

Raspberry Piによる時系列入力判別

横山 大作^{1,a)}

概要：Raspberry Pi を用いて、リズムに合わせたスイッチの操作入力記憶しているものが否かを判定するシステムを構築した。入力はプッシュボタンが押されている間のみ ON となるような時系列であり、これに移動時間窓を適用して特徴量化を行い、ガウシアンカーネルを用いた SVM で判別を行う。設定時に何度か操作入力を行ったものを正例とし、正例をランダムに変更することで負例を作成して学習を行った。システムは学習、判定いずれの場合も Raspberry Pi だけで十分な速度で動作し、ある程度の精度で入力を判別できることが確認できた。また、このようなシステムが python と libSVM を用いて極めて簡単に構築できることも確認された。

キーワード：Raspberry Pi, 時系列信号、判別器

1. はじめに

2015 年夏のプログラミング・シンポジウムのお題のひとつは「下呂でしかできないプログラミング」であった。下呂に至る道すがら、流れ去ってゆく飛騨川の水面を眺めたことを思い出しつつ、もうひとつのお題である Raspberry Pi を触っていると、流れてゆく時系列信号を処理しよう、というテーマが浮かび上がった。著者は Raspberry Pi に触れることは初めてであったが、LED とスイッチの入出力が非常に容易に利用できることから、LED2 つとスイッチ 1 つのみという最小限のインタフェース構成で、何らかのアプリケーションを作ろうと考えた。

さて、現在著者は 2 歳の娘の子育て中であるが、2 歳児はとにかく常に遊びを欲している。お気に入りのおもちゃは数日から 1 週間程度でどんどん変化するが、最近気に入っていたものに、LED が

光るペンダントがあった。押しボタンスイッチを押すと LED が決まったパターンで点滅するだけのごく単純なものであるが、幼児はことごとくこのような光のおもちゃに釘付けになるようであった。もし、スイッチを押した時の反応が様々に変化するようなおもちゃがあれば、幼児は飽きることなく遊びを続けることができ、両親は平穏な生活を取り戻し、ひいては社会平和につながるのではないかと考えた。

そこで、リズムカルにスイッチを押すと、LED が何らかの反応を返すようなアプリケーションを構築することを最終目標とし、その準備として、時系列入力のパターンを、既知のパターンと同一かどうか判別できるような機能を作成することにした。

2. 設計

Raspberry Pi の GPIO を利用して LED の出力を 2 つ、押しボタンスイッチの入力を 1 つ用意する。プログラミングには python を使い、RPi.GPIO ラ

¹ 東京大学

^{a)} yokoyama@tkl.iis.u-tokyo.ac.jp

イブラリ^{*1}を利用して入出力をコントロールする。既知のパターンとの判別はSVMの定番ライブラリであるlibsvm^{*2}を利用する。

2.1 インタフェース

LEDの1つは入力ガイドのために動作中に一定のビートで点滅することとした。4拍子のビートを表すため、LEDは4回に1回だけ高輝度に光るよう、LEDの点灯時間の長さを調節した。ユーザはビートを参考にしながら、4拍子1小節分の好きなリズム(「トントントン」「トトントントン」など)を入力することができる。

もう1つのLEDは、判別結果を表示するために用いる。

2.2 既知時系列パターンの判別

リズムの入力は、4拍子分の時間を32等分し、32個の時点でのスイッチのON、OFF状態を{1, 0}で表現する。既知のリズムとの同一判定には、SVMを用いることにする。32個の{1, 0}の入力をそのまま利用すると時間的なぶれに弱いことが予想されるため、大きさ4の時間窓を設定し、時間窓内に入った1の数を数えて1つの素性とし、時間窓を移動させて4拍子分全体の特徴量とすることにした。

学習時には、ユーザに同じリズムで10回程度入力を行ってもらい、これを正例とした。SVMを利用するため、負例を用意する必要があるが、今回は正例をランダムに変更して負例として与えることにした。変更は、

- 0.5拍程度のランダムな範囲を2つ選び、値をswapする
- ランダムな箇所を1にする

の2つを適用した。

SVMの学習はRaspberry Pi上で行った。学習データ量も特徴量次元数ともに小さなものであり、ガウシアンカーネルを利用してもRaspberry Piで充分高速に学習が行えた。

判定時には、4拍ごとに入力を特徴量化し、学

習した判定器を用いて既知リズムか否かの判定を行った。学習時と同様、特段の工夫をせずとも遅延を感じるほどの処理時間はかからなかった。

3. 考察

このようなシステムが、pythonとlibsvmを利用して極めて容易に実現できることが確認できた。

実用的に問題となったのは、利用した押ボタンスイッチが小さく、またブレッドボードに挿すだけで固定が不十分なため、リズムカルな入力を行っていくことであった。ユーザインタフェースを考えるとときに、ハードウェア的な機構の使いやすさは大切であるという、極めて当然の事柄に改めて気付かされた。今回の用途では、押ボタンスイッチではなく、加速度センサを利用するという改善方法が考えられるが、その場合には素性の作成方法がより複雑になると予想される。

今後の課題として、入力タイミングのLED表示をなくし、入力リズムの制限をなくすことが考えられる。この場合、過去の数秒間の入力を常時判定し、既知パターンを検出する手法が考えられる。既知パターンが入力されている以外の時間領域を無視するなどの工夫が必要になると予想される。

今回作成した判定器は、個人認証などの用途にも利用可能である。しかし、本来の目的であった、人間が飽きない応答を返すためには、異なったアプローチを考えることも必要そうである。機械学習のアプローチでは、人間らしい応答を大量に収集する必要があるが、音楽や日常会話の録音などからそのような典型的応答のリズムが抽出できるか、興味深い研究分野ではないだろうか。

4. 終わりに

子供の成長はすさまじい。LEDが光るだけで喜んでいてる時期はすぐに過ぎ去り、近頃は頭にあんこの詰まったキャラクターの商品をせがむようになっていく。果たして、入力に対してどのような応答を返せば飽きないのか、発達に応じた好みの変化はあるのだろうか。そんなことを思いながら東京に戻った私が発見したのは、無残に破壊され、打ち捨てられているLEDペンダントなのであった。

^{*1} <https://pypi.python.org/pypi/RPi.GPIO>

^{*2} <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>