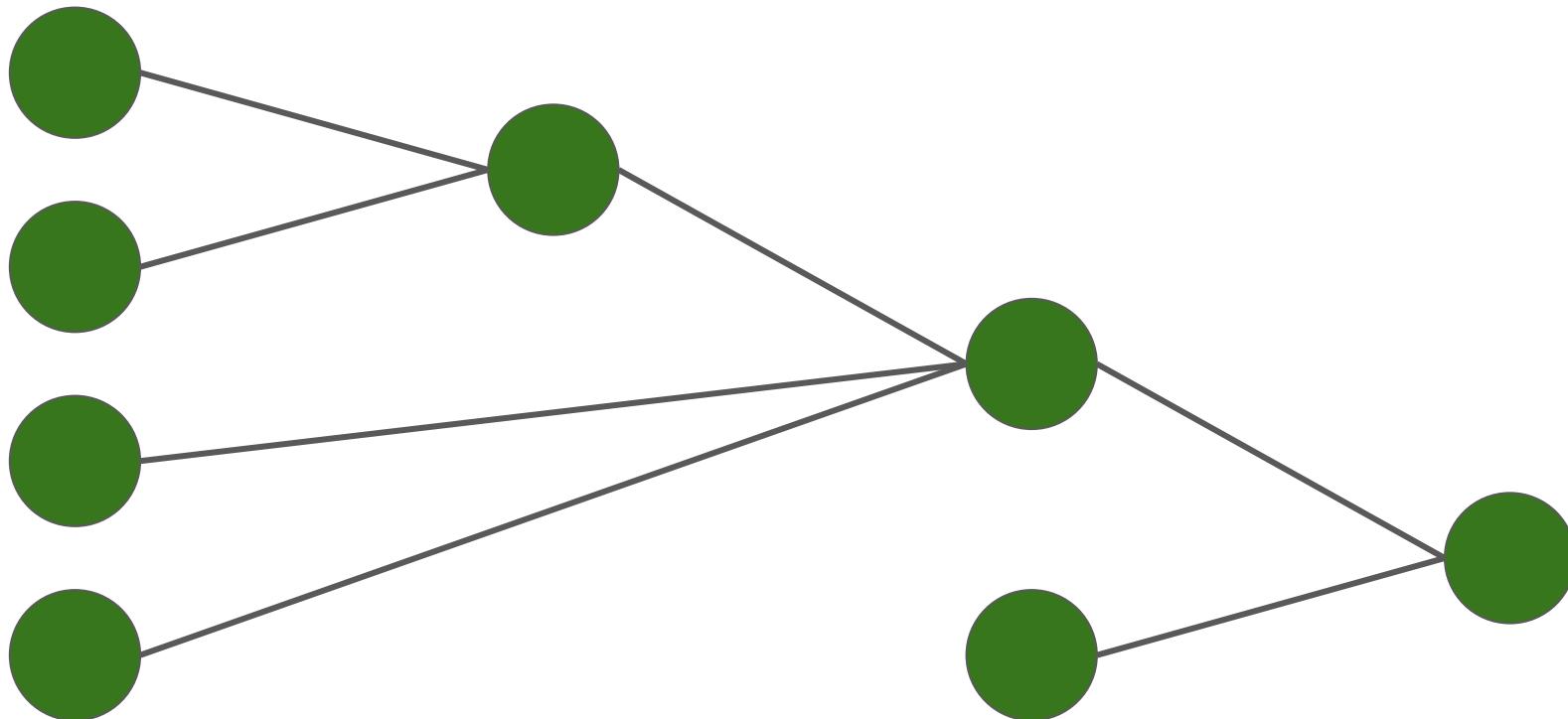
計算グラフの構築後に、データを流して処理を
していく方式のこと

→Tensorflow、Cafeeなどが代表的なフレームワーク

- **計算グラフ**

計算の過程をグラフ化したもの

開発環境



開発環境

- **define-by-run**

データを流して計算を行いながら、計算グラフを構築していく方法のこと

→Chainer、PyTorchが代表的なフレームワーク

開発環境

・開発環境

複数人で開発する場合は、開発環境を同じにする必要

→Pythonやライブラリなどのバージョンが異なると
プログラムが上手く動かないことがある

→複数のプロジェクトに参加している場合、

プロジェクトに合わせて開発環境を変更できることが理想

開発環境

- 開発環境

`pyenv`、`virtualenv`などを使用することで環境の切り替えが可能

- `pyenv` : Pythonのバージョンを管理するツール
- `virtualenv` : 1つのシステム内に異なる環境を構築することができるツール

→開発環境を開発者同士で共有することも可能である

開発環境

・開発環境

- Dockerを使うことでOSレベルから開発環境を構築できる
→OSなども統一させたいときに使用する
- 1つのパソコン内に仮想的に構築した環境を**仮想環境**という
仮想環境は互いに独立しており、影響を与えない特徴
- 基本的には**仮想環境**を構築し、その環境上で**開発環境**を整える

