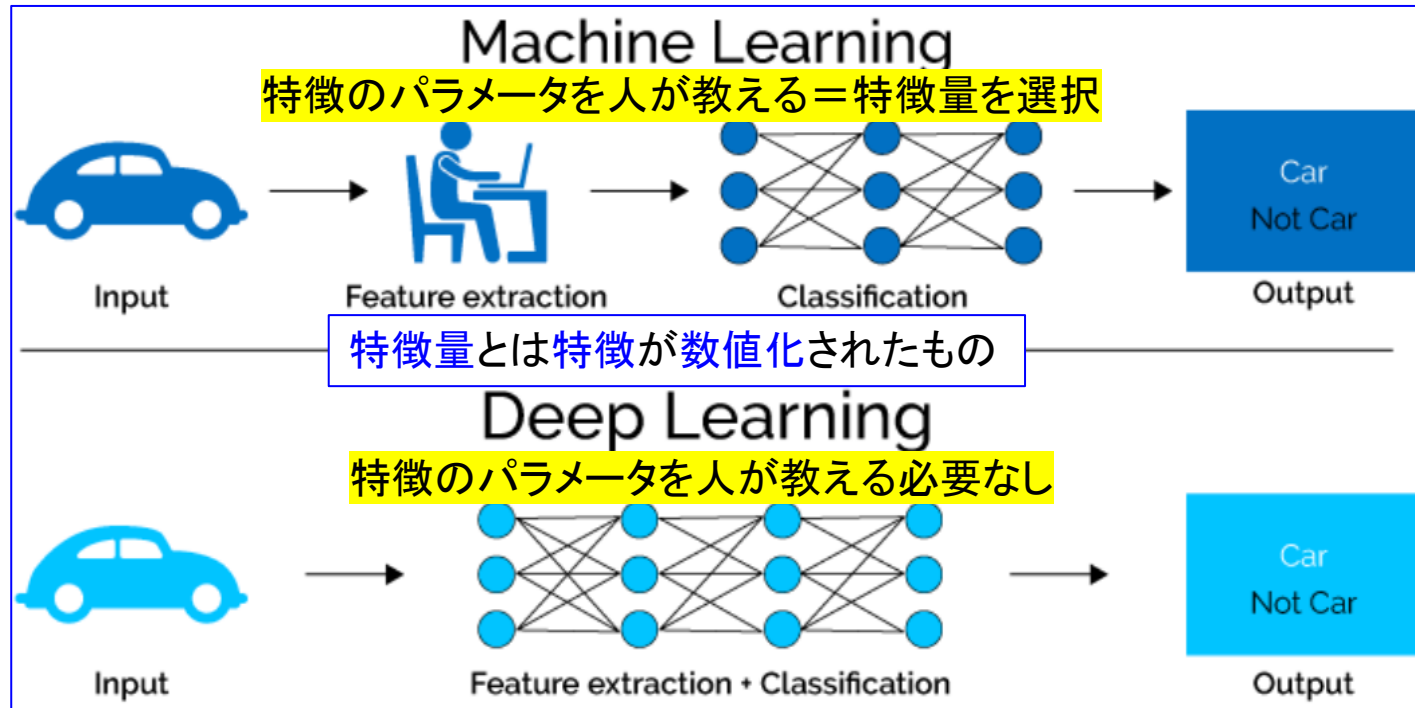


機械学習と深層学習の違い



機械学習は、入力された大量データから機械が自ら法則性や関連性を**学習**し、アルゴリズムに基づいて未知のものを**予測**、**判断**する技術

例: 気象データをもとに収穫量や収穫時期を予測、タクシーの需要供給予測、株価予測、

深層学習は、コンピュータが**自動的に**データから**特徴・パターン**を発見する技術

例: 画像認識、動画認識

活用事例：深層学習を用いてきゅうりの等級を判別



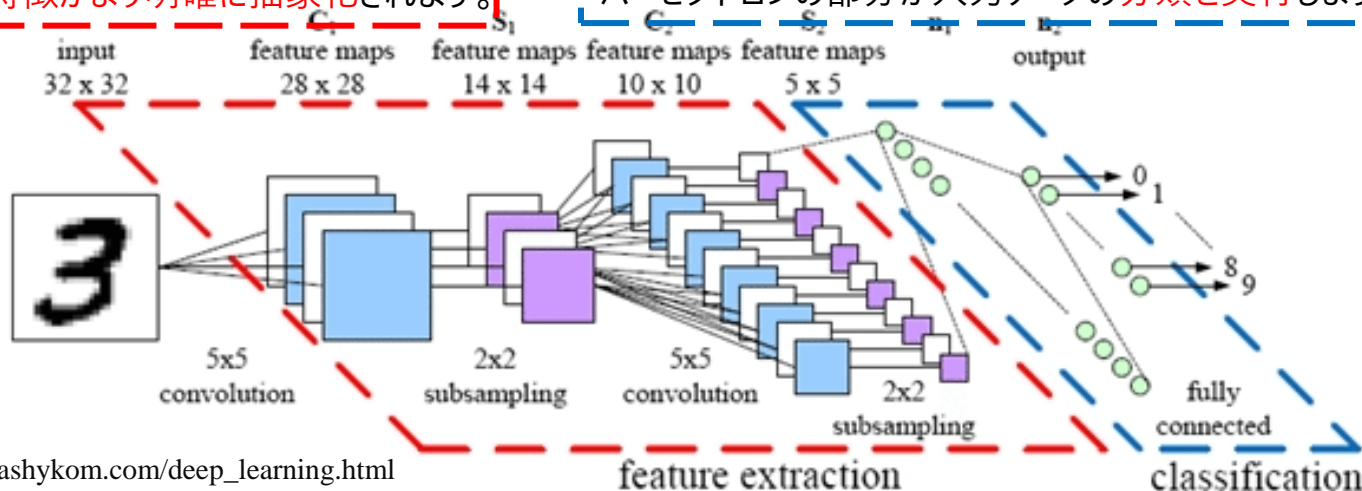
引用：https://interface.cqpub.co.jp/qly1_01/

多層ニューラルネットワーク

畳込み層とプーリング層を重ねることによって入力データの各特徴がより明確に抽象化されます。

CNNで有名なLeNetの例

こうして抽象化された各特徴を手がかりとして、最後の多層パーセプトロンの部分が入力データの分類を実行します。



引用: https://www.koi.mashykom.com/deep_learning.html

5×5 convolutionという表示は、5行5列のフィルター行列（カーネル）で畳み込むことを意味している。

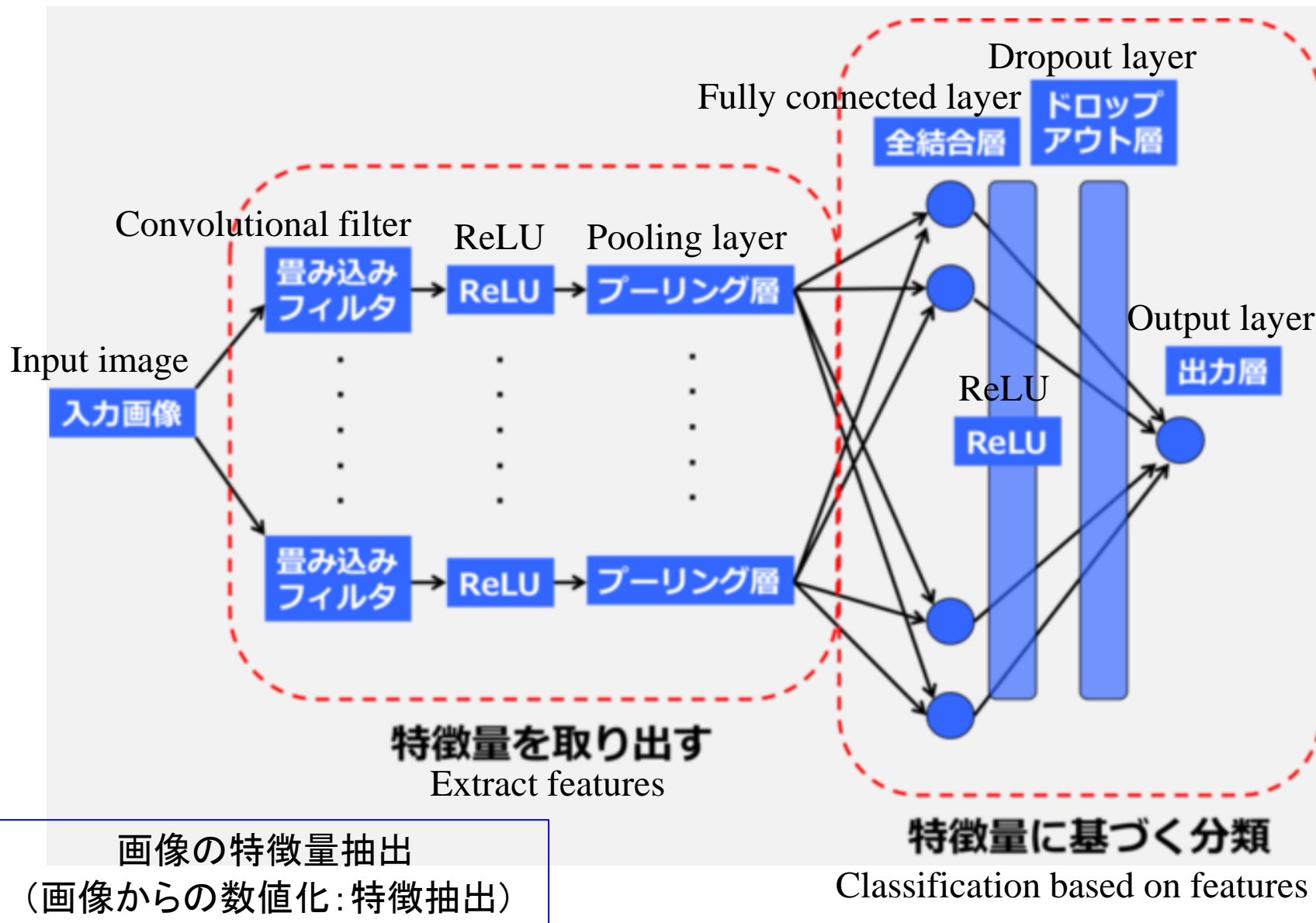
2×2 subsampling（pooling）という表示は2×2行列でデータサイズを縮小することを意味している。

