立教大学
RIKKYO UNIVERSITY

研究紹介

2021/08/16

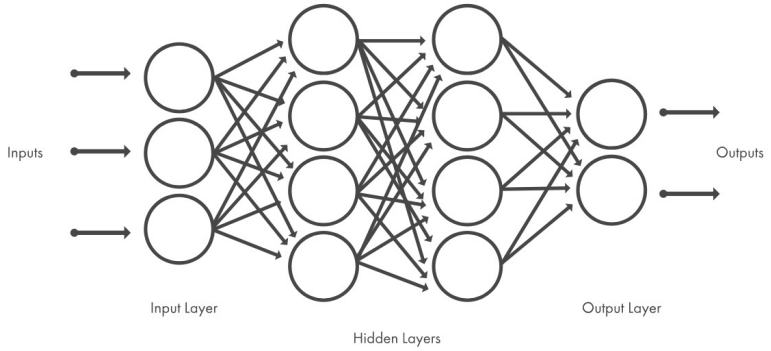
長谷川凌太郎
瀧研究室

目次

- 深層学習について
- 研究テーマについて
- 主に用いる技術
- 研究環境
- 深層学習におけるアンサンブル法
- 知識の蒸留
- 今後の展望

深層学習

- 深層学習（ディープラーニング）
- 人間の神経細胞の仕組みを数値的に再現したニューラルネットワークを用いた機械学習手法の一つ。多層構造のニューラルネットワークを用いることが特徴。
- モデル自らが考えて答えを推論することができるようにしていく。
- 入力データに対応して予測が出力されるため、その予測と正解との差分を最適化していく。



研究テーマについて

- 深層学習モデルが学習を通して人間の文化（思考様式やモノの見方）を継承しているのではないか、という仮説に基づき、異文化コミュニケーション学的観点から文化を継承した深層学習モデル同士の新たな連携性を構築する。
- 複数の深層学習モデルを活用する手法では、精度の向上・騙されにくくなる・説明性を改善する可能性を持つ。
- 将来的な研究の狙いは、
  - 文化（思考様式）の継承による、人間とのコミュニケーションの改善。
  - 多様なAI間での知識の交換や議論のような連携性の構築。

研究テーマについて

「異文化コミュニケーション学的な知見をAIに応用できるのではないか」

という疑問からスタート。

- 異文化間コミュニケーションや文化の継承、などの考え方が人工知能と非常にマッチしている。

- 単一の推論モデルを１つの文化を内包した個体とみなした時に、複数の推論モデル間で何らかのコミュニケーションパスを構築することができれば、精度の向上や現状の深層学習技術では難しい説明性の改善に貢献できるのではないか。

「文化」とは行動・思考様式のこと。

研究テーマについて

目指すもの

＜精度の向上＞

- キーポイントは多様性を活かす。
- 多様性があることによって、様々な視点の多様な回答・意見が出現、見落としが無くなる、状況が変わっても対応できる、などの利点が存在する。
- 組織やグループでその多様性を活かすために、リーダーシップ論・ファシリテート手法・異文化コミュニケーション学が存在する。
- このような考え方や理論を人工知能にもうまく適用すれば、精度の向上に繋がるのではないか。

研究テーマについて

＜説明性＞

- 現状の深層学習モデルでは説明性はなく、入力に対して推論が出力される。
- 推論に強く使われた画像の部位は表示できるが、判断根拠は示すことができない。
- 文化の隠れている部分は無意識の領域。他の文化と対比することによって初めて認識し、説明ができるようになる。
- 異なる文化(教師の文化)を継承した推論モデル同士での対比ができる環境を構築することによって説明性を付加することができるのではないか。

→今まで導入できなかったところへAIを活用できるようになる。

主に用いる技術

- 研究環境
- 深層学習モデルのアンサンブル法
- 知識の蒸留

研究環境

- プログラミング言語：Python
- 深層学習ライブラリ：Tensorflow / keras
- 主に使用するデータセット：CIFAR、Stanford Dog Dataset、Tiny ImageNet
- GPU：GeForce RTX 2070 SUPER

深層学習におけるアンサンブル法

- アンサンブル法
- 複数の深層学習モデルの予測確率を平均（ソフトアンサンブル）したり、多数決（ハードアンサンブル）を取ったりすることで精度が上がる傾向にある。
- このとき、各弱学習器がお互いに似ていない、多様性があるほど精度が上がる傾向にある。
  - Tensorflow.kerasでモデル作成、学習
  - 弱学習器はすべて同じモデルで同じように学習（Conv7層Dense1層）
  - ソフト、ハード、他３種類の方法でのアンサンブル。
  - 評価指標はaccuracyを基準に。

深層学習におけるアンサンブル法

	予測確率	平均(ソフト)	多数決(ハード)
弱学習器	→(0.6,0.4)		(1,0)
弱学習器	→(0.6,0.4)	(0.43,0.56)	(1,0)
弱学習器	→(0.1,0.9)		(0,1)
	A or B	予測結果:B	予測結果:A

