

機械学習 (6)

概要

1. 機械学習の基礎

- 概論
 - 教師あり学習
 - 教師なし学習
 - 強化学習

2. 教師あり学習

- ランダムフォレスト
- サポートベクターマシン

3. 教師なし学習と確率モデリング

- クラスタリング
- ナイーブベイズモデル
- 混合ガウスモデル
- クロスバリデーション・モデル選択

4. 各種データと機械学習

- テーブルデータ
- 行列データ
- 時系列データ
- グラフデータ

補助資料：<http://small-island.work/>

5. 深層学習

ニューラルネットワークの基礎
深層学習の基礎
画像処理・自然言語処理におけるモデル

6. 画像処理・自然言語処理（再訪）

画像処理基礎
自然言語処理基礎

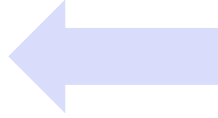
この講義の目的

目的

1. 実際のデータに対する処理をイメージできるようになる
2. 各データに対する基本的なアプローチを理解する
3. 画像処理や自然言語処理の難しさを知る
(なかなか人間のようにはできない)
4. 深層学習に持っていくまでの部分に重点を置く

データの種類

- 画像・動画
- 自然言語
- 音声



今回

- テーブルデータ
- 行列データ
- 時系列データ：
 - タイムスタンプの付きデータ
 - 信号データ：波形・振動、脳波、その他生体信号など
- グラフデータ、関係データ

画像データに対する処理

画像データの前処理

- 画像データはどこから来るのか？
- 画像データ的前提

光と色

→ 立体の認識

→ 画像の基本操作：
フィルタリング
画像局所特徴量

→ 深層学習による一般物体認識

画像データはどこから来るのか？画像データ的前提

電磁波

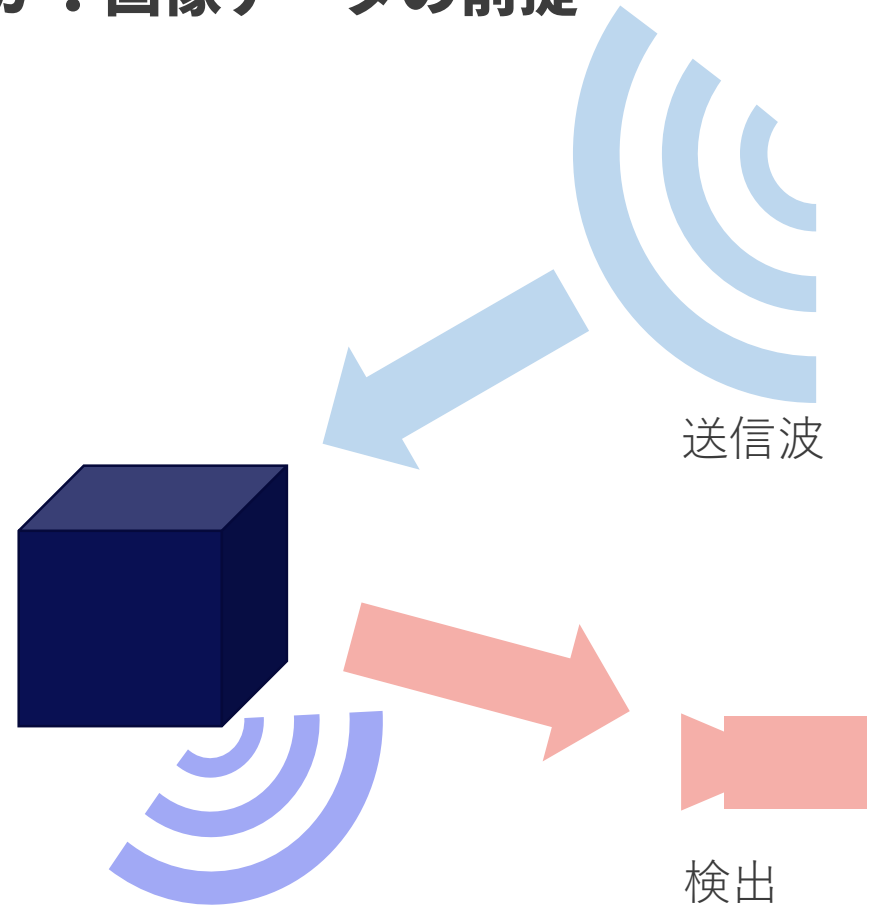
- 可視光カメラ
- X線画像、レントゲン画像
- ガンマ線画像
- 赤外線カメラ
- ハイパースペクトルカメラ

音波

- 超音波画像・ソナー画像

コンピュータグラフィックス

- シミュレーション画像

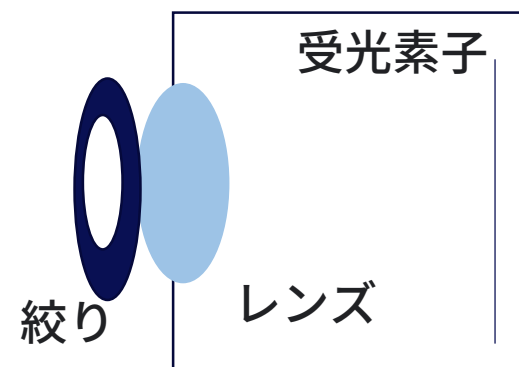
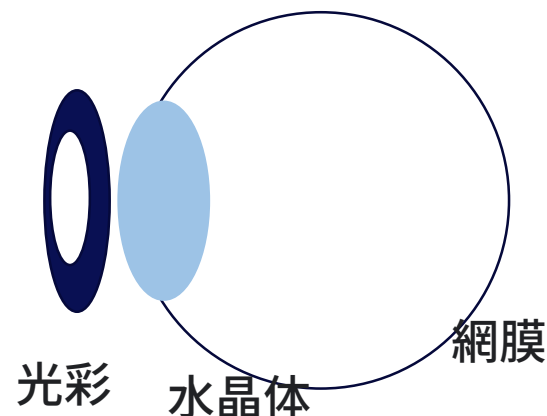


- 受動的に何かの反射を測定
- 能動的に送信波の反射を測定
- 計測対象自体が発している信号を検出

目とカメラの違い

カメラ	目	
レンズピント調整	角膜 水晶体	角膜が3/4~2/3の屈折力、水晶体の厚みを変えて1/4~1/3の屈折力範囲を可変調節（オートフォーカス）
受光素子（撮像）	網膜	網膜上に 桿体：約1億2000万個 錐体：約600-700万個中心に錐体が集中
絞り、露光調節	虹彩 錐体の役割分担	虹彩の直径2~6mm程度で調節（オート露光）

S錐体、M錐体、L錐体と呼びそれぞれおよそ青、緑、赤の波長を感知



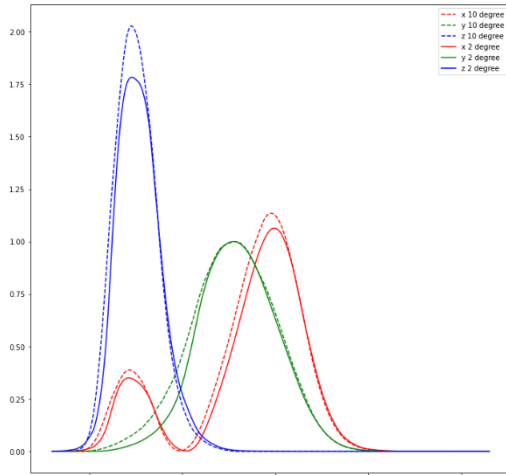
カメラは複数レンズのため、レンズ間の反射によってフレアやゴーストが発生する場合があります



収差：波長ごとにレンズの屈折率が異なる収差の影響を回避するために実際は複数レンズになっている

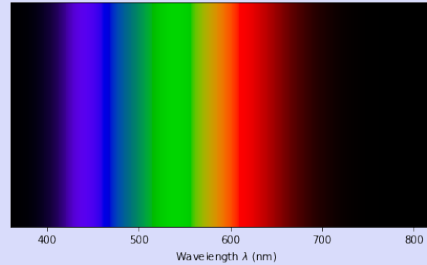
色(1/2)