深層学習(ディープラーニング)とは、人間の脳神経系のニューロンを数理モデル化したニューラルネットワークの層が幾重にも重なる、3層以上の多段構造を持つ機構を用いた学習

ディープラーニングはデータの中に存在しているパターンやルールの発見、特徴量の設定、学習なども機械が自動的に行うことが特徴である。

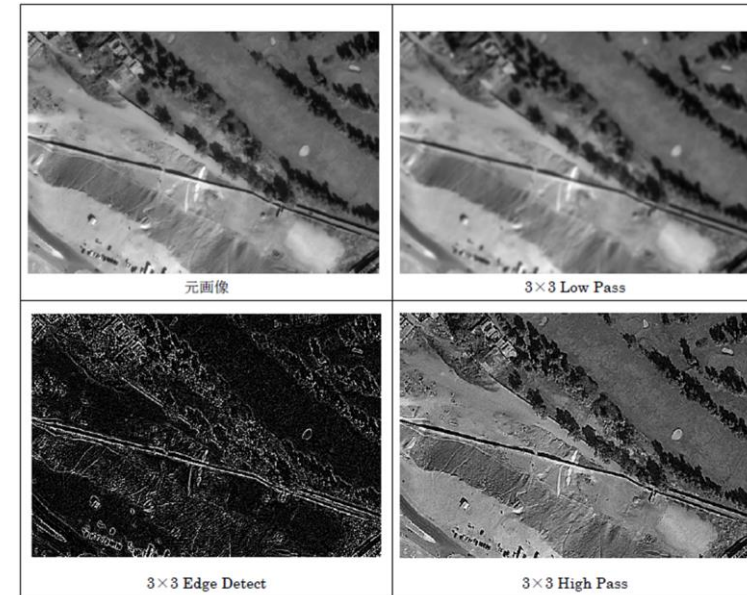
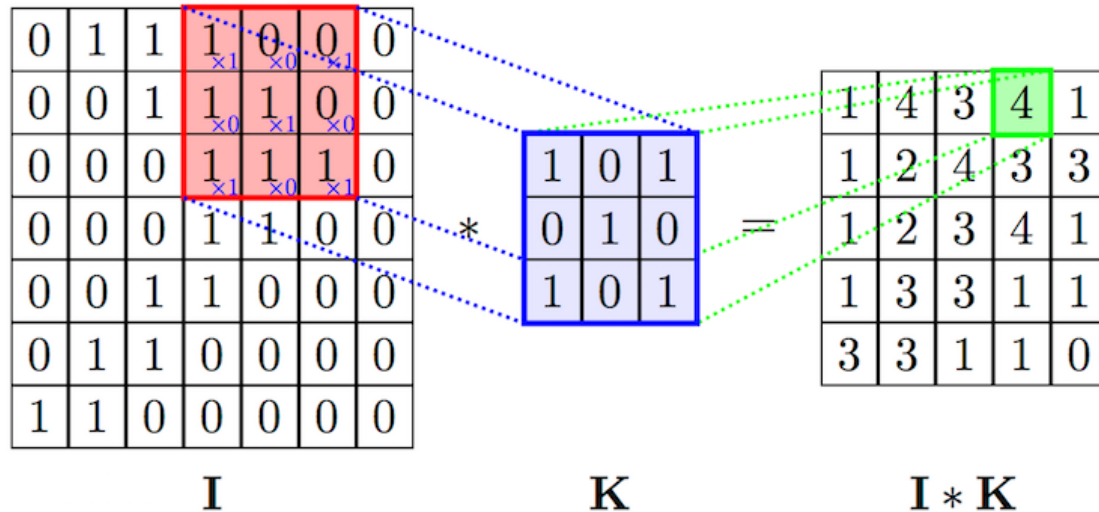
第1層の畳み込み層で入力データ(32x32サイズ)を5x5のフィルター行列で畳み込みを行い、28x28サイズの出力特徴マップとして出力。この特徴マップを2x2サイズのフィルター行列でサブサンプリング（プーリング）を行い、次の畳み込み層に14x14サイズで出力。この14x14サイズの特徴マップを5x5サイズのフィルター行列で畳み込みを行い、10x10の特徴マップとして、プーリング層に出力。2x2フィルター行列を持つプーリング層は、5x5サイズの特徴マップを出力して、全結合層(多層パーセプトロン)に入力信号として渡される。全結合層の出力は、活性化関数を通して、10クラスのカテゴリに分類される。

畳み込みニューラルネットワーク



畳み込み

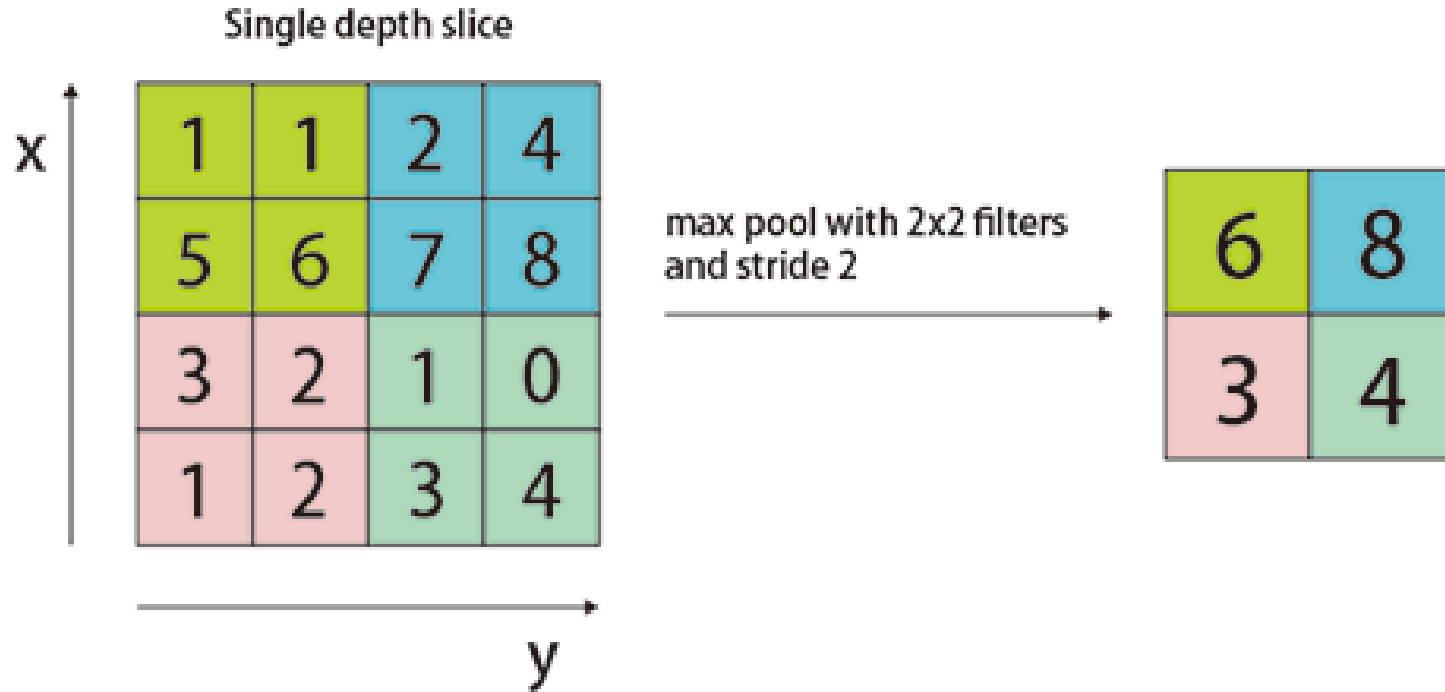
畳み込みのカーネル(フィルター)行列 K , 3×3 行列が下に描かれています。



