

令和3年度 卒業研究

安眠用 BGM を自動選択する研究

－ 安眠用 BGM を自動選択するモデル開発 －

Study to automatically select background music for restful sleep

--- Development of a model to automatically select BGM for restful sleep ---

函館工業高等専門学校

生産システム工学科 情報コース

5年 中村 碧

第 1 章 序論

第1節 英文アブストラクト

The goal of this research is to develop a model for AI to automatically select BGM for sleep. Eventually, we would like to develop an application that outputs appropriate music based on the biometric information (heart rate, sleep quality) given by the user. This is because many people do not sleep well with existing sleep BGM, and there are few applications that contribute to reducing sleep disorders using biometric data. In this study, we used a three-layer neural network with FFNN to train the system. As a result, although the training was successful, the number of data was small and the test data did not provide satisfactory results. As a future prospect, I thought it is necessary to collect a lot of data and train it, and to have a system that can acquire data in real time.

第 2 節 研究背景

近年、スマートウォッチなど個人が利用できるウェアラブルデバイスの発達によって生体データの収集が個人で容易にできる時代になった。この影響で生体データを利用した個人向けのサービスやソフトウェアが開発されている。

特に、医療の分野におけるウェアラブルデバイスの活用はすでに多く行われており、例えば、患者の生体データを日常的にあつめることによって通常の診察・検査でわかる情報に大きく左右されず、高い精度で診察できるシステムの開発が進んでいる。

第 3 節 研究目的

睡眠時の生体データを活用し、睡眠の質を上げようとする無料のソフトウェアの数は少ない。そこで本研究はダウンロードされた音楽の中から睡眠時の心拍数、睡眠の質に合わせた音楽を選択する人工知能のモデル開発を目的とする。

第 4 節 開発環境

本研究での開発環境は以下の通りである。

第 1 項 依存ライブラリ

keras 2.7.0 [1]

numpy 1.21.5 [2]

Google Fit [3]

第2項 開発言語

Python 3.7.12

第3項 コンピュータ とウェアラブルデバイス

OS: MacOS

ウェアラブルデバイス : Mi smart band 6 [4]

第2章 関連技術

第1節 3層全結合型 Feedforward Neural Network (FFNN)

FFNN[5]は脳の神経細胞を数学的にモデル化したパーセプトロンを多数結合した多層パーセプトロン (MLP) の構造のことで再帰構造を持たない。今回使用した FFNN は隠れ層を1とした3層ニューラルネットワークである。

第2節 Mi Smart Band 6

本研究では中国のデバイスメーカーXiaomiが発売しているウェアラブルデバイスをデータの収集を目的として利用している。

価格は5980円で、付属センサーとして心拍センサー以外に高精度6軸センサー（低消費電力の3軸加速度計と3軸ジャイロスコープ PPG）を内包している。また、睡眠モニタリング機能もあり、深い睡眠、浅い睡眠、REM (Rapid Eye Movement) 睡眠を観測することができる。バッテリーは14日間持つように設計されており、1日中使っていても問題ない。

第3節 Google Fit

値段と機能の面で優れている Mi smart band 6 だが SDK や API が整備されておらず、データストアから生体データを取得するのは難しい。したがって、端末が取得する生体データをストックできる Google Fit を使用することとした。ウェアラブルデバイスから得られた様々な生体データは Google Fitness Store に格納され、参照することができる。

Google Fit は生体データをデータストアに同期するのは時間がかかるものの、1日たてば csv ファイルとして生体データをエクスポートできるので今回の生体データはそれらを参照して使用している。

第4節 TensorFlow

TensorFlow は Google が開発した機械学習専用のソフトウェアライブラリである。現在では様々な分野で利用されており、本研究でもこちらのライブラリを使用している。

テンソルとは多次元配列のことで、1次元配列であるスカラーや2次元配列であるベクトル、さらに3次元配列のことを指す。これらの配列の概念をデータに対して扱えるようにするのがテンソルである。

第5節 Keras

Keras は前節で記載した TensorFlow をバックエンドとして動作する高レベル API である。そのため、Keras は非常に使いやすく、構造もシンプルであり、開発の際にドキュメントを調べる時間が短縮できる点が非常に便利である。本研究では Keras を使って深層学習を行っている。