開発環境

- 開発環境

パソコン

仮想環境

仮想環境

仮想環境

仮想環境

仮想環境

仮想環境



Webサービス

| Webサービス

- Webサービス

技術者を支援するサービスは数多く存在する

→無料で利用できるサービスもあり、学習しやすい環境

→海外のサービスは大規模で利用しやすいものが多い

| Webサービス

- Webサービス

- GitHub

- ソースコードを保存・共有ができるサービス

- バージョン管理、プロジェクト管理なども行える

- arXiv

- 論文の公開や閲覧ができるサービス

| Webサービス

- Webサービス

- Google Scholar

- 論文の検索することができるサービス

- OpenAI Gym

- 強化学習のシミュレーション用プラットフォーム

| Webサービス

- Webサービス

- Kaggle

- データサイエンティストやエンジニアなどが

- モデルや分析手法の精度などを競い合うプラットフォーム

- Coursera

- 世界中の大学の授業を受講できるサービス

| Webサービス

- Webサービス
 - MOOC (MOOCs)
大規模なオンライン講義が受けられるサービス
 - Stack Overflow
プログラミング技術に関するナレッジコミュニティ



アセスメント コンセプト検証

| アセスメントとコンセプト検証

- ・アセスメント（評価）

データの収集とモデルについて検討を行っていく

→開発当初に予定していたデータ収集方法を改めて考えていく

予定していた方法でデータを収集することができない

こともよくあるので、どのようにすればデータを

収集することができるか考えていく必要がある

| アセスメントとコンセプト検証

・アセスメント（評価）

データの収集だけでなく、AIモデルについて検討していく
→予定通りに運用できるのか、開発コストはいくらなのか、
タスクをこなすために必要な時間はどれくらいかなども考える

→開発コストが高すぎる場合、タスクの処理に時間が
かかりすぎる場合は、他のAIモデルを導入することを考える

| アセスメントとコンセプト検証

- コストがかかる理由

3Dセンサなどを利用して質の高いデータを収集したり、
大量のデータを加工したりするのは想像以上のコストがかかる
→異なる方法で収集した複数データを一貫したデータに加工する
画像・動画ならば対象となる物体を中心に切り抜く など
→必要なデータがない場合は再びデータを収集する

| アセスメントとコンセプト検証

・アセスメント（評価）

多面的にAIプロジェクトを評価していき、

AIプロジェクトを実行する方が良いのか決断をする

→評価の結果によっては開発しないという選択肢を選ぶこともある

→AIは加速度的に発展しており、同じAIを開発するにしても、

現在開発するのか、1年後に開発するのかでコストは大きく変動

| アセスメントとコンセプト検証

- AIプロジェクトとROI（投資利益率）

時間や資金をかけて高度なAIモデルができる場合でも
投資額に対する利益率（ROI）が低いとき、
または投資額以上の資金を回収することできないなどは
AIプロジェクトを中止することが多い

→投資家や銀行などの利害関係者に説明できないため

| アセスメントとコンセプト検証

- ・コンセプト検証 (PoC : Proof of Concept)

開発することを決めたら、効果を測定したり、

技術的に可能なのかなどを判断するためにコンセプトを検証

→AIモデルを実際に作成し、実務でも使用できる

精度になるまで繰り返し改善をしていく

→当初予想していなかった問題などが発見されることもある

| アセスメントとコンセプト検証

- コンセプト検証 (PoC : Proof of Concept)

コンセプト検証の段階では、モデル 자체を変更することも多い
→失敗しながら成功に近づいていくイメージである

→実際にモデルを開発し、精度やコストを確認していく中で
実務で使用することが難しい場合はプロジェクトを終了する



モデルの学習・運用

作成者：辻 大貴

| モデルの学習・運用

・モデルの学習

検証が終わり、プロジェクトとして価値があることが分かれば
実際にモデルを学習させていく

→多種多様なデータを使用してモデルの予測精度を高めていく
テスト的に一部ユーザー向けにサービスを提供したりして、
モデルの実際の精度を測定していくことも大切

| モデルの学習・運用

・運用と再学習

- 実際にモデルを運用することで分かることが多い
→思ったよりもコストがかかる、処理に時間がかかる、
予測精度が思っていたよりも低い などなど
- 予想と現実には大きなギャップが存在する
予測精度が低い場合は追加でデータを学習させたりする

自動運転



| 自動運転

- **自動運転のレベル (SAE J3016)**

自動運転技術は米国自動車技術者協会 (SAE) によって定義
→レベル0からレベル5の6つのレベルに分けられている
各レベルの名称、運転主体の違いを理解することが大切

- **レベル0 (運転自動化なし)**

人間が全ての運転タスクを担当する

| 自動運転

- レベル1（運転支援）

システムが加速・減速制御、ハンドル操作の一方を実施
→運転主体は人間である

- レベル2（部分運転自動）

システムが加速・減速制御、ハンドル操作とともにを行う
→運転主体は人間である

| 自動運転

- レベル3（条件付き運転自動化）

限定した領域でシステムが運転タスクを実施するが、
人間が制御する必要もある

→運転主体は基本的にはシステム、人間が制御している場合は人間

→システムからの警告や要求に迅速に応答する必要がある

| 自動運転

- レベル4（高度運転自動化）

限定した領域でシステムが全ての運転タスクを実施

- レベル5（完全運転自動化）

領域を限定せずシステムが全ての運転タスクを実施

自動運転

・自動運転と法律

2021年、日本でレベル3の自動運転車が販売された

→渋滞していないときの高速など

特定条件下でシステム主導による運転が可能

→自動運転中の運転手はスマートフォンを操作することが可能

→道路交通法、道路運送車両法が改正されつつある

自動運転

- 自動運転と法律

2019年からながら運転への罰則が強化された

→作動状態記録装置の整備と記録保存が義務付けられた

- 作動状態記録装置

自動運行装置の作動状態を確認するために

必要な情報を記録する装置のこと