カーネルの枠を一定間隔で入力データ I の行列枠上をスライドさせます。各スライドごとに、カーネル行列の要素と入力データの対応する要素をそれぞれ乗算し、その総和をもとめて、この総和値を特徴マップ (feature maps) に移します。

上の例では $1 \times 1 + 0 \times 0 + 1 \times 0 + 0 \times 1 + 1 \times 1 + 0 \times 0 + 1 \times 1 + 0 \times 1 + 1 \times 1 = 4$ と計算されています。入力データが 7×7 行列であるときには、特徴マップのサイズは 5×5 行列となります。

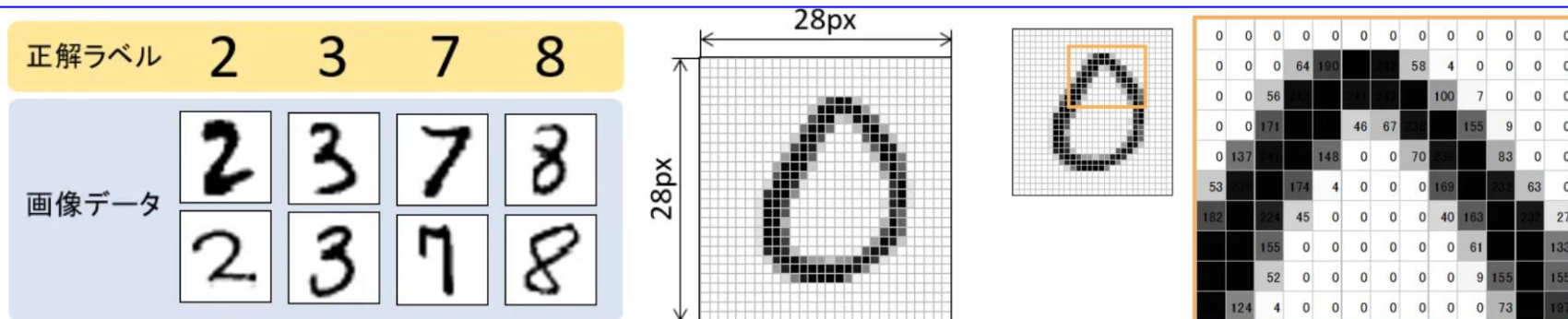
プーリング



max poolingとはフィルター行列で覆われた行列中の最大値をその行列の代表値にすることです。2x2フィルターによるmax poolingでは、入力データを2×2行列ごとに覆い、2×2行列の中の最大値を取り、その値を出力信号として送ります。この結果データサイズが縮小されます。

MNISTデータセット

MNISTとは「0～9」のいずれかの数字が書かれた手書き数字画像のデータセットです。



画像のサイズは28×28(px)で、各画像データには”どの数字が書かれているのか”を表す正解ラベルが付いています。

ラベルデータ

| offset | type | value | description |
|--------|----------------|------------------|-------------|
| 0000 | 32 bit integer | 0x00000801(2049) | 識別子(定数) |
| 0004 | 32 bit integer | 60000 or 10000 | ラベルデータの数 |
| 0008 | unsigned byte | 0 ~ 9 | 1つ目のデータのラベル |
| 0009 | unsigned byte | 0 ~ 9 | 2つ目のデータのラベル |
| | | | |
| xxxx | unsigned byte | 0 ~ 9 | 最後のデータのラベル |

画像データ

| offset | type | value | description |
|--------|----------------|------------------|------------------|
| 0000 | 32 bit integer | 0x00000803(2051) | 識別子(定数) |
| 0004 | 32 bit integer | 60000 | 画像データの数 |
| 0008 | 32 bit integer | 28 | 1画像あたりのデータ行数 |
| 0012 | 32 bit integer | 28 | 1画像あたりのデータ列数 |
| 0016 | unsigned byte | 0 ~ 255 | 1つめの画像の1ピクセル目の値 |
| 0017 | unsigned byte | 0 ~ 255 | 1つめの画像の2ピクセル目の値 |
| | | | |
| xxxx | unsigned byte | 0 ~ 255 | 最後の画像の784ピクセル目の値 |

1行目: 正解ラベル

ピクセル0からピクセル783まで(=784)ピクセル値

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q |
|----|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1 | label | pixel0 | pixel1 | pixel2 | pixel3 | pixel4 | pixel5 | pixel6 | pixel7 | pixel8 | pixel9 | pixel1 | pixel1 | pixel1 | pixel1 | pixel1 | pixel1 |
| 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

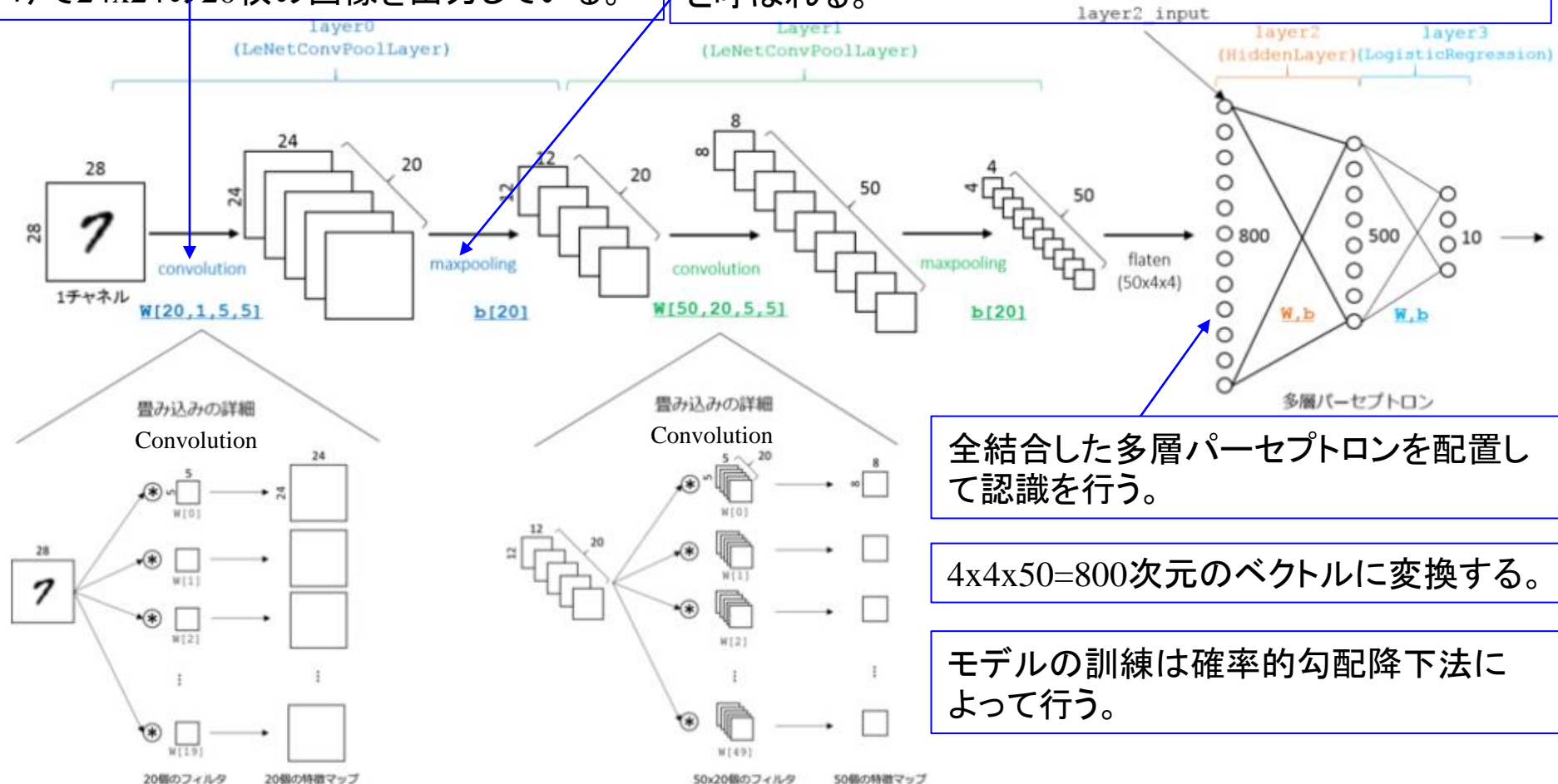
正解ラベルは正解がそのままの数字で入っているがこれをone-hot-encodingに変換
例:「3」→[0, 0, 0, 1, 0, 0, 0, 0, 0]