深層学習におけるアンサンブル法

```python
# 確率の平均を取るアンサンブル（ソフトアンサンブル）
def ensembling_soft(models, X):
    preds_sum = None
    for model in models:
        if preds_sum is None:
            preds_sum = model.predict(X)
        else:
            preds_sum += model.predict(X)
    probs = preds_sum / len(models)
    return to_categorical(np.argmax(probs, axis=1), num_classes=10)

# 多数決のアンサンブル（ハードアンサンブル）
def ensembling_hard(models, X):
    pred_labels = np.zeros((X.shape[0], len(models)))
    for i, model in enumerate(models):
        pred_labels[:, i] = np.argmax(model.predict(X), axis=1)
    return to_categorical(mode(pred_labels, axis=1)[0], num_classes=10)
```

models変数に5つの弱学習器のベストなモデルが格納されている

+= model.predict(X)によってモデル予測の確率を足してく

/len(models)によって平均をとる

argmaxによって一番高い確率のindexを取得
→番号なので、to_categorical関数でone-hotに

enumarate()はインデックス番号も保持しながら取り出せる

argmaxで予測値を１つ選び
→各学習器ごとに行列へ格納

mode()によって最頻値を取得
上のargmaxは番号で出るのでon-hotに

＋αなアンサンブル法

[24]:
```
print(ensembling_both(allmodels,x_val,y_val))
```

```
(0.79368, 0.78704)
```

[25]:
```
def ensembling_original(models, X, y):
    preds_sum = None
    for model in models:
        if preds_sum is None:
            preds_sum = model.predict(X)**2
        else:
            preds_sum += model.predict(X)**2
    probs = preds_sum / len(models)
    ens_y_pred = to_categorical(np.argmax(probs, axis=1), num_classes=10)
    ens_acc = accuracy_score(y, ens_y_pred)
    return ens_acc
```

[26]:
```
ensembling_original(allmodels,x_val,y_val)
```

[26]:
```
0.79248
```

[27]:
```
#ルートをとって平均
def ensembling_original2(models, X, y):
    preds_sum = None
    for model in models:
        if preds_sum is None:
            preds_sum = np.sqrt(model.predict(X))
        else:
            preds_sum += np.sqrt(model.predict(X))
    probs = preds_sum / len(models)
    ens_y_pred = to_categorical(np.argmax(probs, axis=1), num_classes=10)
    ens_acc = accuracy_score(y, ens_y_pred)
    return ens_acc
```

[28]:
```
ensembling_original2(allmodels,x_val,y_val)
```

[28]:
```
0.7964
```

Soft ensembleと同じような処理だが、
平均をとる場合は、各確率を足し合わせて個数で割るが、
足し合わせる前に
・２乗をする処理と、
・ルートをする処理(瀧先生提案)

結果は

soft, hard: (0.79368, 0.78704)
**2の平均: 0.79248
sqrtの平均 0.7964

単体性能
version: 6 0 best val score 0.74528
version: 6 1 best val score 0.73632
version: 6 2 best val score 0.7429600000000001
version: 6 3 best val score 0.7338399999999999
version: 6 4 best val score 0.74456

+αなアンサンブル法

[159]:
```
#変数
def ensembling_original3(u, models, X, y):
    preds_sum = None
    for model in models:
        if preds_sum is None:
            preds_sum = model.predict(X)**u
        else:
            preds_sum += model.predict(X)**u
    probs = preds_sum / len(models)
    ens_y_pred = to_categorical(np.argmax(probs, axis=1), num_classes=10)
    ens_acc = accuracy_score(y, ens_y_pred)
    return ens_acc
```

N乗の平均 (瀧研)
・2乗の平均に類似したコードであり、2乗(**2)していた箇所を n(ここではu)乗に変更(**u)することによって実装できた。 uがハイパーパラメータのような役割を持つ。
結果は、
soft, hard: (0.79136, 0.78276)
**0.0001の平均: 0.7924
**0.001の平均: 0.7924
**0.01の平均: 0.79248
**0.05の平均: 0.79264
**0.1の平均: 0.79272　←best
**0.2の平均: 0.79264
**0.3の平均: 0.79248
**0.5の平均: 0.792　→original2
**0.8の平均: 0.79152
**1の平均: 0.79136　→ソフトアンサンブル
**2の平均: 0.78992　→original1
**3の平均: 0.788
**4の平均: 0.78696
**5の平均: 0.78592

知識の蒸留

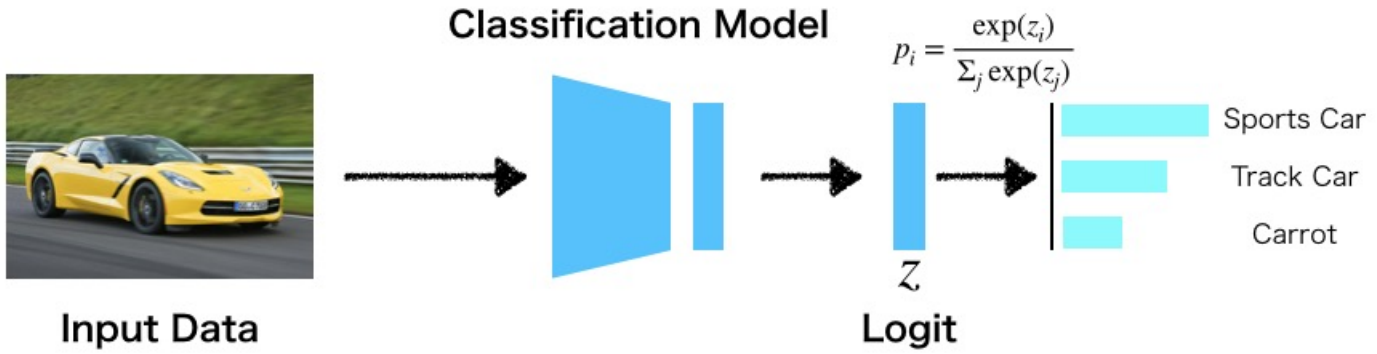
- Distilling Knowledge 知識の蒸留とは：
- 温度つきSoftmax cross-entropy による大規模モデルから小規模モデルへの知識の転移
- →巨大で複雑なモデルの知識を小さくシンプルなモデルへと転移させることで、小さいモデルで大きいモデルと同等の精度で推論を実行できるようにする。
- TeacherとStudentの関係性。

