

脳波と心拍数を用いた 感情推定モデルの作成

システム工学部2回生 西浦 奏絵



1

目次

- 1, 研究の背景・目的
- 2, 到達目標
- 3, 研究方法
- 4, 研究結果
- 5, 今後の展望
- 6, まとめ

2

2

研究の背景・目的

感情を定量化する試み

表情や音声

中枢神経系の変化

自律神経系の変化

スタンフォード大学wangの提言

3

3

研究の背景・目的

感情を定量化する試み

表情や音声

脳波の変化

心拍数の変化

スタンフォード大学wangの提言

4

4

到達目標

脳波と心拍数を用いた
個人に応じた感情推定モデルの作成

5

5

研究方法

1, データの取得

脳波: 脳活動の際に発生する微弱な電気信号

→ MindWave Mobile2で測定

心拍: 一定の時間内に心臓が拍動する回数

→ fitbit versa4で測定



簡易脳波計 MindWave Mobile2



心拍計 Fitbit versa4

6

6

研究方法

1, データの取得

最終的に分類したい状態は4つ

- ・面白い: 漫才を見る
- ・嬉しい: スマホゲームで対戦相手に勝つ
- ・悲しい: 映画を見る
- ・平常: その他

上の3状態に関して5分から10分ほどのデータを取得する実験を5セット行った

7

7

研究方法

2, データの処理: 脳波

デバイスで計測した脳波を専用アプリケーションを通してRawデータとして μV の単位に変換し出力



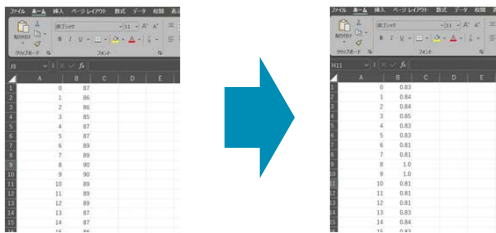
8

8

研究方法

2, データの処理: 心拍

デバイスで取得した心拍数に対して、最大心拍数の80%の値を閾値として、0.0~1.0の値に正規化を行う



9

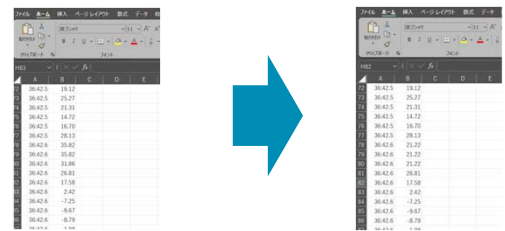
9

研究方法

2, データの処理: ノイズ

脳波には筋電、呼吸などによるノイズが入る

- 最大数値の80%を超える数値があった場合、
超える前(5秒間)の平均で統一する

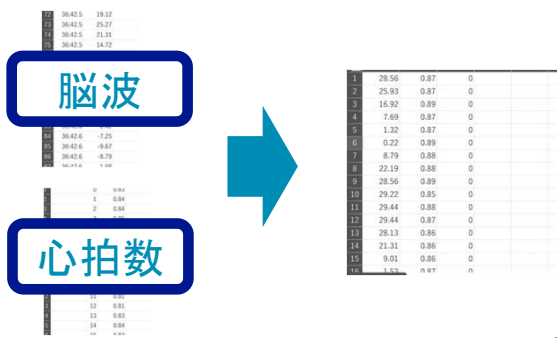


10

10

研究方法

2, データの処理: 結合

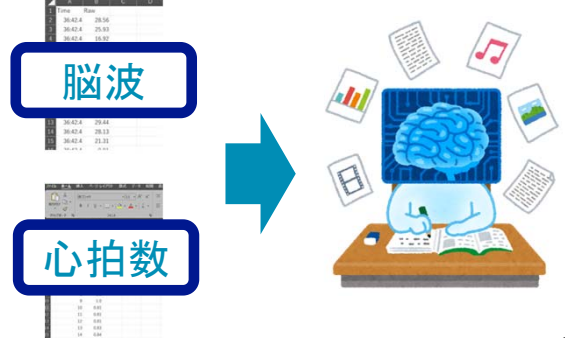


11

11

研究方法

3, ニューラルネットワークでのモデル化



12

12

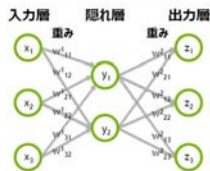
研究方法

* ニューラルネットワークとは

→ 生物の神経細胞の構造を模倣して作られた**機械学習モデル**

多層パーセプトロン(3層)