畳み込みニューラルネットワーク

通常のニューラルネットワークとの違いは、**畳み込み層**と**プーリング層**の部分にある。

28x28ピクセルの「7」という入力画像に対して5x5のそれぞれ異なるフィルタを20個かけて24x24の20枚の画像を出力している。

矩形内の最大の値を取り出して新しい画像を出力するような処理を行う。最大値に置き換える方法は**max pooling**と呼ばれる。



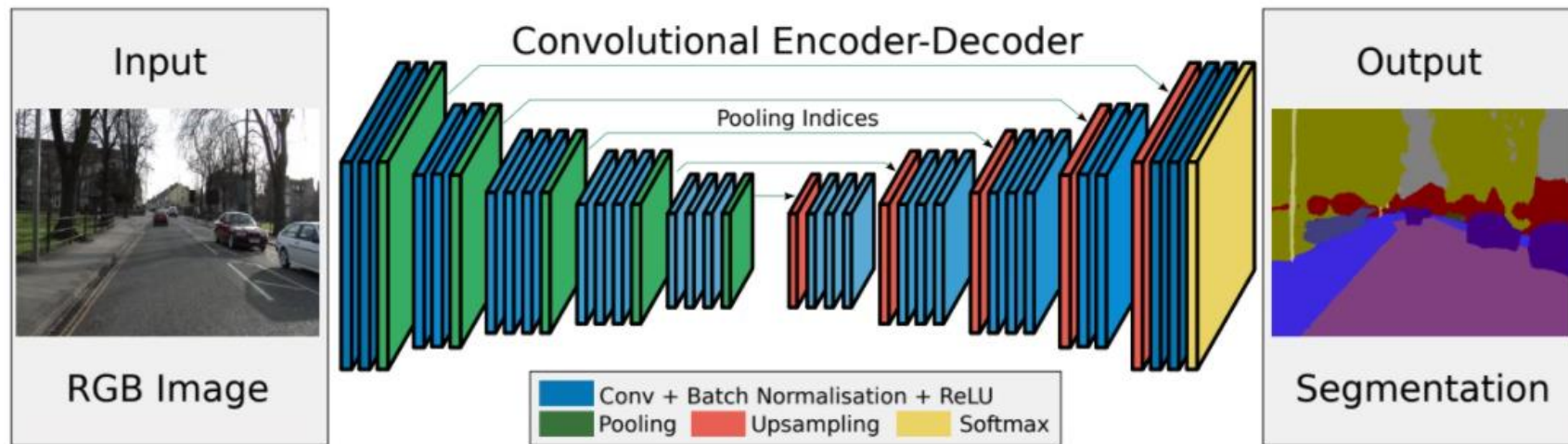
全結合した多層パーセプトロンを配置して認識を行う。

$4 \times 4 \times 50 = 800$ 次元のベクトルに変換する。

モデルの訓練は確率的勾配降下法によって行う。

SegNet: 画像セグメンテーションニューラルネットワーク

SegNet: Image Segmentation Neural Network



SegNetのアーキテクチャには**Encoderネットワーク**と**Decoderネットワーク**が構成されており、EncoderネットワークはVGG16モデルの一部を用いて、物体の特徴情報を抽出する。それに対してDecoderネットワークはこれまで得られた特徴情報を元に車や道路などの物体ごとにラベリングを行います。








RGB画像が与えられると、画像に存在する車や道路などにラベリング付けを行い、**番号の異なるラベルを違う色で表現するという仕組み**です。SSDと比べると物体の領域が正確になっているほか、自動運転に望まれる道路のフリー領域や道路標識も同時に予測することができます。

Encoder: 畳み込み層、プーリング層、BatchNormalization層から構成されています。畳み込み層では画像の局所特徴を抽出し、プーリング層でダウンサンプリングをし、更にスケール不変なものを後ろの層に伝播していきます。BatchNormalization層は、訓練データの分布を正規化することで勾配消失などを防ぎ、学習を加速させます。

Decoder: Encoderによって物体の種類と大まかな位置情報を持つ低解像度の特徴マップが得られたが、それを元の解像度マップに対応付けを行うのがDecoderです。具体的には低解像度マップをアップサンプリングして、畳み込み処理を行うことによって物体の形がよりきれいに補完されます。

深層学習により解決できる問題(タスク)の例

Source: <https://blog.albert2005.co.jp/2020/08/18/ssap/>

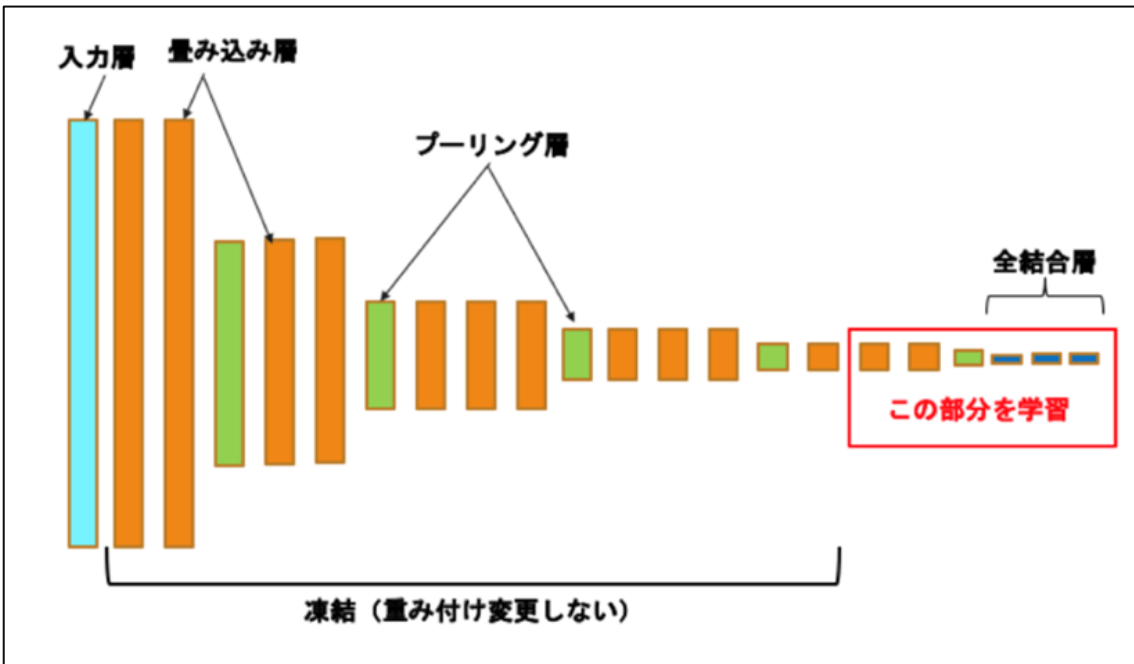
| タスク | 画像分類 | 物体検出 | Semantic Segmentation セグメンテーション | Instance Segmentation インスタンスセグメンテーション |
|-------------------------------|---|---|--|--|
| 入力画像 |  |  |  |  |
| 出力結果 | 車 |  |  |  |
| 与えられた画像が何であるか(クラス)を識別するタスクです。 | | 画像分類を行い、さらにその物体の場所も識別するタスクです。多くはBBBoxと呼ばれる矩形を用いて各物体の位置を特定します。 | BBBoxを使用せず、ピクセル単位で各クラスの領域を求めるタスクです。複数の物体がある場合でも同じクラスであれば区別なく同じクラスとして識別を行います。 | 各クラス領域をピクセル単位で識別し、さらに同じクラスでも別の物体(Instance)であればそれらを区別するタスクです。物体検出とSemantic Segmentationを合わせたタスクで、物体検出よりも精密に物体の形を捉える必要があります。 |

転移学習とは

学習済みのモデルを転用して、新たなモデルを生成する方法

- ・他の画像データを使って学習されたモデルを使うことによって、新たに作るモデルは少ないデータ・学習量でモデルを生成することが可能となる。

- ・例：VGG16モデルは1,000カテゴリ、100万枚以上の画像を認識する訓練がされており、過程において13層の畳み込み層と3層の全結合層の構成で、画像認識に適した重み付けが最適にチューニングされているので、少ないデータでも効率的に学習が行えるようになっている。

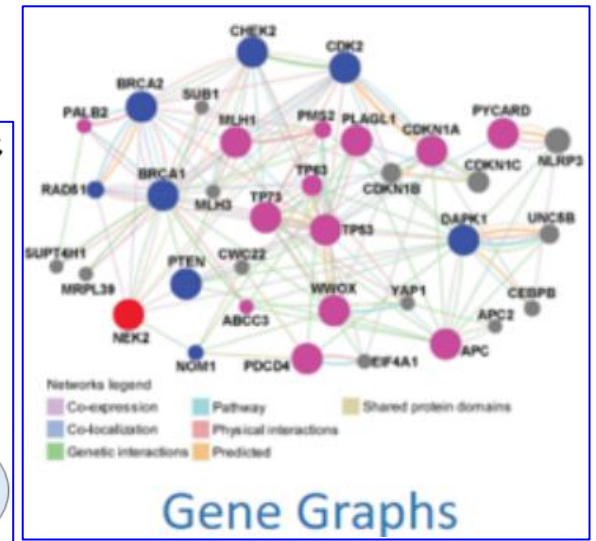
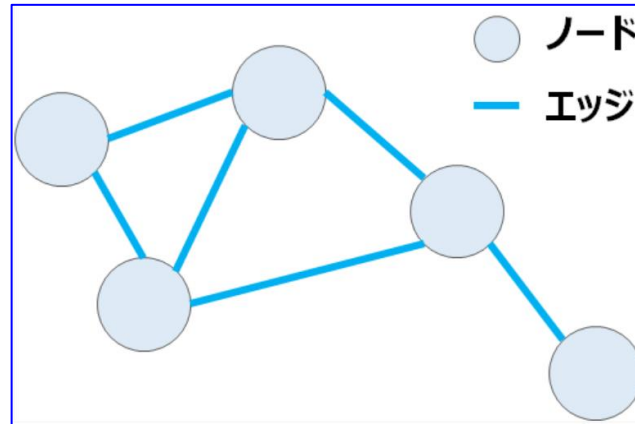