色の見え方の標準を定めたい



CIE 1931 2 Degree Standard Observer
CIE 1964 10 Degree Standard Observer

The Visible Spectrum - CIE 1931 2° Standard Observer



ヤング＝ヘルムホルツの三色説
色別は3要素の刺激の比率に応じて生じる

グラスマンの法則

等色の加法性を前提（積分の線形性）
多くの実用上の測色問題では、加法性の原理は十分に正しい。

$$X = \int_0^{\infty} I(\lambda) \bar{x}(\lambda) d\lambda$$

$$Y = \int_0^{\infty} I(\lambda) \bar{y}(\lambda) d\lambda$$

$$Z = \int_0^{\infty} I(\lambda) \bar{z}(\lambda) d\lambda$$

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 0.41847 & -0.15866 & -0.082835 \\ -0.091169 & 0.25243 & 0.015708 \\ 0.00092090 & -0.00254980 & 0.17860 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

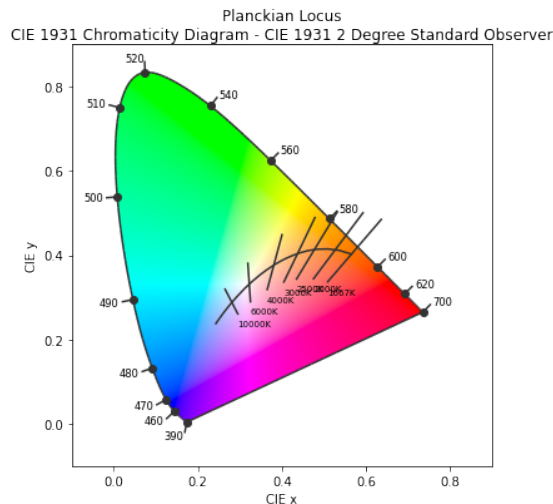
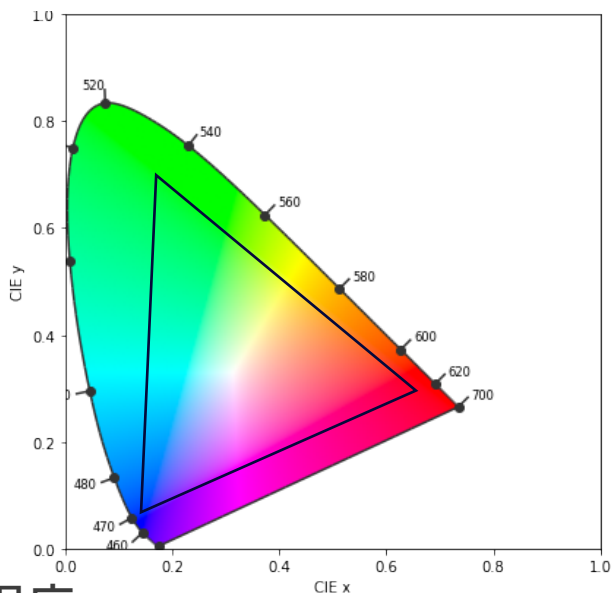
$$x = \frac{X}{X + Y + Z}$$

$$y = \frac{Y}{X + Y + Z}$$

$$z = \frac{Z}{X + Y + Z}$$

色(2/2)

本来、周波数ごとの足し合わせで表現できる範囲は以下のような範囲になるが RGBの3色の混合で表現しようとするると不可能な点が存在する。

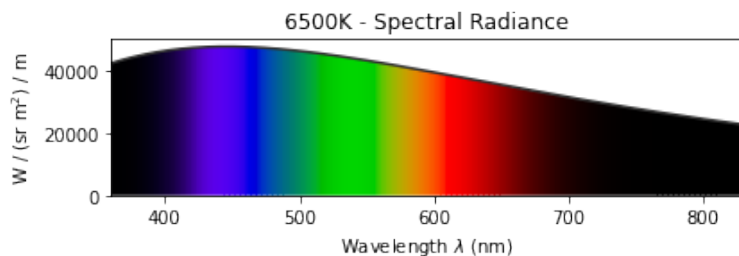


色温度
規準となる色を定量的にある温度の
黒体輻射で表現したもの
良く用いられるのがD65(6500 K)

プランクの法則

黒体から輻射される電磁波の分光放射輝度

$$I(\lambda, T) = \frac{2hc^2}{\lambda^5} \cdot \frac{1}{e^{hc/\lambda kT} - 1}$$



6500K - Colour



6500K

特定の色のものを認識する

例えばオレンジ色のものを認識したい時、RGBの混合でオレンジ色の範囲を指定するのは直観的に難しいのでRGB表現からHSV表現に変換することが多い

$$\text{MAX} = \max(R, G, B)$$

$$\text{MIN} = \min(R, G, B)$$

$$H = \begin{cases} \text{undifined} & \text{if MAX} = \text{MIN} \\ 60 \times \frac{G-R}{\text{MAX}-B} & \text{if MAX} = B \\ 60 \times \frac{B-G}{\text{MAX}-R} & \text{if MAX} = R \\ 60 \times \frac{R-B}{\text{MAX}-G} & \text{if MAX} = G \end{cases}$$

$$V = \text{MAX}$$

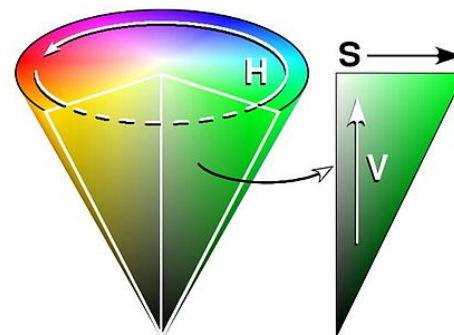
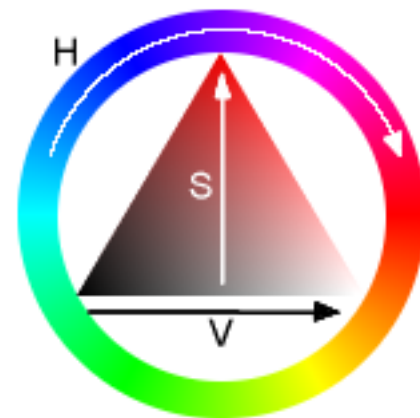
$$S = \text{MAX} - \text{MIN}$$

HSV表現

色相 (Hue / 色の種類)

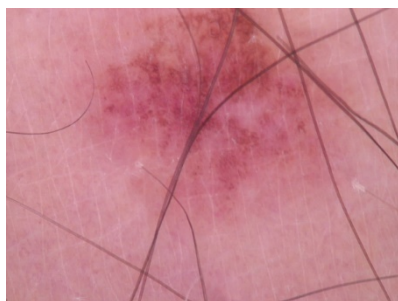
彩度 (Saturation / 鮮やかさ)

明度 (Value / 明るさ)



特定の色のものを認識する例

例えばターゲット部位を中心に画像を切り出したい場合



実際の画像からリアルタイムに特定の色を抜くのは単純に難しい

例：クロマキー合成

(実際はライティングや反射の調整、後処理などで調整することが多い)