ニューラルネットワークの構造



- * 入力層
外部からのデータを受け取る層
- * 隠れ層
データの非線形な特徴や関係性を学習するための層
- * 出力層
最終的な出力を生成する層

引用: ビジネス+IT「ニューラルネットワークの基礎解説」
URL: <https://www.ebit.jp/article/cont1/33345a>

13

13

研究方法

3, ニューラルネットワークでのモデル化

・全データ数: 約12000

→ 各感情ごとに5分~10分のデータ取得 × 5回

・分類数: 4

→ 「平常時」: 0, 「面白い」: 1, 「嬉しい」: 2, 「悲しい」: 3

・学習データを80%、評価データを20%

14

14

研究方法

3, ニューラルネットワークでのモデル化

```
import numpy as np
import tensorflow as tf
from tensorflow.keras import layers, models

# データの読み込み
raw_data = np.loadtxt('data.txt', delimiter=',')
raw_data = raw_data / 10000.0

# データの分割
train_data = raw_data[:12000]
test_data = raw_data[12000:]

# ニューラルネットワークの構築
model = models.Sequential([
    layers.Dense(10, activation='relu'),
    layers.Dense(10, activation='relu'),
    layers.Dense(4, activation='softmax')
])

# 学習
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
model.fit(train_data, test_data, epochs=100, validation_data=test_data)
```

```
def sigmoid(x):
    return 1 / (1 + np.exp(-x))

def softmax(x):
    e_x = np.exp(x)
    return e_x / np.sum(e_x, axis=-1)

def predict(x):
    h = np.dot(x, w1)
    h = sigmoid(h)
    z = np.dot(h, w2)
    return softmax(z)
```

15

15

研究方法

3, ニューラルネットワークでのモデル化

	A	B	C	D	E	F
1	Raw1	Raw2	result	predict		
2		3.74	0.87	0	0	
3		8.13	0.80	0	2	
4		18.68	0.91	1	1	
5		30.54	0.84	3	3	
6		35.60	0.87	2	0	
7		29.66	0.89	0	0	
8		18.90	0.88	0	2	
9		8.13	0.88	0	0	
10		-2.20	0.89	3	3	
11		-12.52	0.97	3	3	
12		-15.16	0.79	0	1	
13		-7.47	0.91	0	0	
14		3.52	0.88	2	0	
15		13.18	0.81	1	1	
16	raw op	noc	1	1		

正解率: 70~80%ほど
→ 想定よりも高い結果に

16

16

研究方法

3, ニューラルネットワークでのモデル化

* 良かった点

- ・想定よりも高い結果が得られた
- ・平常時でない場合のデータが正しく分類できていた

* 改善点

- ・平常時の場合に誤って分類されることが多い
 - データ数の多さが問題?
 - データ数の調整など

17

17

研究結果

脳波・心拍数のデータ取得

取得したデータの処理

ニューラルネットワークへの適応

感情分類モデルの作成

18

18

課題

- 今回作成したモデルを用いて、実際に実験をすることが出来なかった
 - 新たなデータで感情分類を行う予定
- 汎用性のモデルを作成することが出来なかった
 - 脳波にも閾値を設けて実験を行う予定
- リアルタイムでの感情分類が出来なかった
 - プログラムなどの改善を試みる予定

19

19

今後の展望

* 今回作成した感情推定モデル、プログラムなどを用いて
自主研究を予定

- 今回のデータは無意識化での感情
 - 意識して感情を持つ(セルフマインドコントロール)とどうなる？
- 認知バイアスと感情の関係性
- 意識はプログラムで模倣できるか？

20

20

まとめ

- * 脳波と心拍数を用いて感情を推定するモデルを作成した
 - 正解率が高めのモデルを作成出来た
 - 課題点が多いため、改善していく
 - 汎用モデルの作成
 - リアルタイムでの測定
- * モデルを使用して新たに自主研究を行う予定

21

21