https://arxiv.org/pdf/1503.02531.pdf

知識の蒸留

- 通常の分類モデル
  - 対数尤度pをcross-entropyで最大化する。(下の式を最小化)

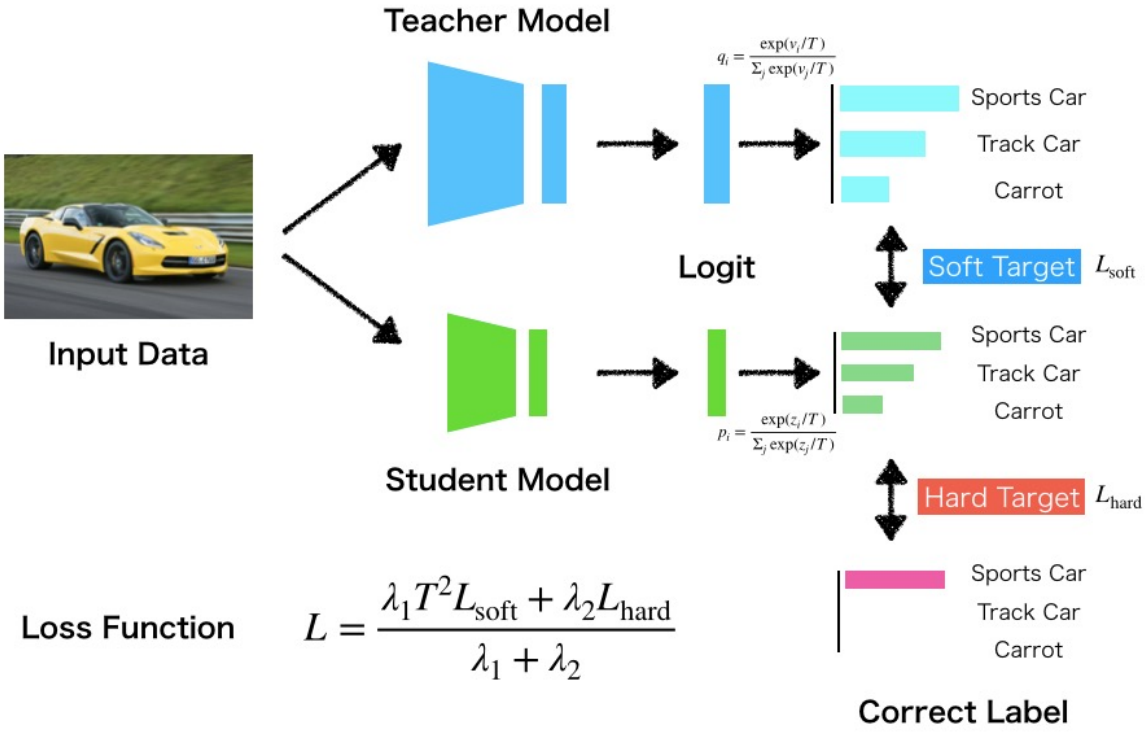
$$L = -\sum_{i=0}^{n} q_i \log p_i$$



解説記事より抜粋

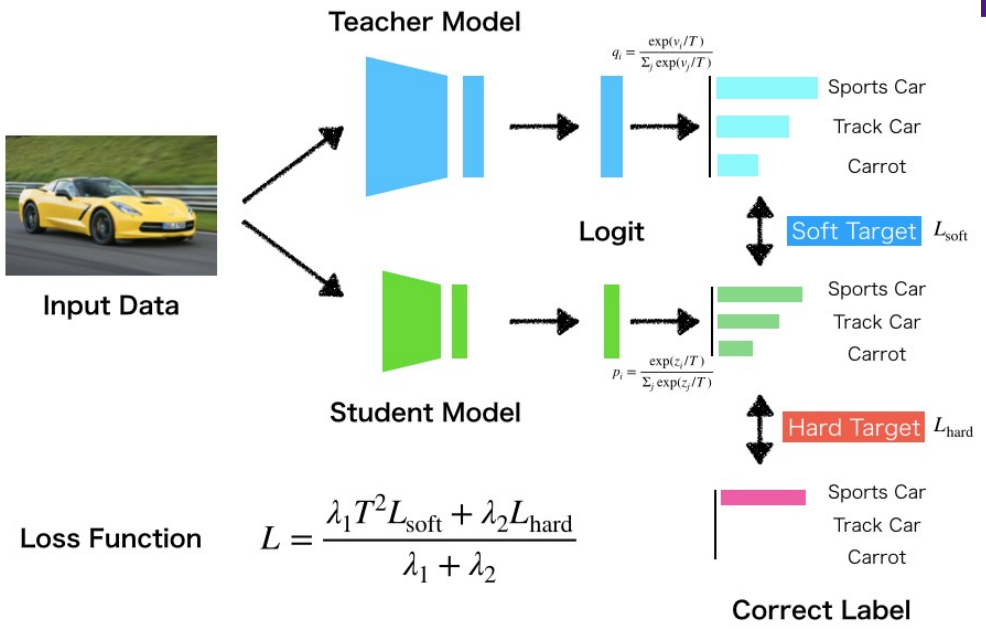
知識の蒸留

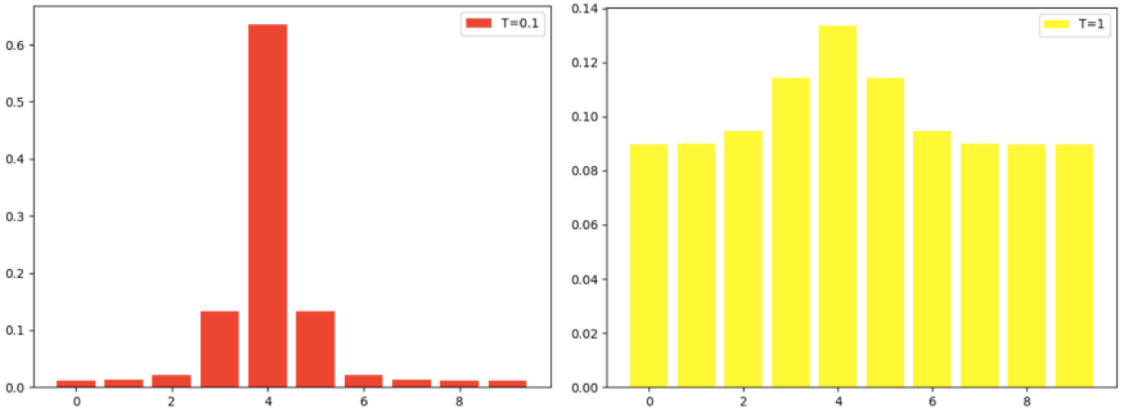
- 知識の蒸留：
- Teacherの出力（予測確率）を使ってStudentの学習を行う。



２つの損失項で最適化される。
Soft target: Teacherの予測確率と
Studentの予測確率を近づける項
Hard target:正解ラベルと予測確率を
近づける項(普通のクロスエントロピー)

知識の蒸留





特殊なパラメータ：温度T

Teacherから得られる予測確率は入力xの指数関数exp(x)から計算されるので正解ラベルについて高く、相対的に不正解ラベルについてはとても小さいものになる(確率の偏り)
なので温度付きsoftmaxを導入して温度パラメータTをT>1に設定することで不正解ラベルに対する予測確率を強調して学習する

知識の蒸留

- 元論文の精度(MNIST)

	# of layers	# of hidden units per layer	Test error cases
Teacher	2	1200	67
Student	2	800	146
Student (Distilled)	2	800	**74**

少ないパラメータで近い汎化性能が得られる

今後の展望

- 今後はよりフォーカスした研究活動を行なっていきたい。
  - 上記手法を組み合わせ、異文化コミュニケーション学の観点を取り入れた精度の向上するアルゴリズムの研究
  - 知識の蒸留と畳み込みフィルタの可視化を組み合わせて、文化の継承を解明、ブラックボックスを解明する研究を検討している。

Recent Works

カプセル型のモデル
- 複数の異なるモデルに寄生する形で学ぶモデル。
- CIFAR10での小規模モデルではaccuracyの上昇を確認。より深いモデルResNetでは学習を阻害。今後の改良を目指す。

プラスチックデータの分析
- 環境に優しく（環境分解性）、熱に強い（熱安定性）プラスチックの生成条件の探索
- 主成分分析などの前処理と、RandomForestやガウス過程回帰などの機械学習・ベイズ統計学を用いて予測モデル構築。