- ・16層目までを凍結(重み付けを変更しない)して、畳み込み層の最後の2層と全結合層で学習する。

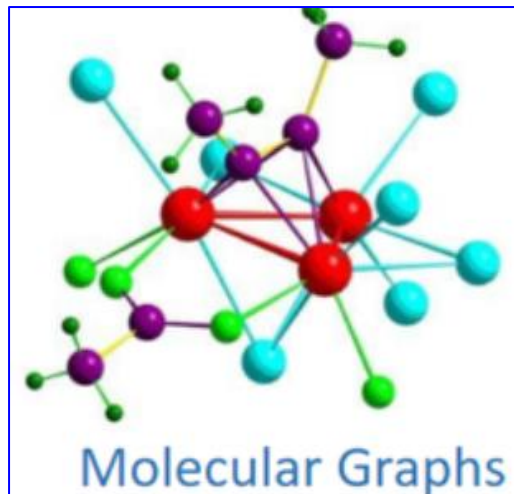
- ・凍結(フリーズ)していない部分を再生成して、その部分だけで新たに画像を追加学習する。

- ・赤枠部分を学習させることでVGG16モデルの高い特徴量抽出を残しつつ、少ないデータセットで精度の高い学習モデルを構築することが可能となる。

グラフと深層学習 (GNN・GCN) により解決できる問題 (タスク) の例



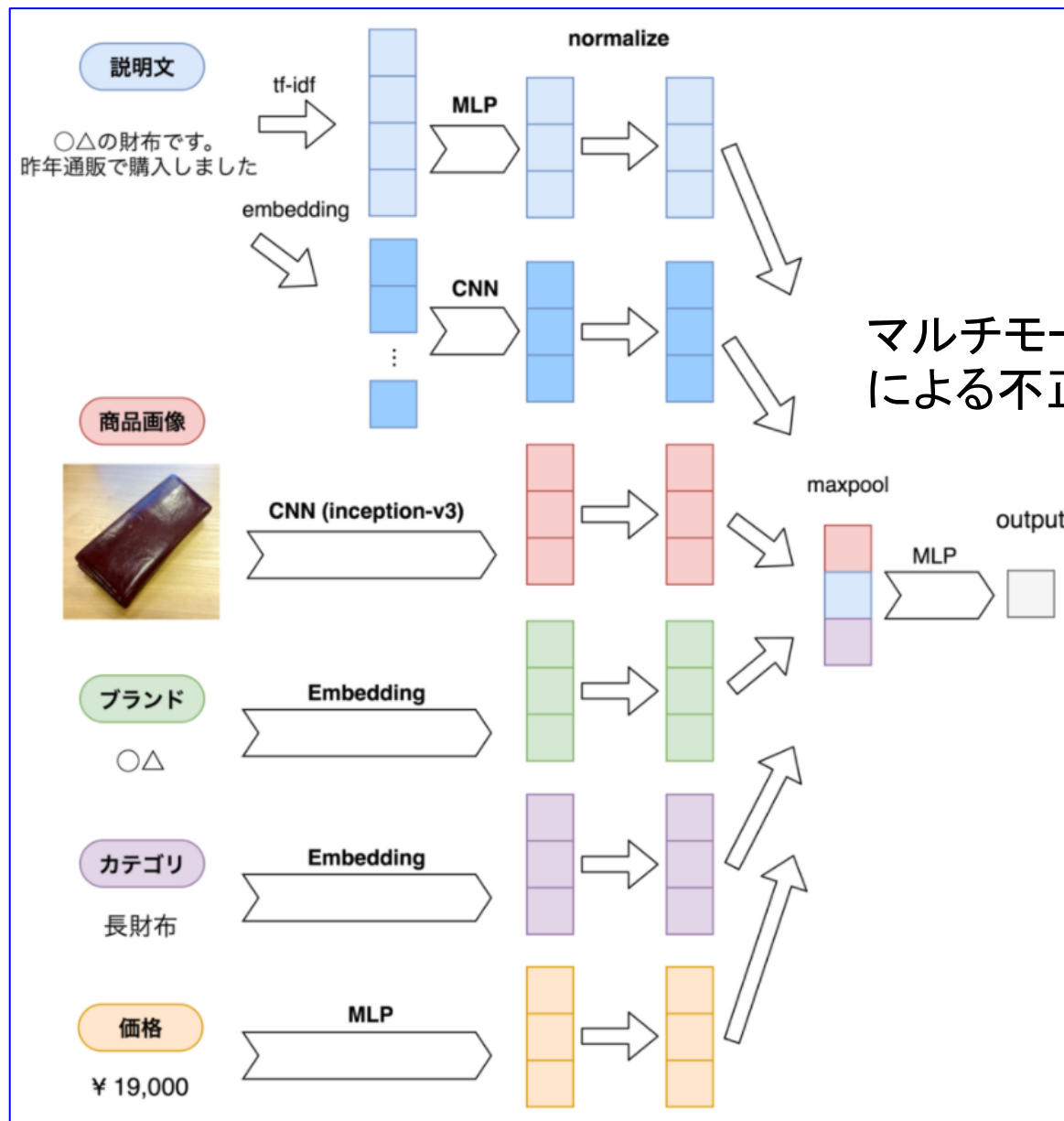
- ・コミュニティ: ノード→人、エッジ→友人関係
- ・交通: ノード→地点、エッジ→ルート
- ・化合物: ノード→原子、エッジ→結合



Source: <https://note.com/tak1/n/n8751959d885b>

マルチモーダル深層学習

映像、テキスト、音声などの複数のモダリティを統合的に処理、複数要因を考慮した予測が可能



マルチモーダルモデル
による不正出品の検知

各業種におけるビッグデータの活用事例

1. 来客予測⇒食品ロス削減

- ・開発環境: Microsoft Azure
- ・Cognitive Services: 画像、言語、音声、検索、知識の5分類29種類のAPIを提供
- ・『Face API』『Emotion API』: 店内カメラの映像⇒性別・年齢・新規/リピーター・感情を判定
- ・スクレイピング: 気象庁による気温や降水量などのデータ、近隣の宿泊客数、スマレジのPOSデータ、食べログや自社サイトのアクセス数などを収集

翌日の来客数 = 気象データ × 売上データ × 近隣の宿泊客数
⇒ 来客予測の精度向上に向けて
× 来客属性 (性別・年齢・新規/リピーター・感情)

Source: <https://k-rip.gr.jp/wp/wp-content/uploads/2020/11/eco-vol132-3.pdf>

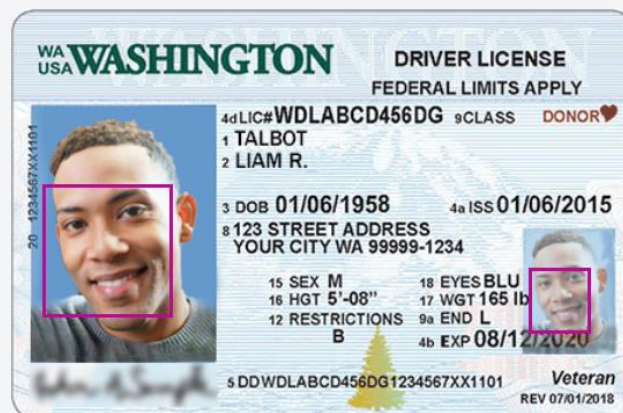
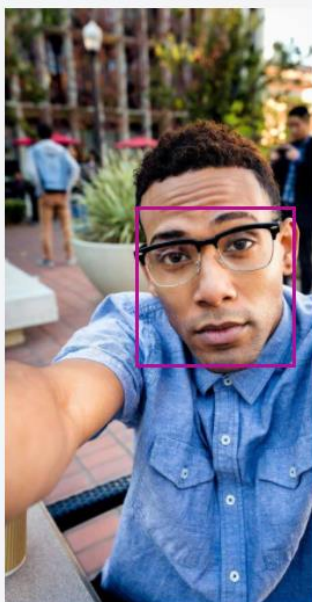
Source: https://japan.cnet.com/extra/ms_ebiya_201710/35108734/2/