画像データに対する処理

画像データの前処理

- 画像データはどこから来るのか？
- 画像データ的前提

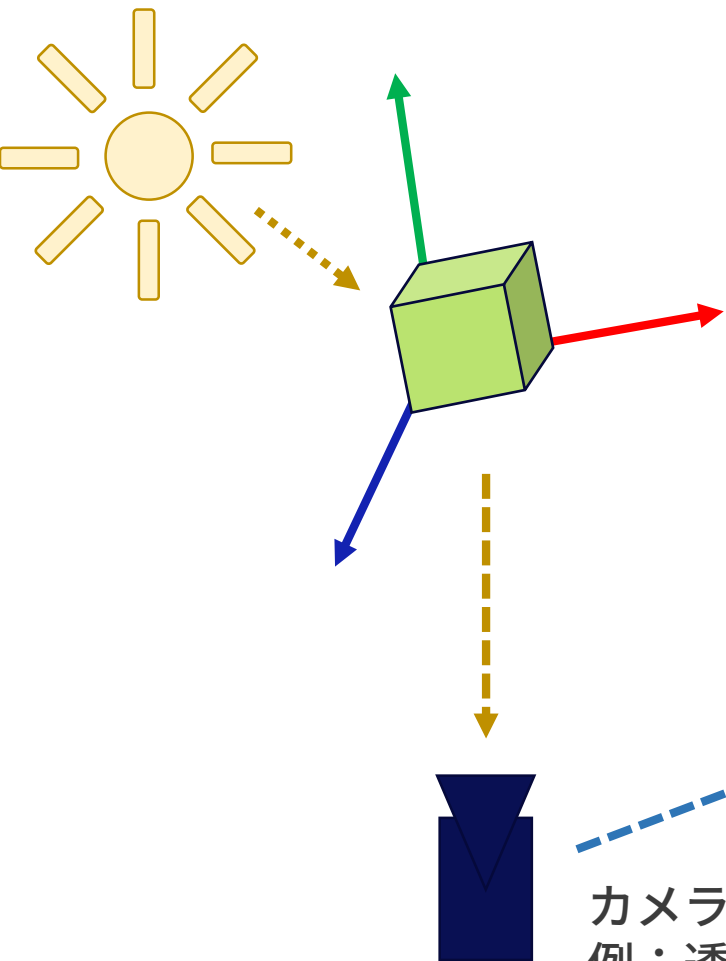
光と色

→ 立体の認識

→ 画像の基本操作：
フィルタリング
画像局所特徴量

→ 深層学習による一般物体認識

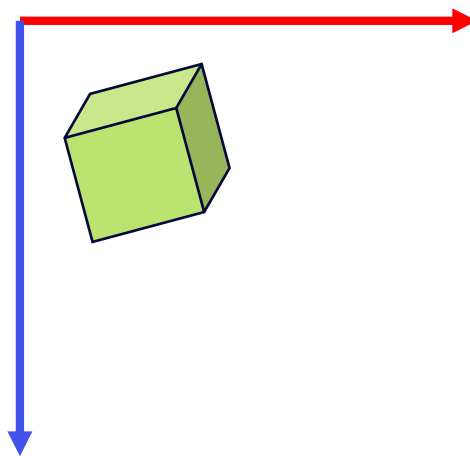
立体の認識



立体を認識する方法

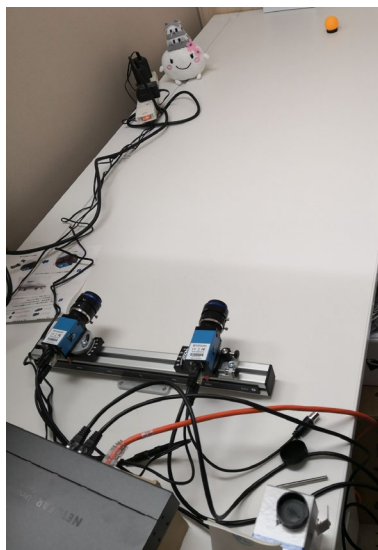
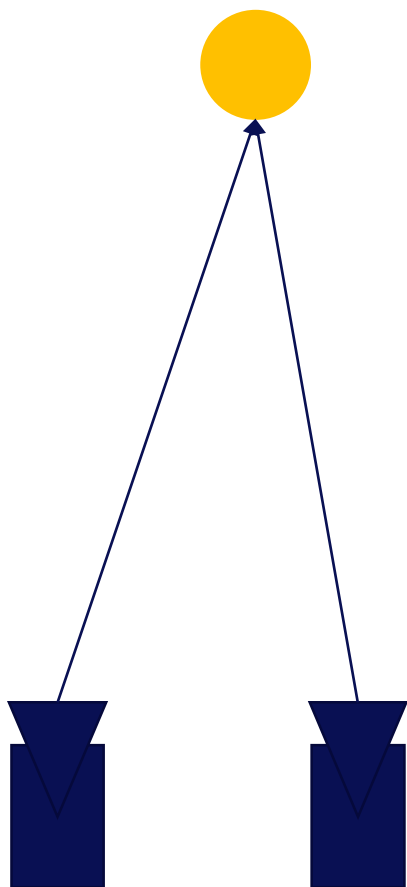
- 距離を直接計測する（デプスカメラ）
- 2次元から起こし上げる
- 2台以上のカメラを使う

カメラのモデル
例：透視投影変換

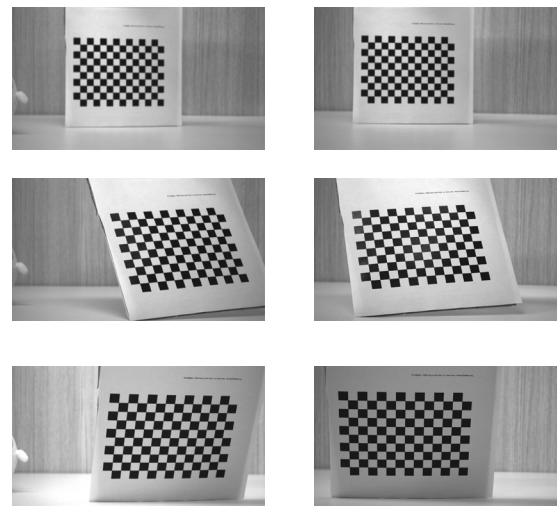


ステレオカメラ

例：特定の色を認識+2台のカメラで
ピンポン玉を追跡する

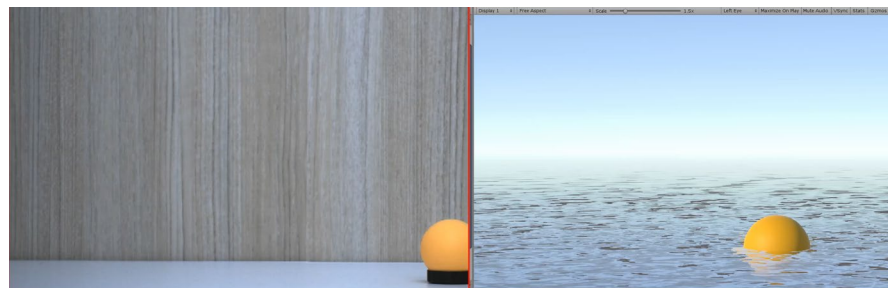


キャリブレーション
カメラの内部パラメータの推定
カメラ間距離・位置関係の推定



実際のカメラ画像
(片側)

バーチャル空間に
再構成したもの



画像データに対する処理

画像データの前処理

- 画像データはどこから来るのか？
- 画像データ的前提

光と色

→ 立体の認識

→ 画像の基本操作：
フィルタリング
画像局所特徴量

→ 深層学習による一般物体認識

フィルタリング

0.5	1.0	0.7
0.8	0.3	0.2
0.5	0.2	0.8
⋮	⋮	⋮
⋮	⋮	⋮

Image

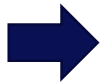
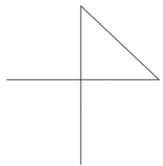
0.8	0.7
0.5	0.8

filter

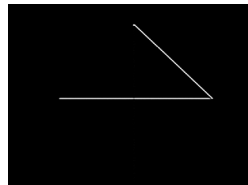
$$0.5 \times 0.8 + 1.0 \times 0.7 + 0.8 \times 0.5 + 0.3 \times 0.8$$

1.74	1.6
1.6	1.12
⋮	⋮
⋮	⋮

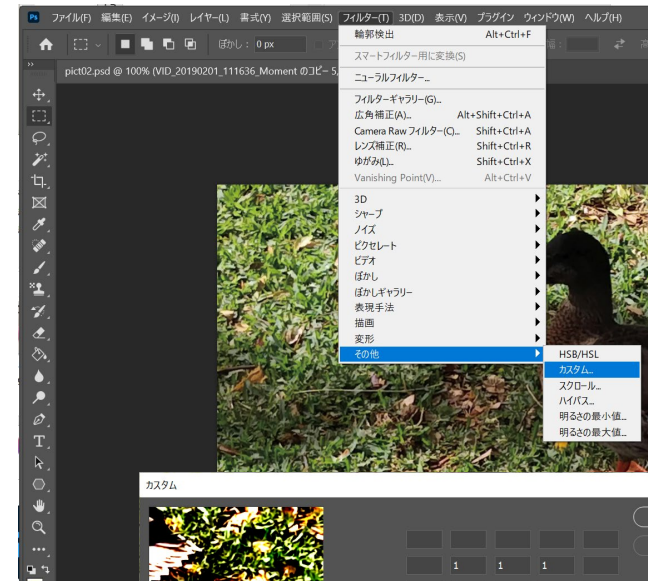
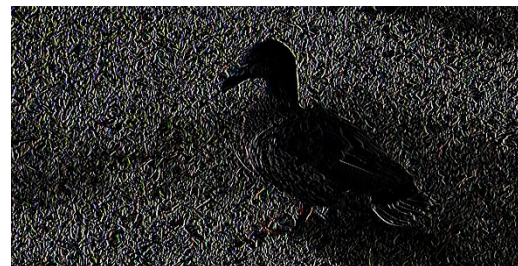
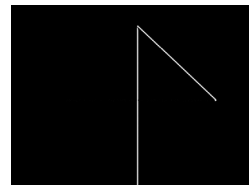
Image