第3章 モデルの開発

第1節 入力データ

本研究では入力データとして心拍数（最小、最大、平均）睡眠の質（REM（Rapid Eye Movement）睡眠、深い睡眠、浅い睡眠）を選択した。

第2節 データのラベル付け

曲の終了時の平均心拍数が開始時の平均心拍数より低ければ寝やすい曲と判断し、高ければ寝にくい曲と判断するようにした。そして、寝やすい曲と判断した場合を1で表し、寝にくい曲と判断した場合は0とした。

本研究で使用した曲は8曲であるため、ラベルとして8ビットのバイナリデータとすることにした。例えば、3曲目の曲が寝やすい曲と判断された場合、00100000がラベルとなる。

第3節 モデルの構築

モデルの開発を進めるにあたって図1のモデルを構築した。

入力層のユニット数は、心拍数（平均、最大、最小）と睡眠の質（REM、深い睡眠、浅い睡眠）の合計7個とした。

次に、隠れ層（中間層）には \tanh (双曲線)活性化関数を利用し、ユニット数は110個とした。

そして、出力層は Softmax 活性化関数を利用した。選択する音楽は8曲であり、その8曲の中からより良い睡眠を提供できる音楽を選ぶため、出力層のユニット数を8個とした。

また、入力層、隠れ層、出力層の3つだけでは学習結果が安定しないことが予想できたため、安定化を図るために隠れ層と出力層の間に Dropout 層[6]を追加した。本研究では Dropout (0.25) として学習を意図的に遅らせた。

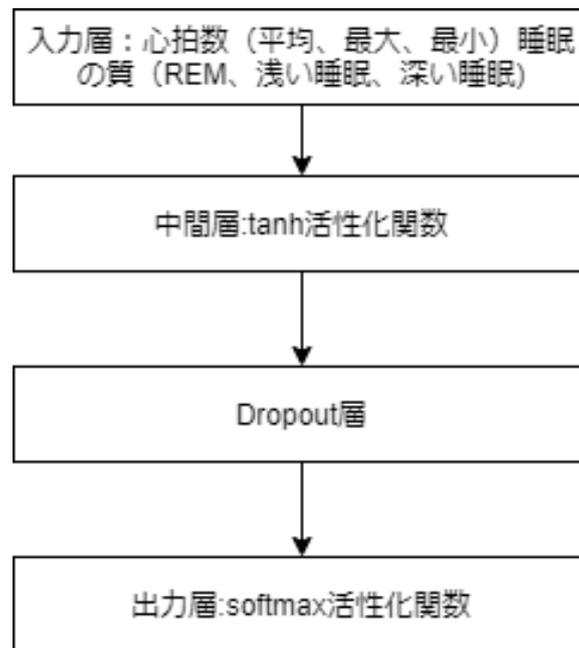


図1 モデル構築図

第4章 評価実験

第1節 データセット

私個人の心拍数と睡眠のデータを使用して評価実験を行った。学習データセットの数は9個、テストデータセットの数は4個である。

第2節 学習データセットの結果

学習データセットに対して学習率 0.001、エポック数 500、バッチサイズ 2 として学習を行った。その結果、学習前の正解率は 11%であったのに対し、学習後の正解率は 100%となった(図 2)。

学習前

学習データセット : 損失=2.12, 正解率=0.11

```
[[ 0.053 0.366 0.153 0.111 0.110 0.140 0.007 0.059]
 [ 0.057 0.176 0.191 0.117 0.078 0.271 0.039 0.072]
 [ 0.024 0.507 0.190 0.078 0.041 0.099 0.014 0.049]
 [ 0.044 0.114 0.236 0.165 0.089 0.196 0.079 0.077]
 [ 0.050 0.393 0.103 0.125 0.093 0.164 0.008 0.066]
 [ 0.041 0.365 0.221 0.114 0.085 0.104 0.010 0.060]
 [ 0.036 0.110 0.275 0.118 0.077 0.246 0.070 0.068]
 [ 0.018 0.548 0.153 0.083 0.045 0.099 0.013 0.041]
 [ 0.110 0.116 0.108 0.276 0.125 0.118 0.031 0.116]]
```