『Face API』

画像内の顔を分析する AI サービス

顔検証

2 つの顔が同一人物のものである可能性を検証し、信頼度スコアを取得します。



画像の URL

送信

参照

画像の URL

送信

参照

検証結果: これら 2 つの顔は同一人物のもので、信頼度は 0.83046 です。

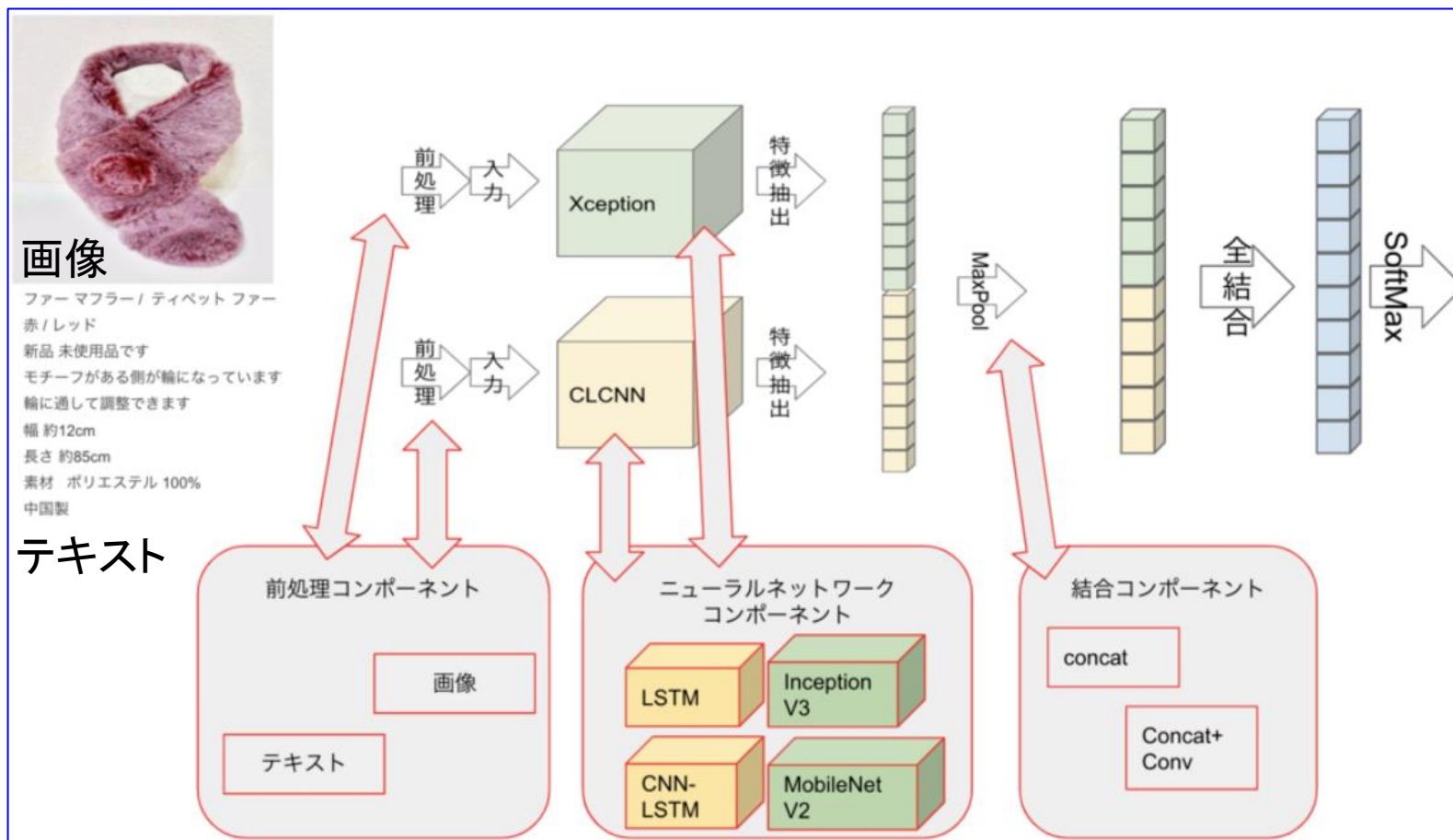
Source: <https://azure.microsoft.com/ja-jp/services/cognitive-services/face/#demo>

各業種におけるビッグデータの活用事例

2. 商品分類

出品内容を分類＝画像×テキスト

テキストと画像を組み合わせることによって、より正確に分類できる



Source: <https://engineering.mercari.com/blog/entry/2019-02-19-180000/>

各業種におけるビッグデータの活用事例

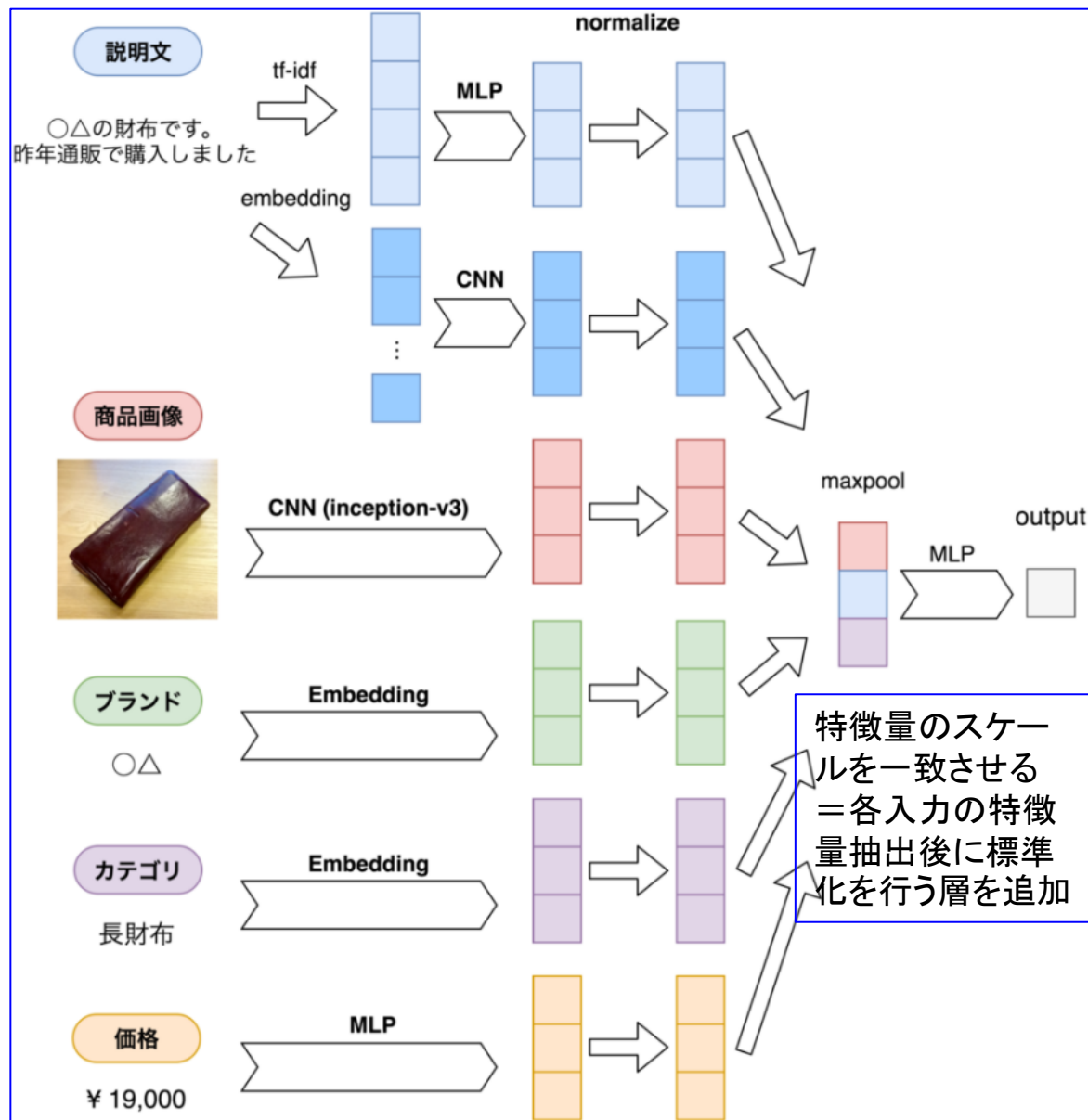
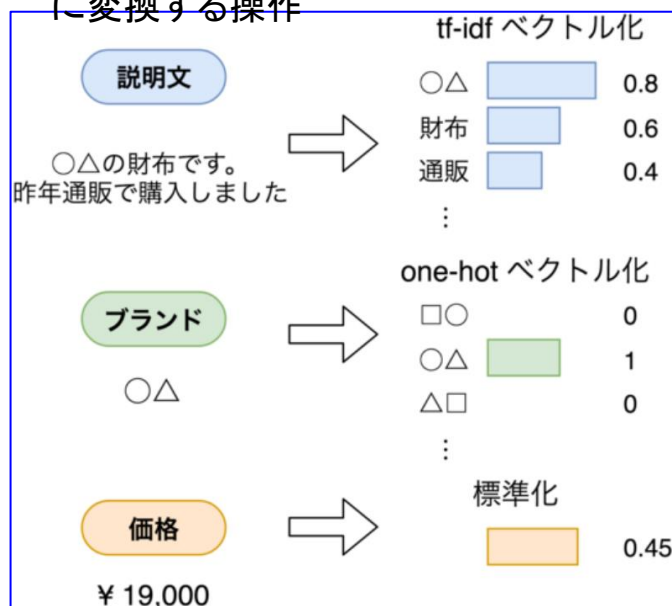
3. 不正出品の検知