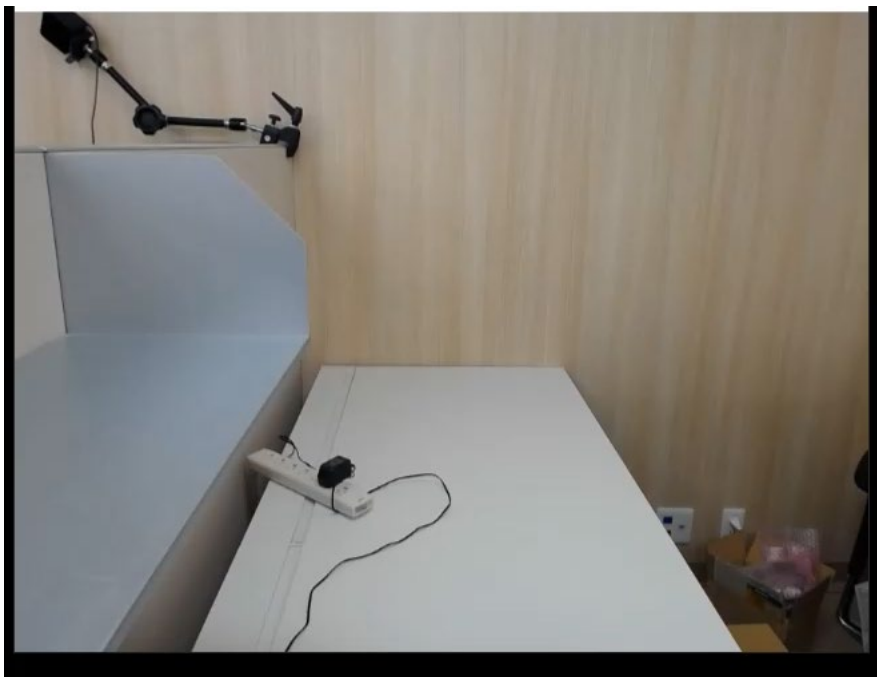
1	2	1
0	0	0
-1	-2	-1



-1	0	1
-2	0	2
-1	0	1



マーカーの認識(1/2)



特定の色の物体の認識
フィルタリング、エッジの抽出
↓
マーカーの発見
↓
カメラモデル・透視投影変換に基づく座標の推定

局所特徴量

画像中の特徴的な点を見つけて、
その点を表すベクトルを計算する

SHIFT:勾配方向のヒストグラムによる特徴記述
・スケール・回転・証明の変化に関してロバスト

Lowe, David G. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision* 60.2 (2004): 91-110.

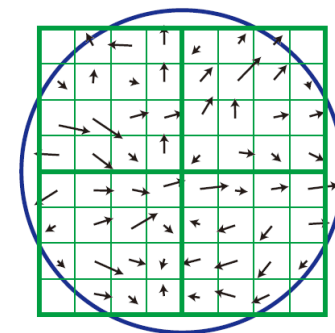
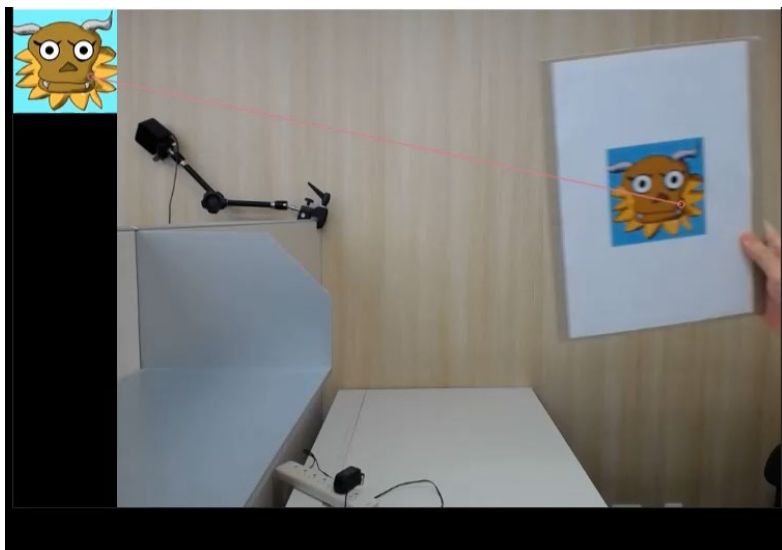


Image gradients

その他にも多くの特徴量がある: SURF, A-KAZE, ORB



マーカーの認識(2/2)



特定の色の物体の認識
フィルタリング、エッジの抽出
↓
局所特徴量マッチング
↓
マーカーの発見（ホモグラフィ行列の推定）
↓
カメラモデル・透視投影変換に基づく座標の推定