ベンチマークデータでの深層学習を用いた不良品検知
- AutoEncoderによる分布の整理を行うことで不良を検知しやすくする。
- 他の手法も要勉強。

立教大学
RIKKYO UNIVERSITY

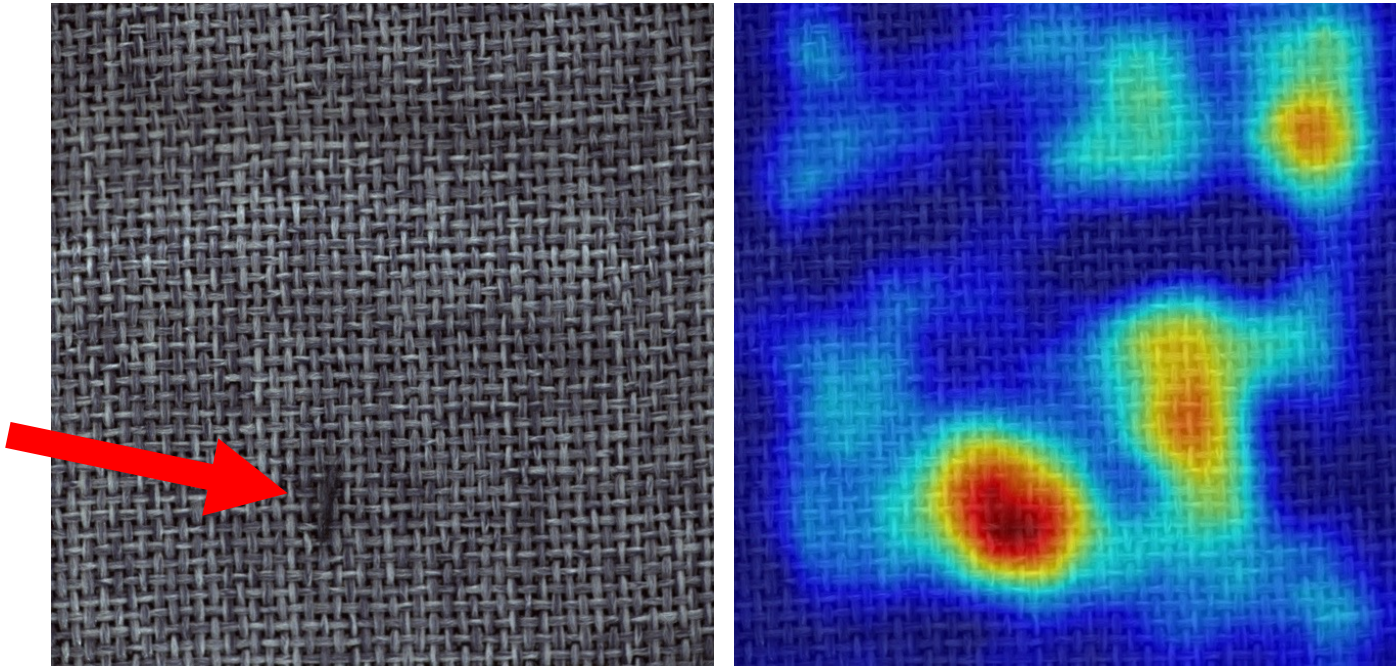
参考文献

『Distilling Knowledge in a Neural Network』

- 下記の解説記事を参考に
- https://paperdrip-dl.github.io/distillation/2018/12/23/Distillating-Knowledge-in-Neural-Networks.html
- 元論文
- https://arxiv.org/pdf/1503.02531.pdf