不正出品の検知

= 画像 × テキスト × ブランド
ID × カテゴリ × 価格 × CNNによる
テキストの特徴量抽出

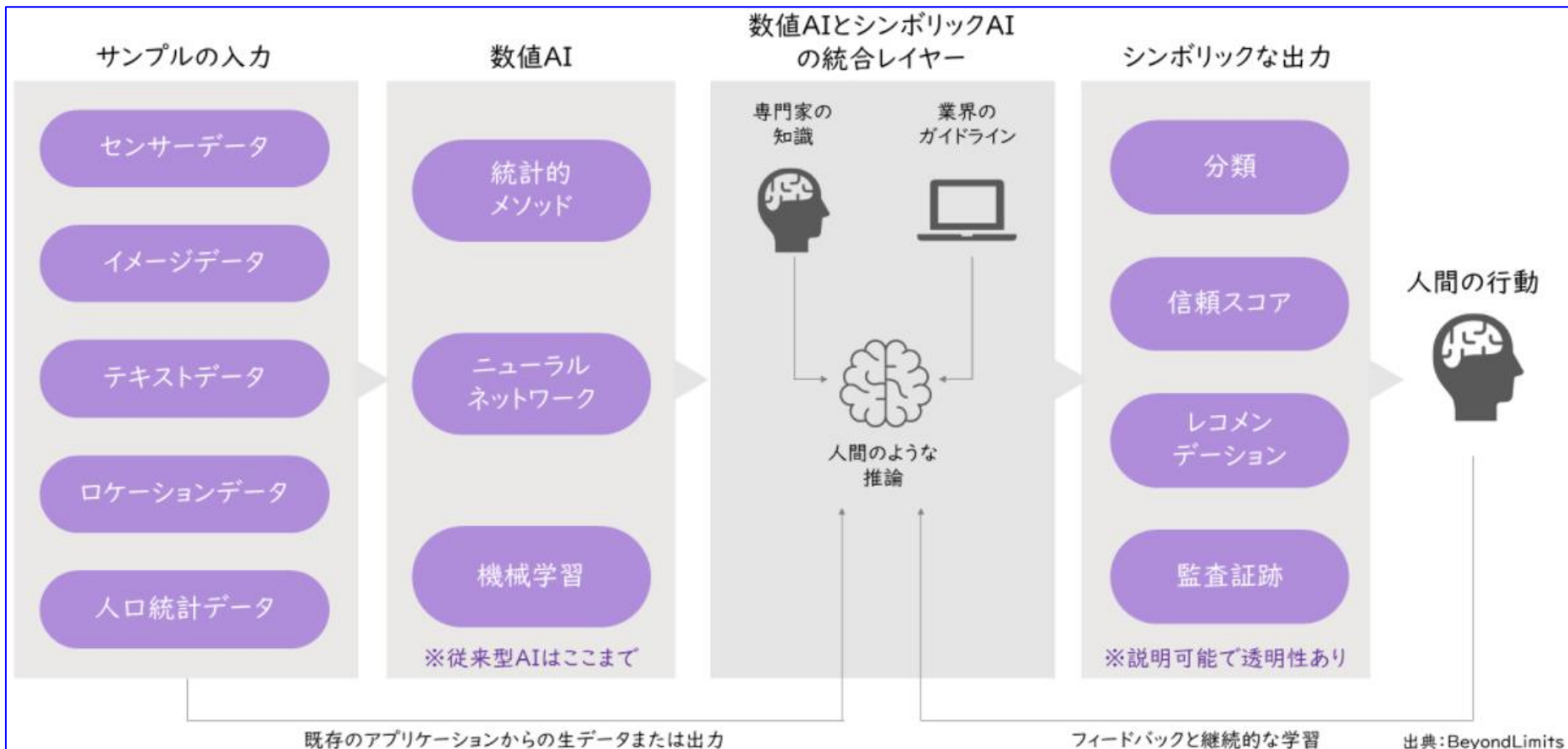
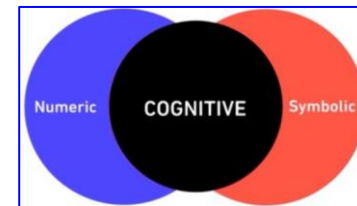
- ・説明文: tf-idfベクトル化
- ・ブランド: one-hotベクトル化
- ・価格: 標準化

標準化: 平均が0で分散が1のデータ
に変換する操作





ビッグデータ×ハイブリッドAI

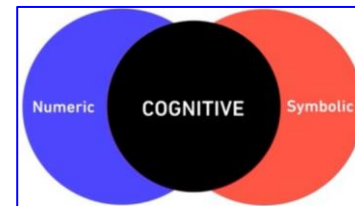


ハイブリッドAI: 知識/経験とAIをシンボリックとして組み合わせたもの

Hybrid AI is concerned with the integration of symbolic (logical) and sub-symbolic (DL-based) AI methodologies into neuro-symbolic architectures.



ビッグデータ×ハイブリッドAI



低炭素

潤滑油配合

需要/供給の予測

トレーディング

フィールド管理
生産管理

製油所/発電所の
運用/保守

検査/計測

自動車運転手
安全性

ウェアラブル
デバイス

ハイブリッドAI



エネルギー

再生
エネルギー

産業用
加工製造

製品配合

半導体

物流

金融

ヘルスケア

Source: https://www.macnica.co.jp/business/ai_iot/columns/136927/

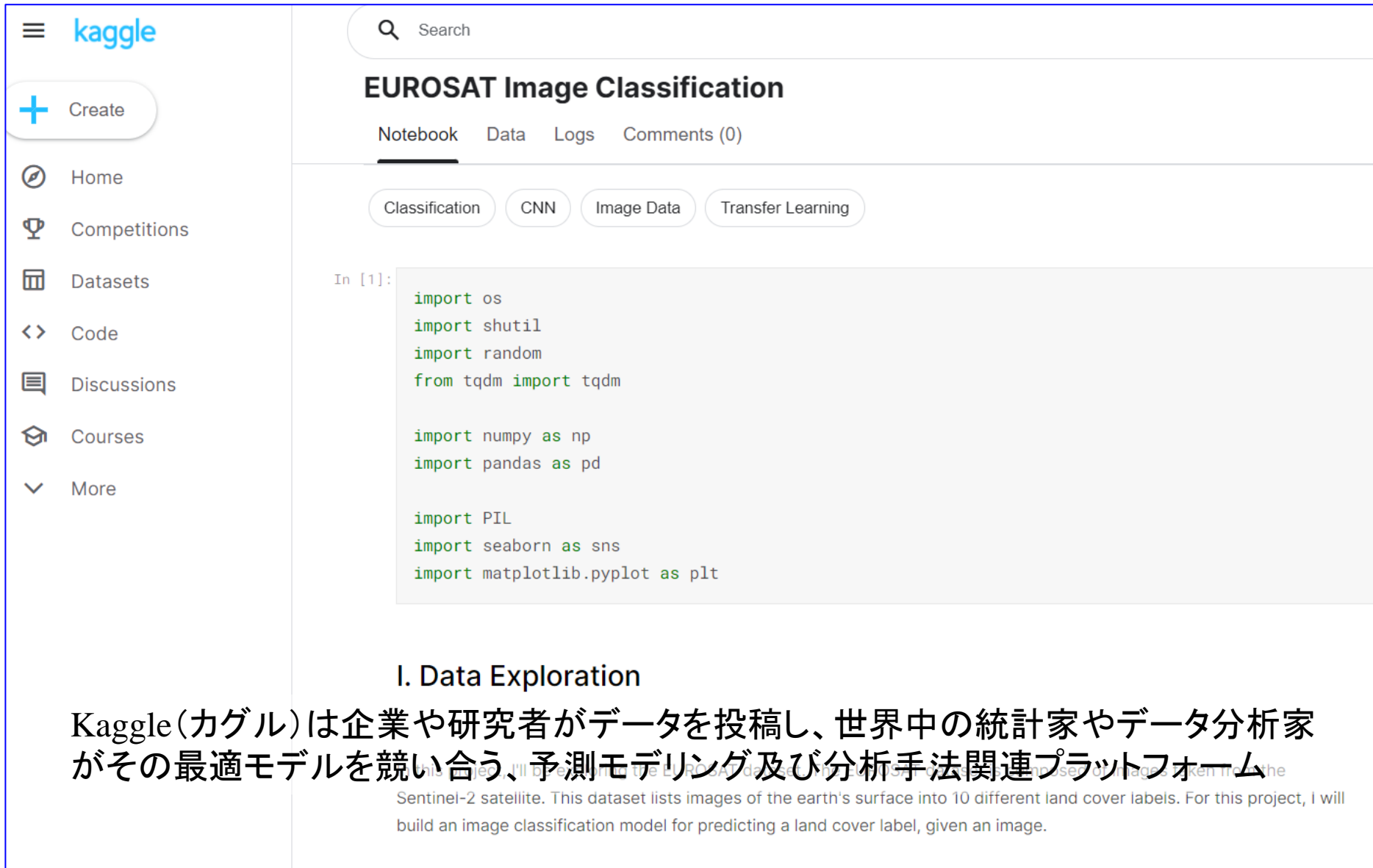
出典: BeyondLimits

BeyondLimits: NASA宇宙ミッション向けに開発されたテクノロジーを活用し複雑な問題を解決
REBOMIX: 米国のBeyond LimitsのAIソリューションの日本市場における認定総代理店