画像データに対する処理

画像データの前処理

- 画像データはどこから来るのか？
- 画像データ的前提

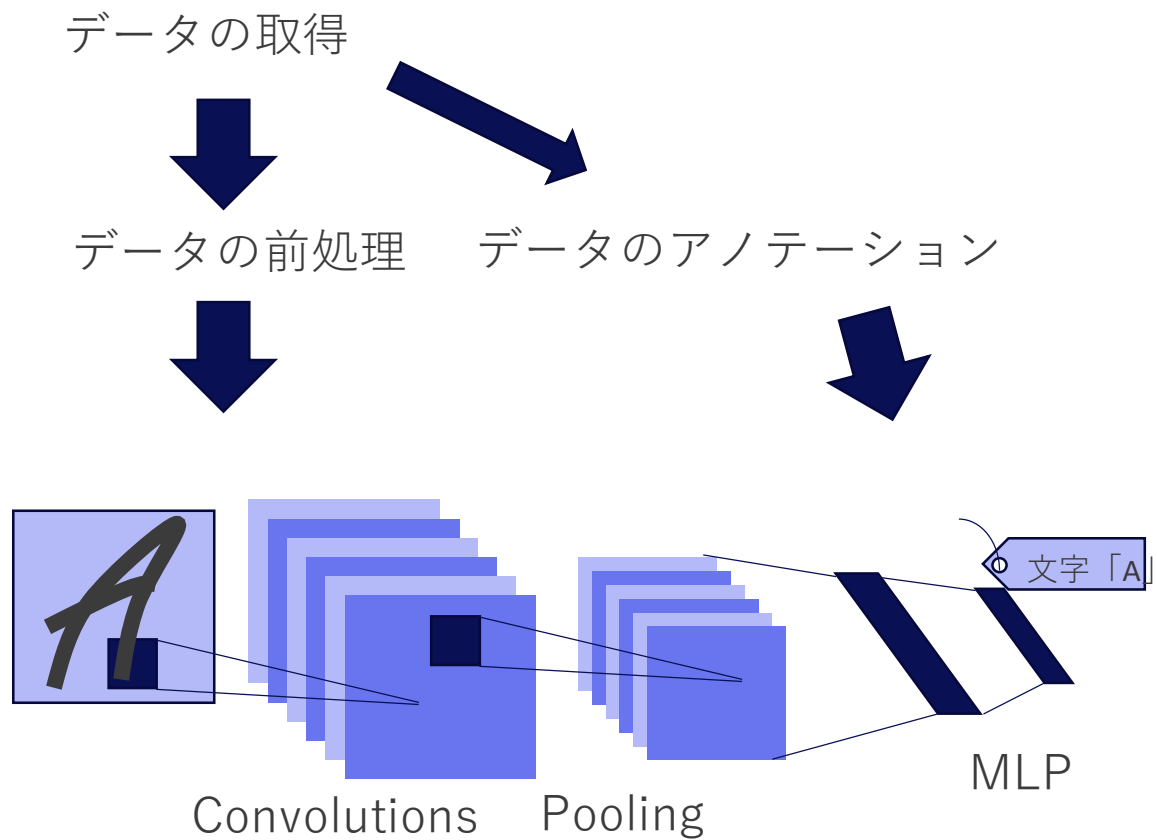
光と色

→ 立体の認識

→ 画像の基本操作：
フィルタリング
画像局所特徴量

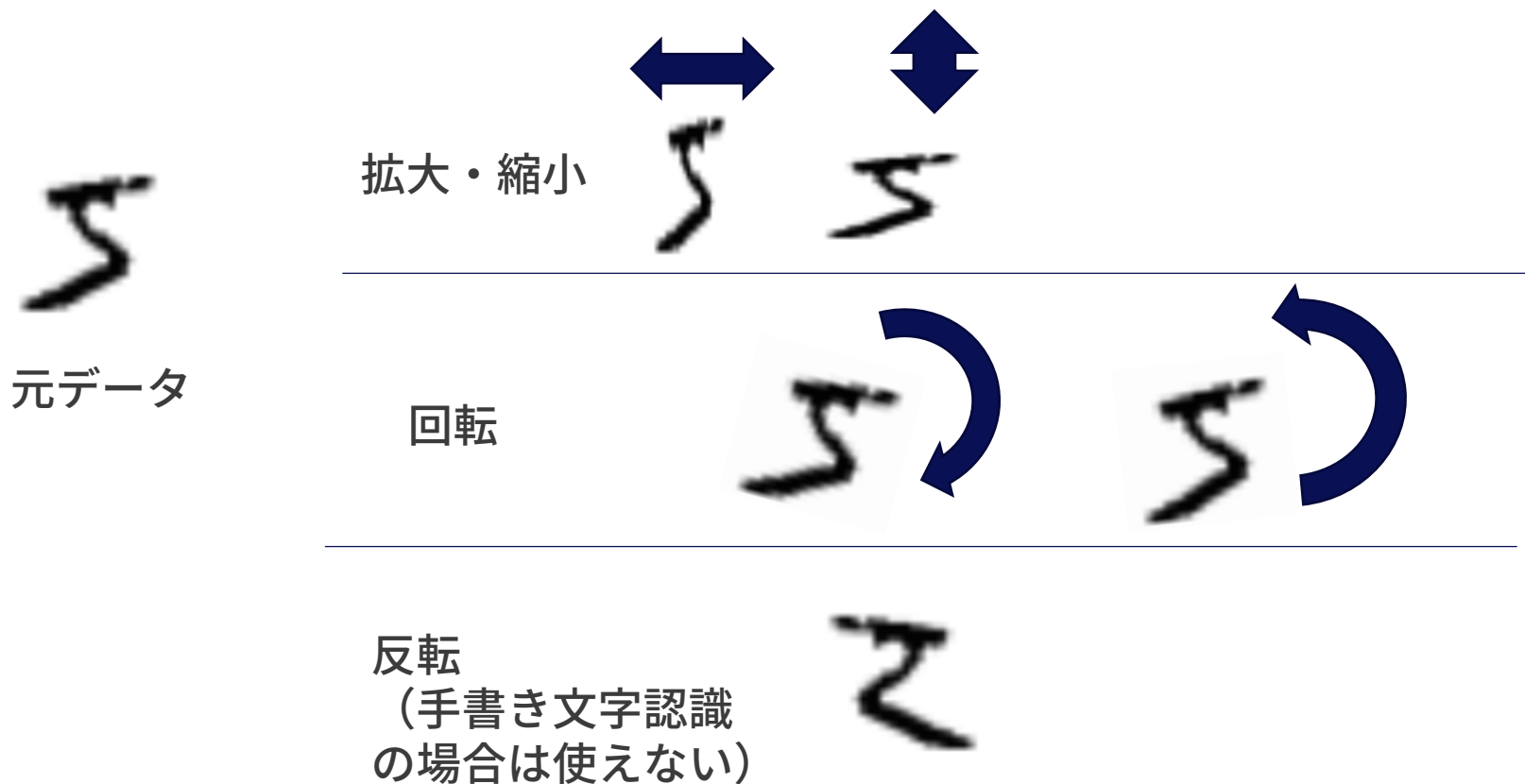
→ 深層学習による一般物体認識

深層学習システム



Data Augmentation

- データに変換・摂動を加えてデータを増やす手法
- 加えた変換によってラベルが変化しないという不変性を持たせられる



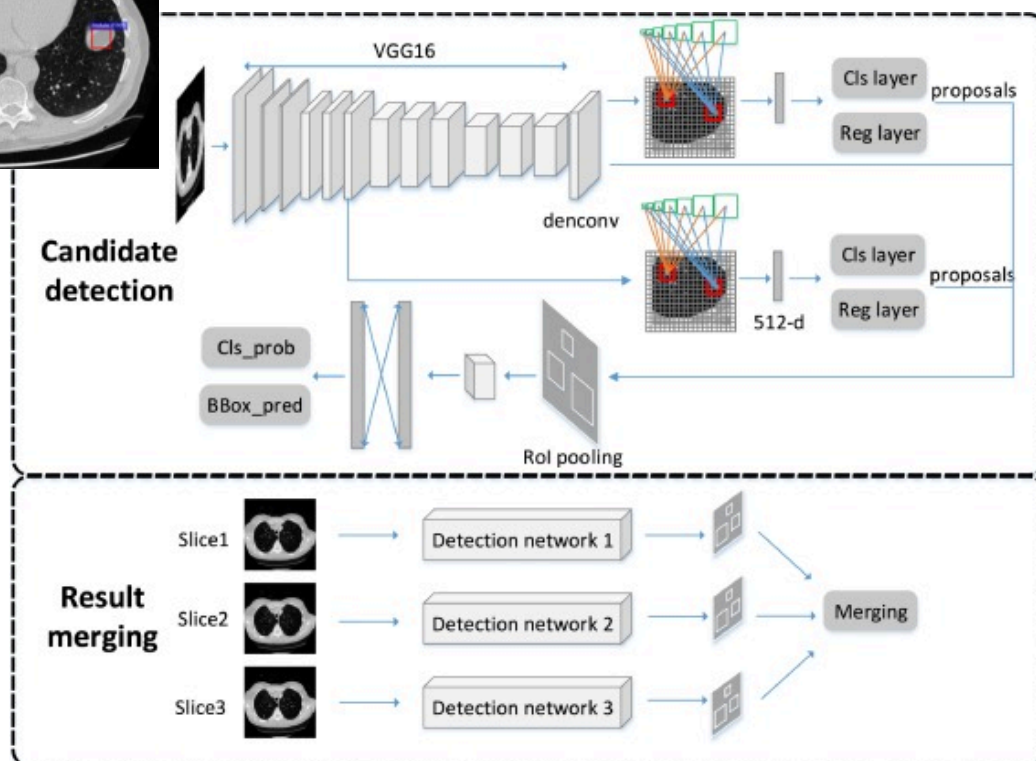
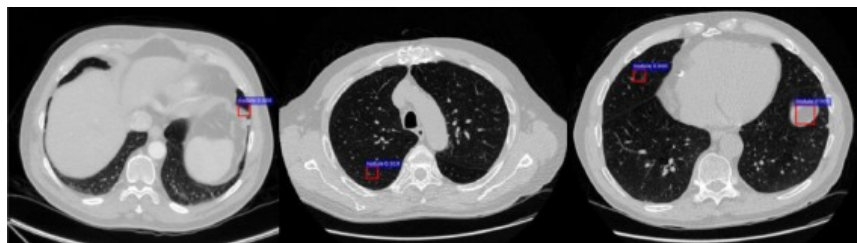
不偏性を考慮した正則化項とみなすことができる
画像などの場合には使いやすい

物体認識・物体検出の例

CT画像からの肺結節検出・認識（病変の自動鑑別）

深層学習にいれる前処理：

ダイナミックレンジ、スキャン感覚の統一、背景除去



Xie, Hongtao, et al. "Automated pulmonary nodule detection in CT images using deep convolutional neural networks." *Pattern Recognition* 85 (2019): 109-119.

自然言語データに対する処理

自然言語データの前処理

- 自然言語データはどこから来るのか？
- 自然言語データ的前提：言語資源、文字コード

→ 自然言語における前処理
基本的な処理

→ 特徴量

- 単語頻度
- tf-idf

→ 自然言語処理の深層学習：BERTの使い方

- 単語の識別：固有表現抽出
- 複数単語の関係：関係抽出
- 文の識別：感情分析
- 複数文の類似判定・論理関係判定：2文が意味的に同じか否かを判別、含意/矛盾/中立などを判定