判断根拠の可視化

- 最後の畳み込み層の勾配を利用し、出力値に反応したピクセルをヒートマップで表示。

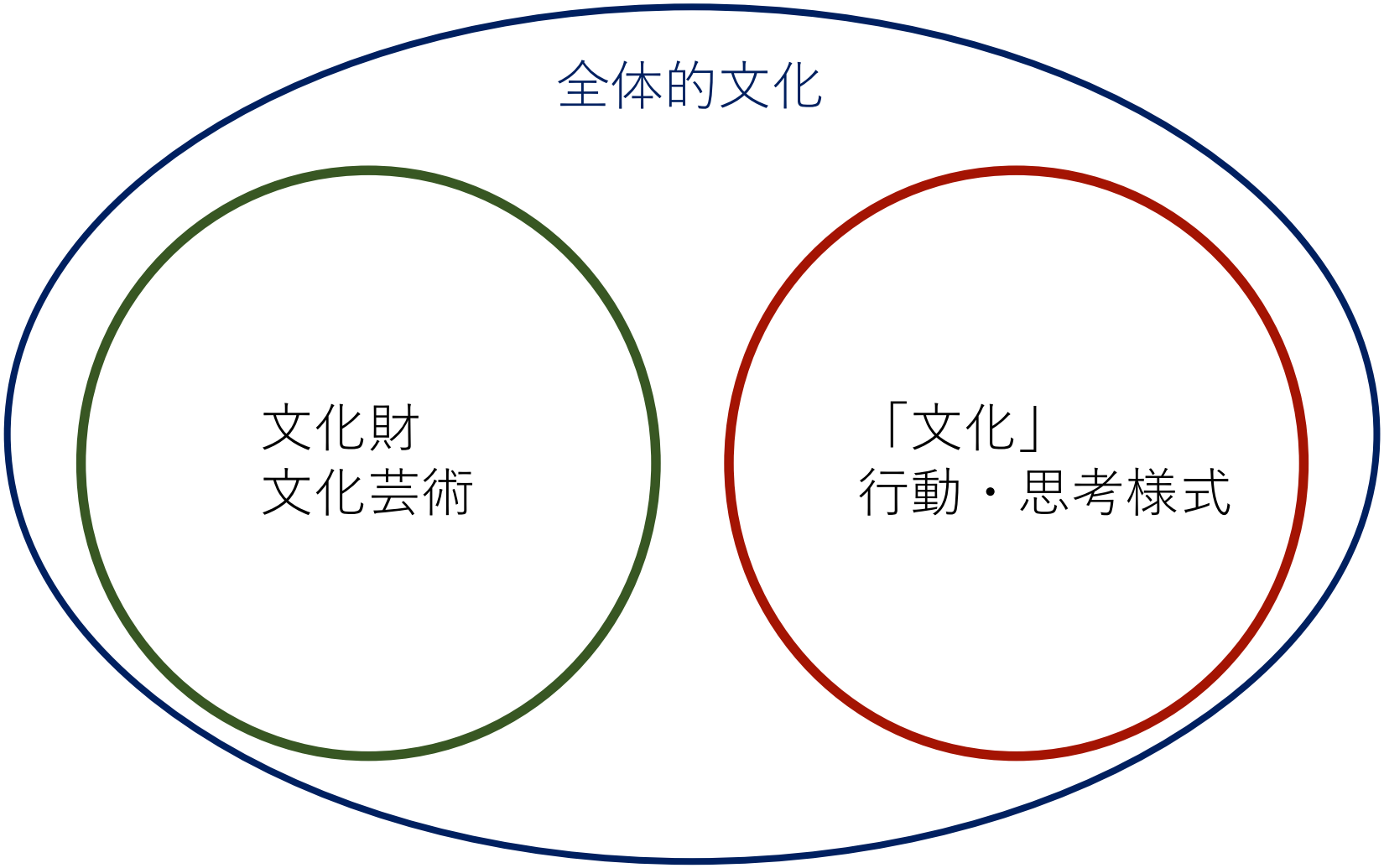


研究テーマ（定義）

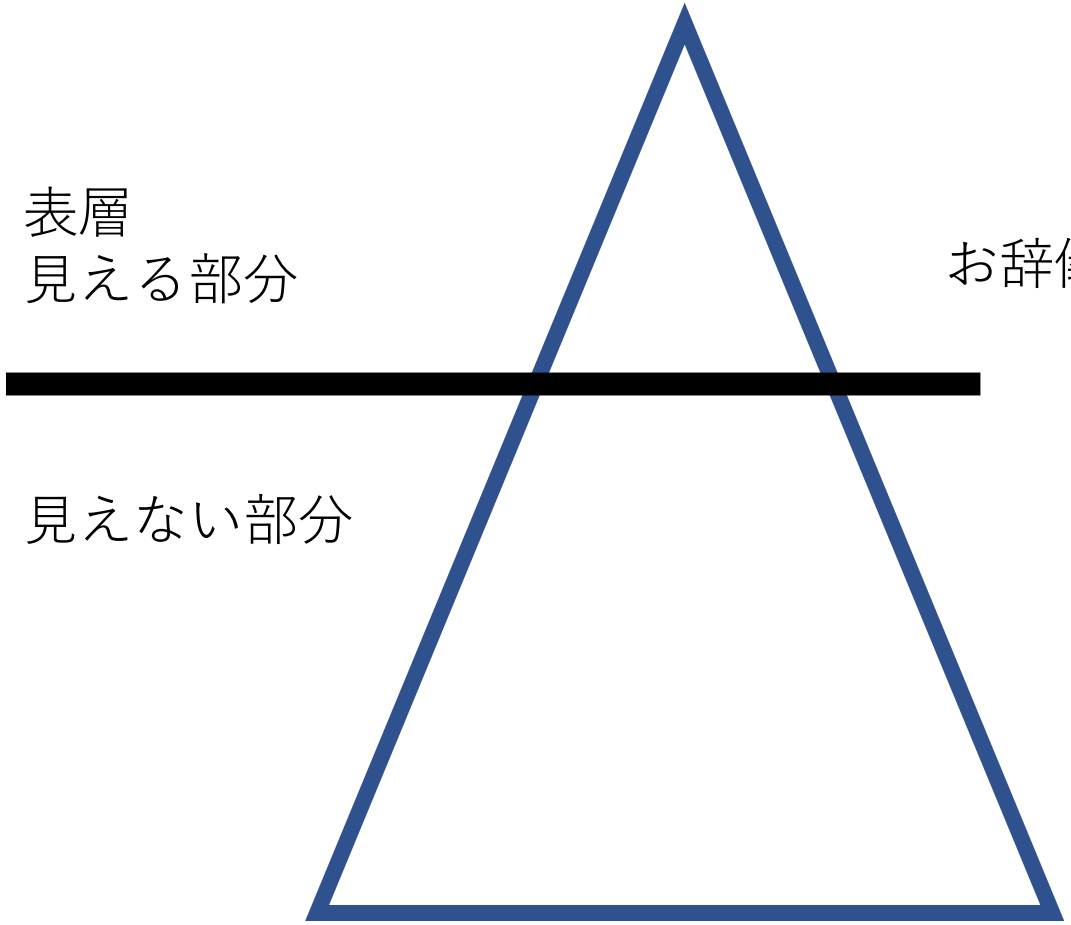
文化とは

「知能によって脅威に対抗し、生存を維持する手段として作り上げたもの」*1
（斗鬼、2003）＜文化人類学＞
「あるグループの中で教えられ、学ばれる、つまり共有される
すべてのもので、グループの中で世代から新世代へ継承されていく」*2
（小坂、2012）＜異文化コミュニケーション学＞