例: 新規の油井からの石油生産の最適化、既存の油井からの一貫した生産量の維持、
あるいは成熟期の油井からの効率的なロングテール生産のメリットを享受することが可能

Source: <https://www.beyond.ai/>

Kaggle



EUROSAT Image Classification

Notebook Data Logs Comments (0)

Classification CNN Image Data Transfer Learning

```
In [1]: import os
import shutil
import random
from tqdm import tqdm

import numpy as np
import pandas as pd

import PIL
import seaborn as sns
import matplotlib.pyplot as plt
```

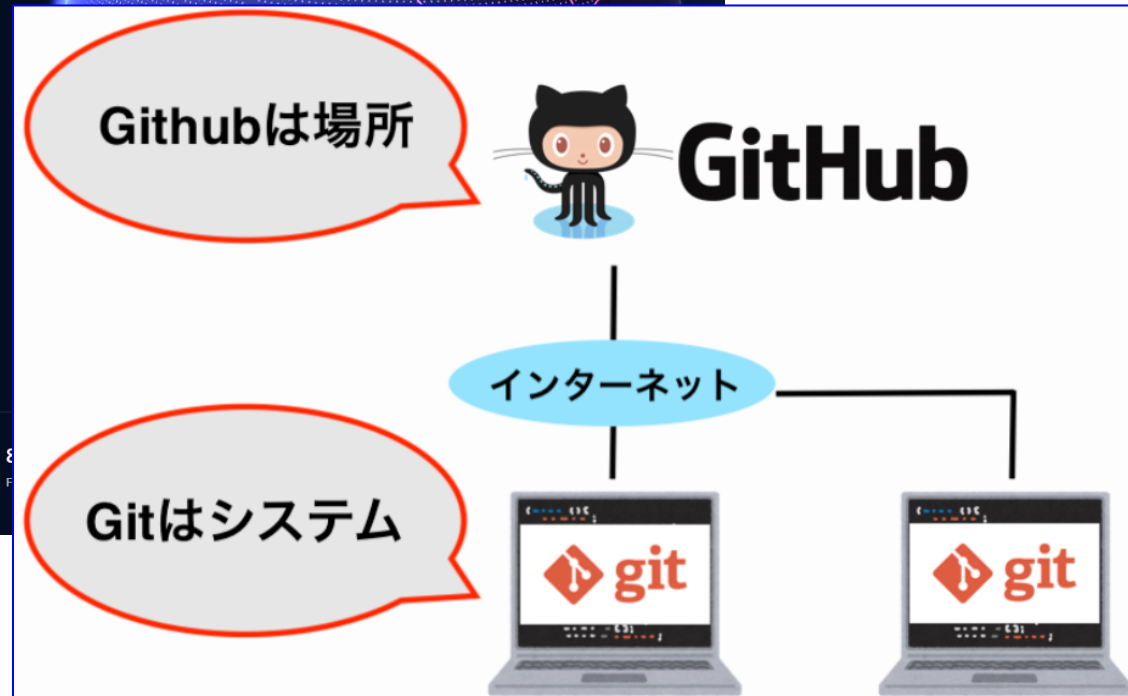
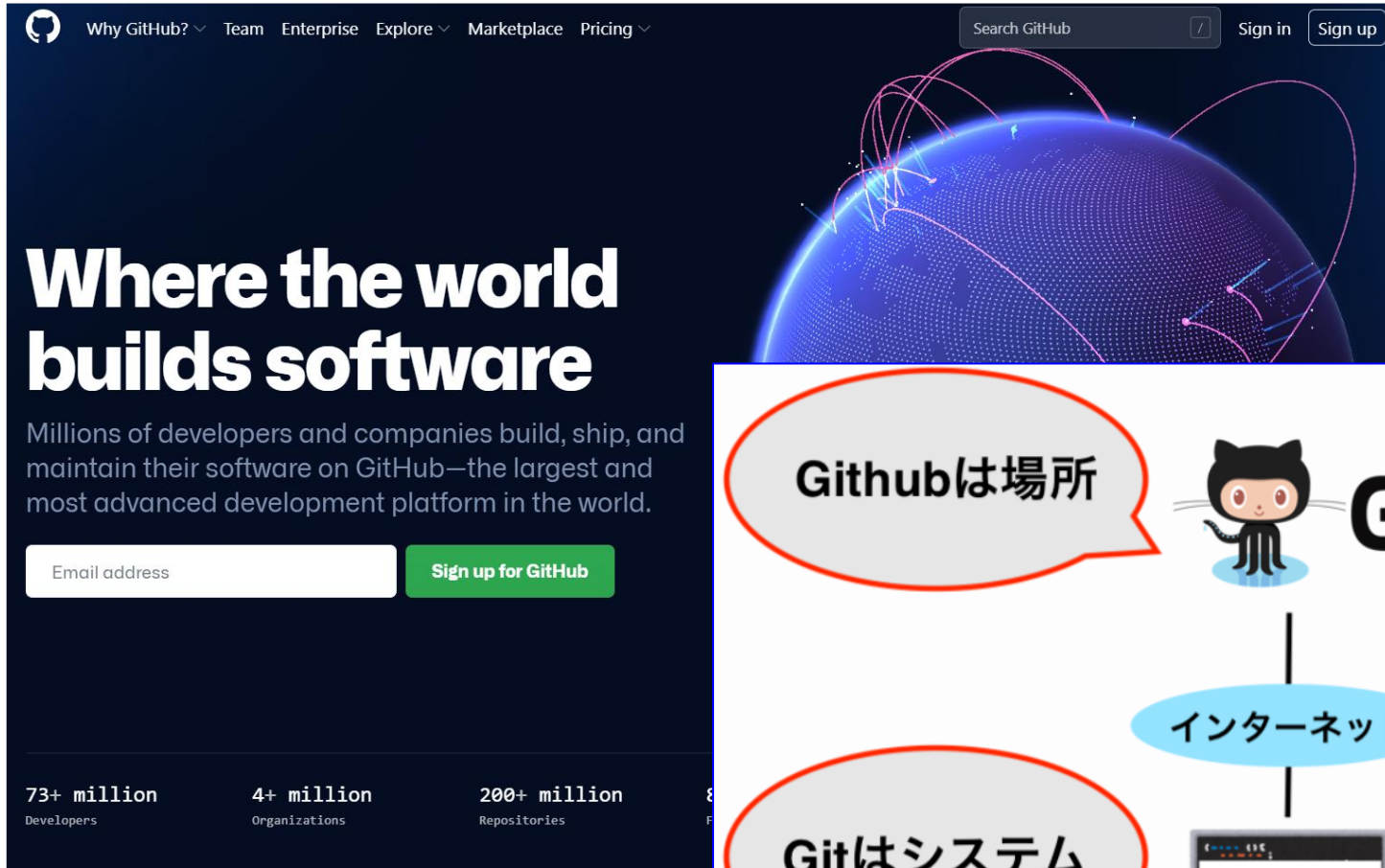
I. Data Exploration

Kaggle(カグル)は企業や研究者がデータを投稿し、世界中の統計家やデータ分析家
がその最適モデルを競い合う、予測モデリング及び分析手法関連プラットフォーム

...this dataset, I'll be examining the EUROSAT dataset, which contains 100,000 images taken from the Sentinel-2 satellite. This dataset lists images of the earth's surface into 10 different land cover labels. For this project, I will build an image classification model for predicting a land cover label, given an image.

Source: <https://www.kaggle.com/>

GitHub



Git: 各ローカルPCにインストールし、プログラムソースをバージョン管理できるシステム
GitHub: プログラムコードやデザインデータなどを保存、公開するウェブサービス

Source: <https://github.com/>; Source: <https://www.sejuku.net/blog/7901>