自然言語

自然言語の例

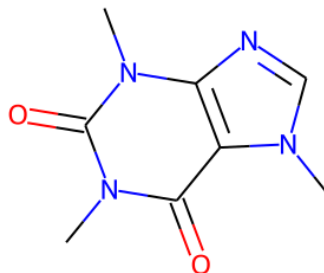
- メール
- ウェブページ
- レビュー
- 文書

これらのデータを用いての

- 情報検索
- 要約
- 分析

シンボルの系列を扱うもの全般

- 行動履歴
- ゲノム配列
- 化合物を表現するSMILES文字列
CN1C=NC2=C1C(=O)N(C(=O)N2C)C



例：入力が画像となっている場合もある
画像処理

- OCR (Optical Character Reader)
画像の文字認識
- 文書の構造認識
- 写真の中のテキスト

↓

自然言語処理

- 文書からの情報抽出

自然言語に含まれる様々な要素

曖昧性：品詞、語義、掛受け、…
「黒い尻尾の大きな犬」

連語：名詞の繋ぎ合わせ
「京都大学」

擬音語・擬態語：
「ふわふわ」

再帰構造：文の中に文の構造がある
「私が昨日食べたごはん」

論理構造：主語・述語関係など

新語・未知語

専門用語・固有名詞

言語資源（コーパス・辞書）の整備

以前は辞書がメインであったが、機械学習によってデータから学習できるようになって特に大規模コーパスが重要となっている

（ただし、大量のデータが用意できない場合や実用は辞書も併用することが多い）

ツリーコーパス(Penn Treebank)
対訳コーパス
Google n-gram コーパス

単語辞書(IPA辞書)
シソーラス (WordNet)
対訳辞書

こういったコーパス作成は非常に労力がかかる

・アノテータと呼ばれる人や専門に扱う企業なども存在する
人間の精度にも限界がある＝複数人間が異なるアノテーションを付ける可能性がある

→自然言語処理の分野では弱教師あり学習や転移学習などのコーパスにできるだけ頼らない機械学習も盛んにおこなわれている

文字コード

コンピュータ上で文字（キャラクタ）を利用する目的で各文字に割り当てられるバイト表現

アルファベットなどは7bit以内に収まるようにコードされている

日本語などの文字は2バイト以上を使ってコードされる

Unicode の文字符号化方式

Unicode は世界で使われる全ての文字を共通の文字集合にて利用できるようにしようという考えで作られている。

歴史的な文字、数学記号、絵文字なども含む

実際にコンピュータにどのように符号化するかはいくつかの種類がある

- UTF-*

文字	16進	文字	16進	文字	16進	文字	16進
NUL	0	DLE	10	SP	20	0	30
SOH	1	DC1	11	!	21	1	31
STX	2	DC2	12	"	22	2	32
ETX	3	DC3	13	#	23	3	33
EOT	4	DC4	14	\$	24	4	34
ENQ	5	NAK	15	%	25	5	35
ACK	6	SYN	16	&	26	6	36
BEL	7	ETB	17	'	27	7	37
BS	8	CAN	18	(28	8	38
HT	9	EM	19)	29	9	39
LF	0a	SUB	1a	*****	2a	:	3a
VT	0b	ESC	1b	+	2b	;	3b
FF	0c	FS	1c	,	2c	<	3c
CR	0d	GS	1d	-	2d	=	3d
SO	0e	RS	1e	.	2e	>	3e
SI	0f	US	1f	/	2f	?	3f

ASCIIコード対応表

その他の日本語の文字コード

- JIS コード
- EUC
- Shift_JIS

自然言語データに対する処理

自然言語データの前処理

- 自然言語データはどこから来るのか？
- 自然言語データ的前提：言語資源、文字コード

自然言語における前処理 基本的な処理

特徴量

- 単語頻度
- tf-idf

自然言語処理の深層学習：BERTの使い方

- 単語の識別：固有表現抽出
- 複数単語の関係：関係抽出
- 文の識別：感情分析
- 複数文の類似判定・論理関係判定：2文が意味的に同じか否かを判別、含意/矛盾/中立などを判定

よくやる前処理

文字コードの統一

nkf コマンドなどが古くから使われている

Stop wordの除去

もともとは検索エンジンでの言葉で、検索語として利用する可能性少なく、利用すると効率が低下するため、あらかじめ除外することを定めた語

日本語：「は」「の」「です」など

英語：「the」「a」「for」など

Stemming：語幹の抽出

日本語：「歩か（ない）」・「歩き（ます）」・「歩け（ば）」などと活用するため、いずれにおいても変化していない「歩」の部分を取り出したり、標準の「歩く」の単語にそろえる

英語：「-ed」「-ing」「-ly」などの取り除く

最近はこれらの手法を使わない手法も多く出てきているが、単語の出現頻度など古典的な統計量を分析する際には重要

より高度な処理

形態素解析：品詞の推定と日本語等の場合は単語の分かち書きも合わせて行う必要がある

恥	名詞,一般,*,*,*,恥,ハジ,ハジ
の	助詞,格助詞,一般,*,*,*,の,ノ,ノ
多い	形容詞,自立,*,*,形容詞・アウオ段,基本形,多い,オオイ,オーイ
生涯	名詞,副詞可能,*,*,*,生涯,シヨウガイ,シヨウガイ
を	助詞,格助詞,一般,*,*,*,を,ヲ,ヲ
送っ	動詞,自立,*,*,五段・ラ行,連用タ接続,送る,オクツ,オクツ
て	助詞,接続助詞,*,*,*,て,テ,テ
来	動詞,非自立,*,*,カ変・来ル,連用形,来る,キ,キ
まし	助動詞,*,*,*,特殊・マス,連用形,ます,マシ,マシ
た	助動詞,*,*,*,特殊・タ,基本形,た,タ,タ
。	記号,句点,*,*,*,。 ,。 ,。
EOS	

MeCabの結果
<https://taku910.github.io/mecab/>

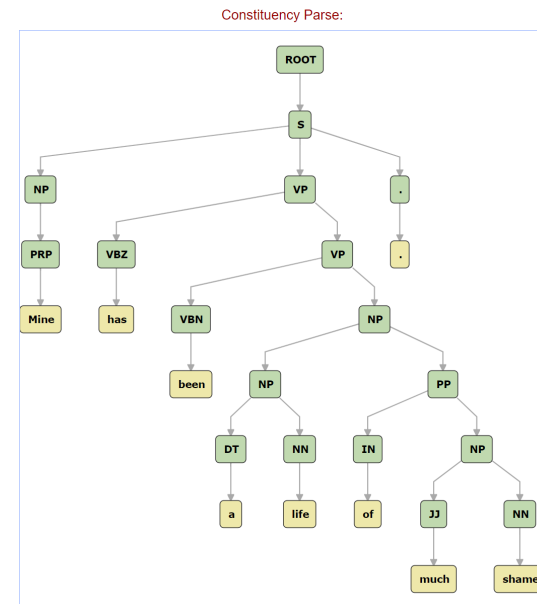
Stanford parserの例
<http://corenlp.run/>

依存関係解析・係り受け解析：
 単語の係り受けの関係を見つける

構文解析：文法に基づく構文木を見つける

述語項構造情報：

述語: 多い	格: ガ, 項: 恥 (項の基本句ID: 1)
	格: ガ2, 項: 生涯 (項の基本句ID: 3)
述語: 送って来ました。	格: ガ, 項: 私 (項の基本句ID: 0)
	格: ヲ, 項: 生涯 (項の基本句ID: 3)



自然言語データに対する処理

自然言語データの前処理

- 自然言語データはどこから来るのか？
- 自然言語データ的前提：言語資源、文字コード

自然言語における前処理
基本的な処理

特徴量

- 単語頻度
- tf-idf

自然言語処理の深層学習：BERTの使い方

- 単語の識別：固有表現抽出
- 複数単語の関係：関係抽出
- 文の識別：感情分析
- 複数文の類似判定・論理関係判定：2文が意味的に同じか否かを判別、含意/矛盾/中立などを判定

単語頻度

ジップの法則(Zipf's law)

大量のテキストに使用された単語の頻度を
集計し、その順位 r と頻度 f の間には次の
関係がある

$$f \approx \frac{c}{r}$$

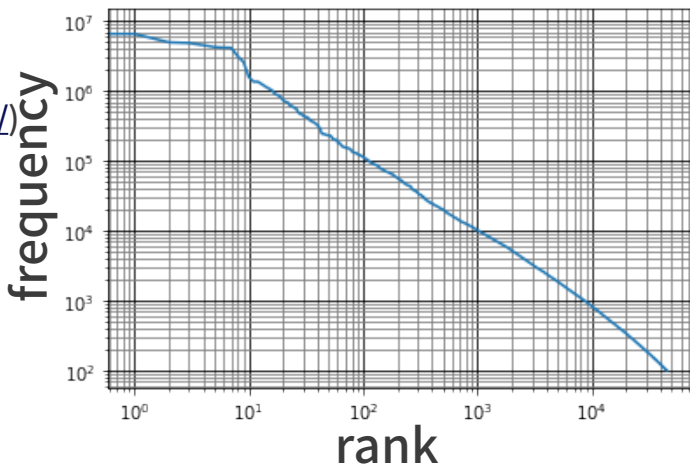
$$f \approx \frac{c}{r^a}$$

$$\log f = -a \log r + \log c$$

青空文庫

(<https://www.aozora.gr.jp/>)

13897作品中の
単語ごとの頻度
単語の種類：
136,835,329



名詞



動詞



文書に特徴的な単語

文書集合全体のなかから特定の文書に特徴的な単語を取り出したい

(単に出現頻度が高いわけではなく)