→集団の中で共有され、継承される**行動・思考様式**のこと。


全体的文化
文化財
文化芸術
「文化」
行動・思考様式

文化は氷山(Iceberg)に例えられる



表層
見える部分

お辞儀をする（動作）

お辞儀＝感謝（意味）

見えない部分

礼儀正しさが自分の身を守る

日本文化は非言語行動が重要視

Example)　お辞儀

道を譲ってもらった→お辞儀をする

＜知識＞
ルール(a=b)自体を指す
　・お辞儀＝感謝
　・道を譲ってもらった→感謝する＝お辞儀

＜文化＞
経験した文化を継承する
　・日本：道を譲ってもらった→お辞儀をする（非言語コミュニケーション）
　・アメリカ：　　→”Thank you”（言語コミュニケーション）

研究テーマ

＜知識ベース型AI＞

知識(ルール)×推論機構
→推論

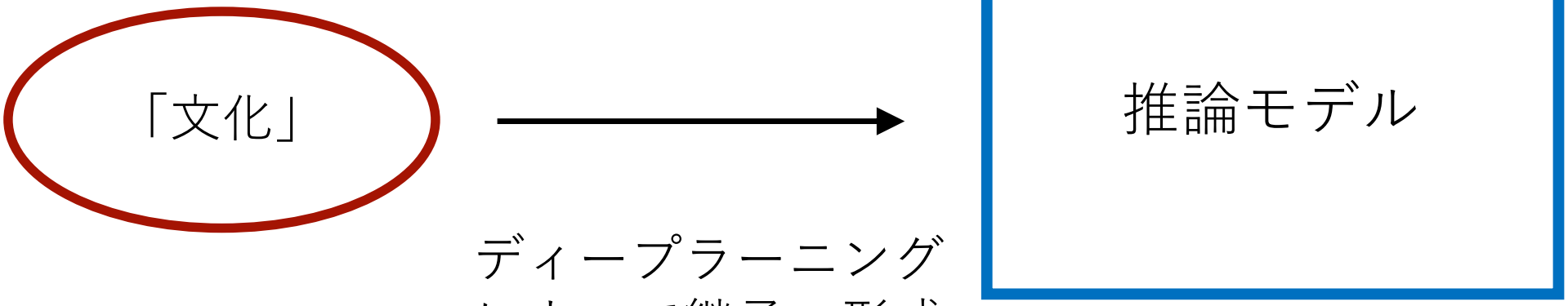
知識a=bをデータに当てはめる

＜ディープラーニング型AI＞

データの抽象的表現を学習
→推論モデルの数値パラメータで推論

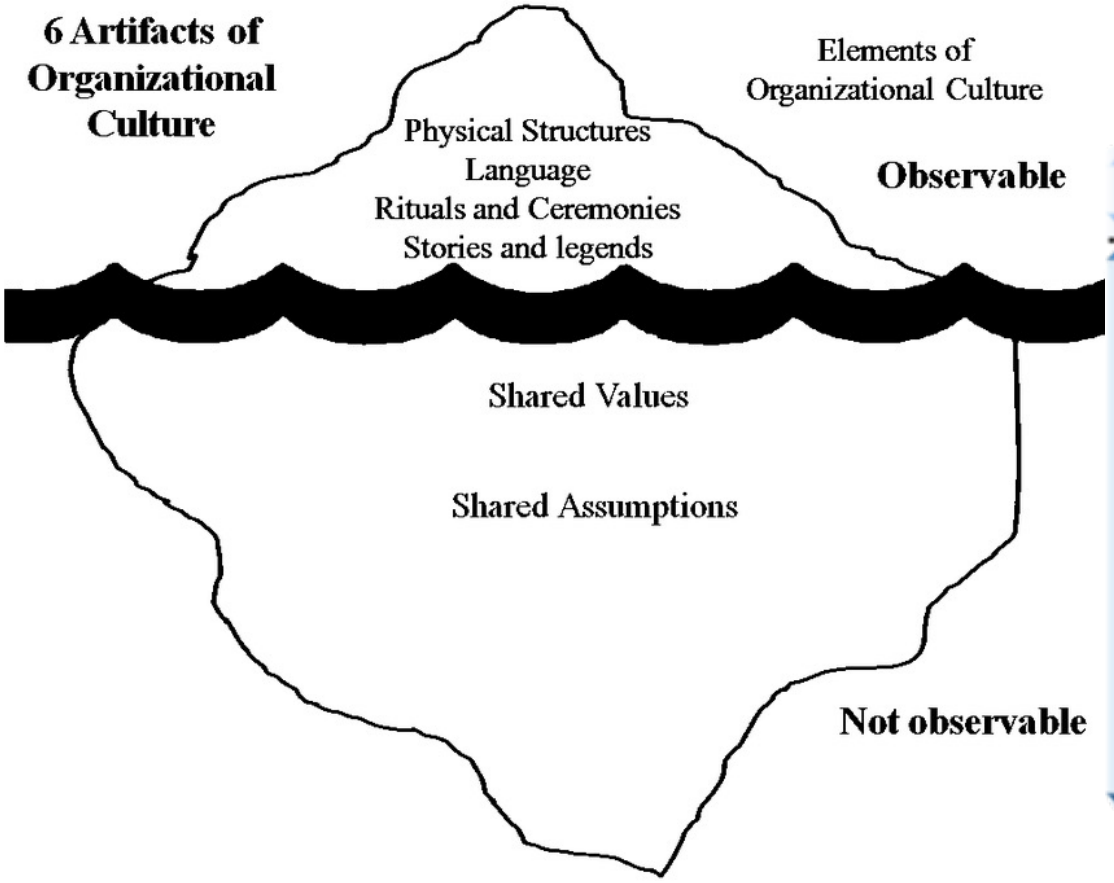
データを通してa=bの文化を継承する

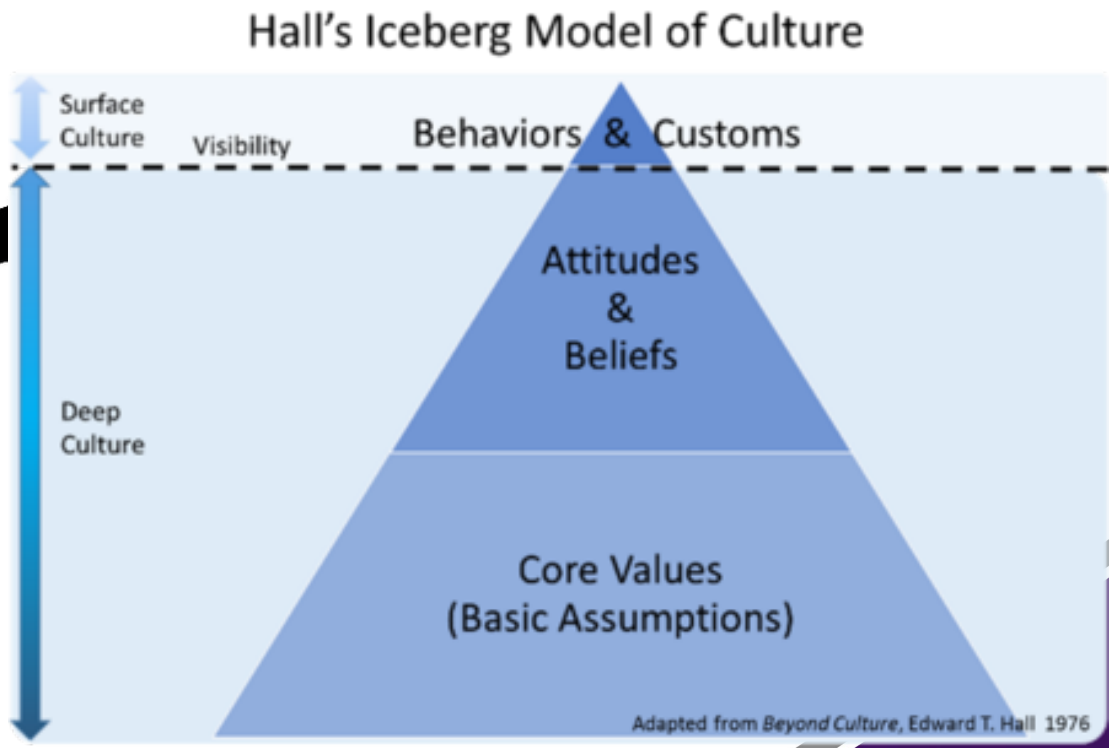
「a=b に至る思考様式」



引用文献

*1 斗鬼正一、2003、『目からウロコの文化人類学入門』、ミネルヴァ書房。
*2 小坂貴志、2012、『異文化対話論入門』、研究社。
*3 加藤泰、2001、『文化の想像力』、東海大学出版会。


6 Artifacts of Organizational Culture
Elements of Organizational Culture
Physical Structures
Language
Rituals and Ceremonies
Stories and legends
Observable
Shared Values
Shared Assumptions