学習後

学習データセット : 損失=0.00, 正解率=1.00

```
[[ 0.999 0.000 0.000 0.000 0.001 0.000 0.000 0.000]
 [ 0.000 0.999 0.000 0.000 0.000 0.000 0.000 0.000]
 [ 0.000 0.000 1.000 0.000 0.000 0.000 0.000 0.000]
 [ 0.000 0.001 0.000 0.999 0.000 0.000 0.000 0.000]
 [ 0.001 0.000 0.000 0.000 0.999 0.000 0.000 0.000]
 [ 0.999 0.000 0.000 0.000 0.000 0.000 0.000 0.000]
 [ 0.000 0.999 0.000 0.001 0.000 0.000 0.000 0.000]
 [ 0.000 0.000 1.000 0.000 0.000 0.000 0.000 0.000]
 [ 0.000 0.000 0.000 1.000 0.000 0.000 0.000 0.000]]
```

図 2 学習結果

第2節 テストデータセットの結果

学習後、1日分の睡眠の生体データを使ってモデルのテストを行った(図3)。結果は正解率25%、損失5.73となり、期待される成果は得られなかった。

テスト

テストデータセット : 損失=5.73, 正解率=0.25

```
[[ 0.930  0.002  0.009  0.014  0.045  0.000  0.000  0.000]
 [ 0.060  0.134  0.405  0.400  0.000  0.000  0.000  0.000]
 [ 0.497  0.044  0.009  0.000  0.450  0.000  0.000  0.000]
 [ 0.725  0.133  0.141  0.000  0.000  0.000  0.000  0.000]]
```

図3 テスト結果

第5章 考察

今回の評価実験で学習したモデルの正解率は25%になった。正解率が低かった理由としてそもそもの学習データとテストデータセットの数が少なすぎることが挙げられる。今回は9個の学習データセットで学習したが、満足いく結果ではなかったため、より多い数の学習データ数が必要であるとわかった。

また多くの人からサンプルとしてデータをとる必要もある。今回の評価実験では、私個人の睡眠時の生体データが主であったため、本当の意味で個人にパーソナライズされたモデルとなってしまった。汎用性を高めるためにも、より多くの人数の睡眠時の生体データを取得する必要がある。

まとめとして、モデル構築は成功したが、学習データ数と学習データ数の元となる人の数がたりなかったため、想定していたような結果には至らなかった。

第6章 今後の課題と展望

モデルで高い精度を出すためには大量のデータが必要であるが、今回の評価実験では学習データの数が根本的に足りていなかったため、これを数多く集める必要があると思われる。

ただし手書き文字などであれば EMNIST などの学習データセットが公開されているが、生体データは個人情報であり非常にセンシティブなデータであるため、学習データセットが公開されるとは思えない。したがって、学習データをより効率よく集める仕組みが必要だと考える。例えば、睡眠時間分の分間心拍数を集めるプログラムを開発すれば、心拍数のデータに関してだけある程度の数を見込むことができる。

また、今後の展望として、BGM の自動選択を行うためにリアルタイムで心拍数と睡眠の質を取得したいと考えている。現在、データストアとして Google Fit を用いているが、データの同期に時間がかかってしまうため、リアルタイムで生体データを取得することが出来ない。したがって、今後はリアルタイムで生体データを取得できる API が公開されているウェアラブルデバイスを購入するか、iPhone の Health care を用いてデータ取得を試みたい。

また3層 FFNN に層を追加する、RNN を使うなどの様々な試みを行うことでテストデータの正解率を上げることができると考える。

参考文献

- [1] Keras : Documentation <https://keras.io/ja/> (最終参考 2022 年 2 月)
- [2] NumPy: <https://numpy.org/> (最終参考 2022 年 2 月)
- [3] Google Fit: https://www.google.com/intl/ja_jp/fit/ (最終参考 2022 年 2 月)
- [4] Mi smart band 6: <https://www.mi.com/jp/product/mi-smart-band-6/specs> (最終参考 2022 年 2 月)
- [5] Feedforward neural network: https://en.wikipedia.org/wiki/Feedforward_neural_network (最終参考 2022 年 2 月)
- [6] Core レイヤー – Keras Documentation : <https://keras.io/ja/layers/core/> (最終参考 2022 年 2 月)