Term Frequency

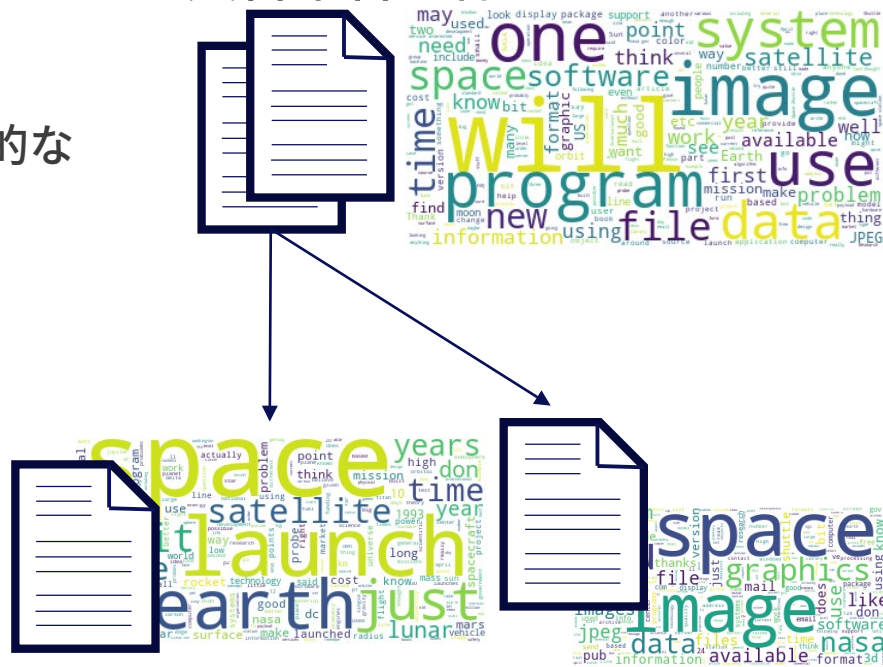
$$tf = \frac{\text{文書 A における単語 X の出現数}}{\text{文書 A における全単語数}}$$

Inverse Document Frequency

$$idf = \log\left(\frac{\text{全文書数}}{\text{単語 X を含む文書数}}\right)$$

$$tf-idf = idf \cdot tf$$

文書集合全体



tf-idf



自然言語データに対する処理

自然言語データの前処理

- 自然言語データはどこから来るのか？
- 自然言語データ的前提：言語資源、文字コード

→ 自然言語における前処理
基本的な処理

→ 特徴量

- 単語頻度
- tf-idf

→ **自然言語処理の深層学習：BERTの使い方**

- 単語の識別：固有表現抽出
- 複数単語の関係：関係抽出
- 文の識別：感情分析
- 複数文の類似判定・論理関係判定：2文が意味的に同じか否かを判別、含意/矛盾/中立などを判定

自然言語処理関連の有名なモデル

- GNMT (Google's NeuralMachine Translation) [Wu+ 2016]
Encoder-decoder model: 8-layer LSTM + residual connection
- Elmo[Matthew+ 2018]
- Transformer [Vaswani+ 2017]
Attentionのみを使用したニューラル翻訳モデル
- BERT[Devlin+2018]
BERTは、大規模コーパスから事前学習（単語補間のタスクと隣接文予測のタスク）させたTransformerモデルに、目的のタスクごとに再度学習をしたら多くのタスクで高い得点を出した
- GPT-2[Radford+2019]
 - Transformer ベースの48層NNを800万文書で学習
 - 特に生成で有名

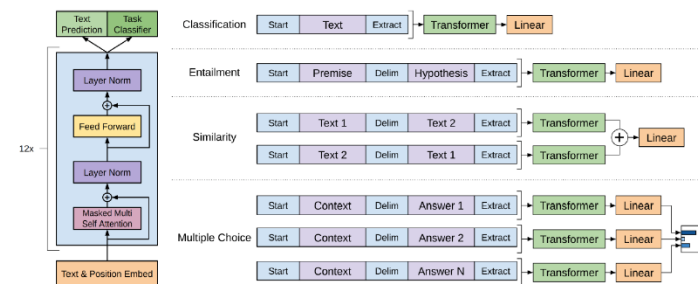
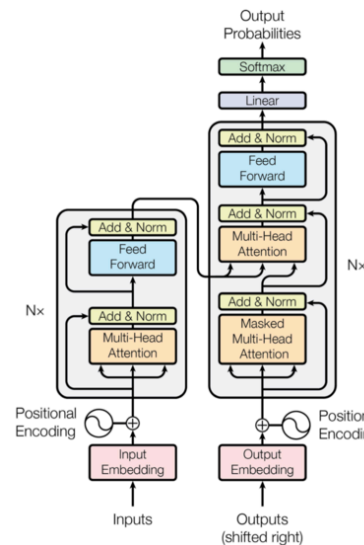
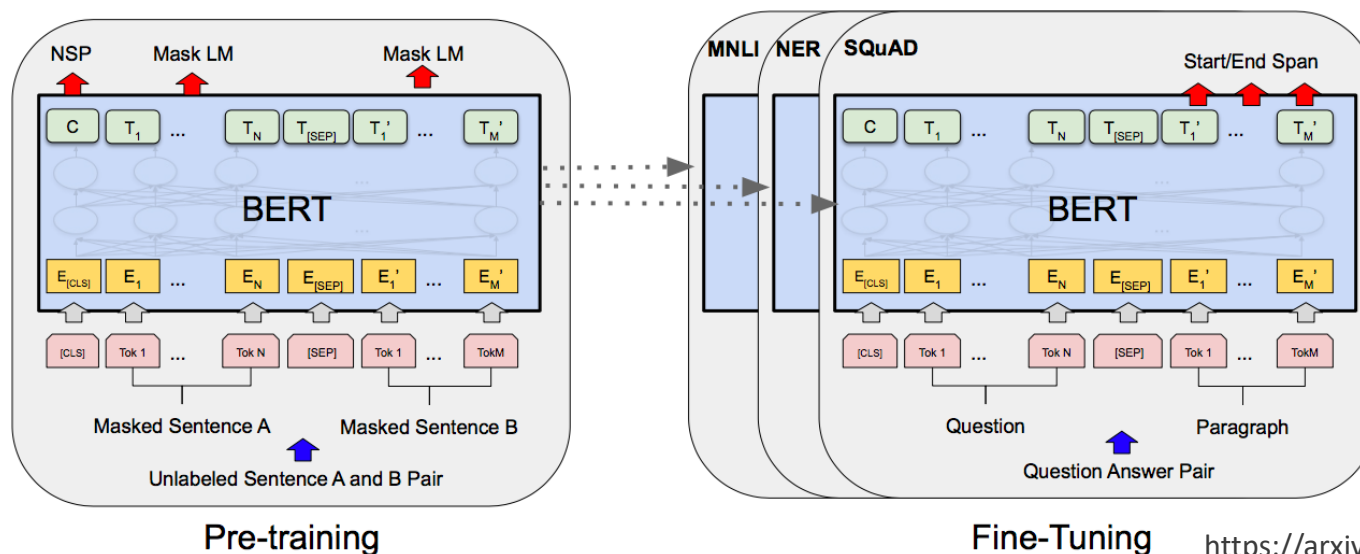


Figure 1: (left) Transformer architecture and training objectives used in this work. (right) Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

BERTの構造

Bidirectional Encoder Representations from Transformers

「Transformersからの双方向エンコーダ」



Fine-Tuning <https://arxiv.org/abs/1810.04805>

MLM (Masked Language Modeling) と NSP (Next Sentence Prediction) を学事前習 (Pre-training) させることで精度向上 11個のタスクで従来手法を上回る性能

MLM: 文章中の単語を一部マスクして周辺からその単語が何かを予測するタスク
NSP: 2つの文章をインプットして、それらが連続した文章かそうでないかを予測
これらの二つのタスクを学習するためには人手による教師データが必要ない

BERTを使ってできること

GLUE

(The General Language Understanding Evaluation)

複数のNLPタスクのベンチマーク

(平均をスコア化)