Not observable


Hall's Iceberg Model of Culture
Surface Culture
Visibility
Behaviors & Customs
Deep Culture
Attitudes & Beliefs
Core Values (Basic Assumptions)
Adapted from Beyond Culture, Edward T. Hall 1976

論文紹介

『Ensemble Distribution Distillation』

- Hirono Okamoto, Matsuo Lab
- 概要：一般の蒸留は、複数モデルの予測の平均を一つのモデルの出力に近づけるように学習するため、それぞれのモデルの予測を学習しておらず、アンサンブルの多様性を失っている。
- 提案手法は複数モデルの予測の分布を一つのモデル(Prior Network)に蒸留することで、分類精度を一般の蒸留モデルと比べて落とすことなく、外れ値に対して頑健なモデルにする。

https://www.slideshare.net/DeepLearningJP2016/dlensemble-distribution-distillation-218038516

論文紹介

- 蒸留の目的：ネットワークを小さくして分類精度を落とさずに推論速度をあげること
- 複数のモデル(親モデル)で学習したモデルをある一つのモデル(子モデル)に蒸留することを考える
  - one-hotで学習するよりも情報量が多いため、予測性能が一般的に良くな

$$L(\emptyset, D_{ens}) = \mathbb{E}\left[KL\left[\mathbb{E}_{\hat{p}(\theta|D)}[P(y|x;\theta)] \| p(y|x;\theta)\right]\right]$$

アンサンブルによる予測　　新たに学習したいモデルの予測

それぞれの元のモデルの多様性を失っている