- MNLI : 2文が含意関係かどうかの分類タスク
- QQP : 質問内容が同じかを分類するタスク
- QNLI : 2文が質問と回答のペアかの分類タスク

ク

- SST-2 : 映画レビューに対する感情分析タスク
- CoLA : 文の言語的に正しいか分類タスク
- STS-B : 2文の類似度を5段階評価するタスク

- MRPC : 2文の意味が同じかを当てる分類タスク
- RTE : 小規模な含意関係の分類タスク

+

SQuAD : 質疑応答タスク

CoNLL : 固有表現抽出タスク

SWAG : 入力分に後続する分を 4 候補から選択する

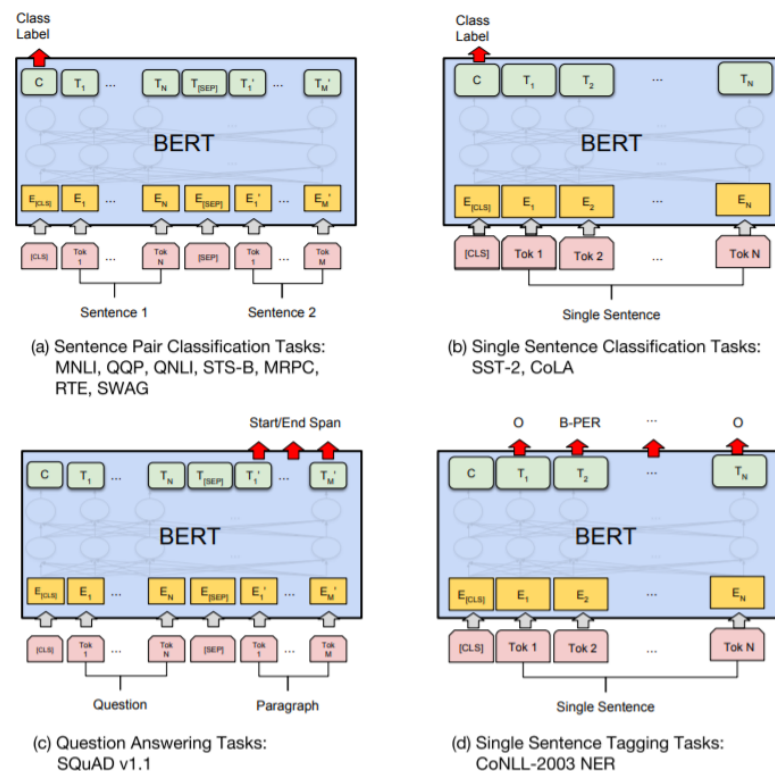
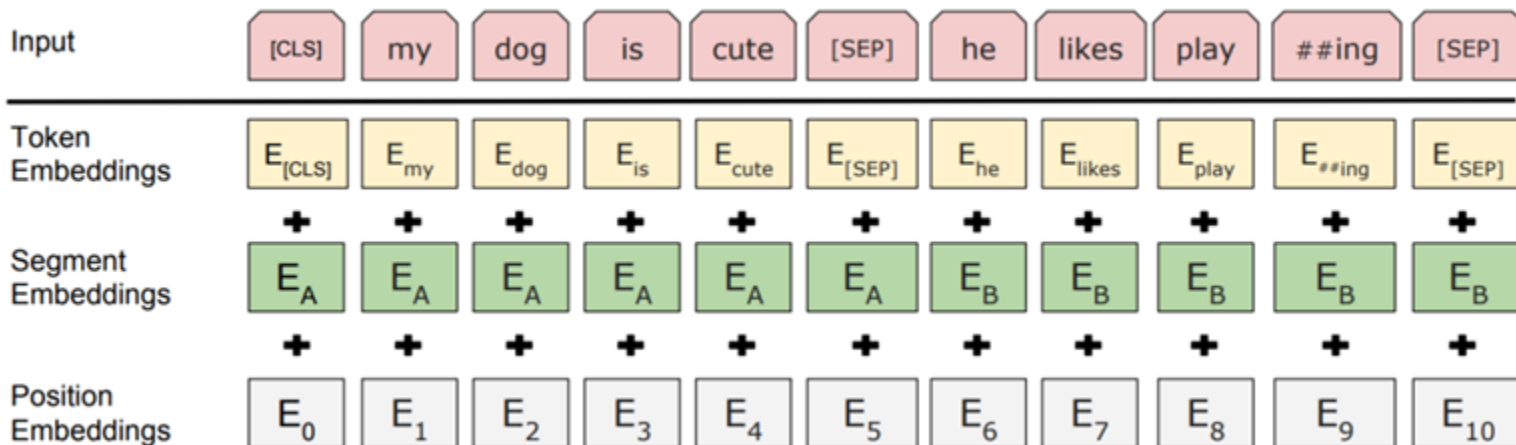
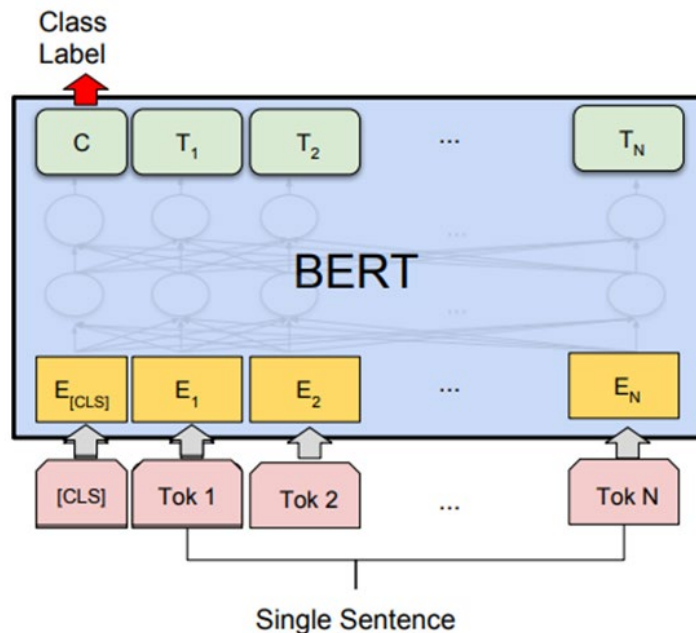


Figure 4: Illustrations of Fine-tuning BERT on Different Tasks.

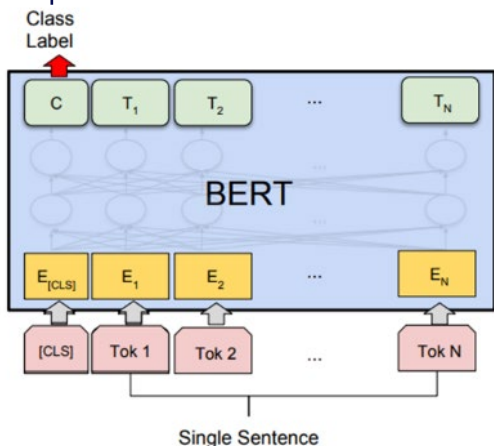
BERTの構造

実際の実出力
(ミニバッチサイズ, 系列長, ベクトル
次元数 2)



例：文章のclassification(感情分析)

ポジティブ/ネガティブ？



恥の多い生涯を送って来ました。	ネガティブ	0.82
自分には、人間の生活というものが、見当つかないのです。"	ネガティブ	0.88
自分は東北の田舎に生まれましたので、汽車をはじめて見たのは、よほど大きくなってからでした。	ポジティブ	0.95
自分は駐車場のブリッジを、上って、降りて、そしてそれが線路をまたぎ越えるために造られたものだという事には全然気づかず、.....	ポジティブ	0.77
しかも、かなり永い間そう思っていたのです。	ポジティブ	0.80
ブリッジの上ったり降りたりは、自分にはむしろ、ずいぶん垢抜あかぬけのした遊戯で、それは鉄道.....	ポジティブ	0.78

使い方：実践用

- BERT系のモデルを使う場合には専用のトークナイザを利用する（自前で前処理はしないことが多い）
- 言語ごとにモデルがある（日本語）
- 学習データを工夫したものもある（BioBERT, Clinical BERT）

基本的な使い方は、学習済みモデルを読み込み、
手元のデータで追加で再学習して使う

学習済みモデルもいくつか公開されている

<https://huggingface.co/models?search=Japanese>

自然言語データに対する処理

自然言語データの前処理

- 自然言語データはどこから来るのか？
- 自然言語データ的前提：言語資源、文字コード

自然言語における前処理
基本的な処理

特徴量

- 単語頻度
- tf-idf

自然言語処理の深層学習：BERTの使い方

- 単語の識別：固有表現抽出
- 複数単語の関係：関係抽出
- 文の識別：感情分析
- 複数文の類似判定・論理関係判定：2文が意味的に同じか否かを判別、含意/矛盾/中立などを判定

まとめ

- 画像・自然言語ともに、計算機でできることと人間ができることに差があるので注意する
- 深層学習が出てきて分析の方法が大きく変わりつつある：
深層学習にどうもっていくか？深層学習を組み合わせ何を実現するか？
深層学習が必要な問題か？
- それぞれのデータに応じて様々な手法があるので、大まかには知っておく必要がある