

# 自作AIステップ1…最小構成ニューラル・ネットワークを作る

ご購入はこちら

牧野 浩二, 西崎 博光

いよいよChainerを使ってプログラミングの練習をします。いきなり難しいことを行うと分かりにくくなりますので、図1のように階段を上るイメージで少しずつステップ・アップしていきましょう。

最初はできるだけ簡単なニューラル・ネットワークでイメージを固めましょう。例題は何でもよいのですが、ひとまずここでは、2入力1出力のAND回路をChainerで作ってみましょう。

## プログラムを動かしてみる

### ● プログラムは提供します

ここではand.py(リスト1, 次頁)というファイルを用います。以下のサンプルを使います。本誌ウェブ・ページからダウンロードできます。実行するには、以下のコマンドを打ち込みます。

```
$ python and.py
```

実行が始まるとリスト2のように表示され、左端の数字が2000になると終わります。この例では約43秒でした注1。1行目に英語が並んでいます。その意味を表1に示します。

特に重要なのがテスト・データを入れたときの正答率です。この例では最初は0.75(75%)となっています。そして、終了時は1(100%)となっています。最後の4行は結果を分かりやすくするために示しています。

ただし初期値はランダムに決まっていますので、2000回で必ずしも100%にはならない場合があります。その際にはエポック数を大きくして再実行してみてください。

### ● 改造したいこと(つまづきポイント)

うまく学習できて答えが出ました。しかし、皆さんが実際に使うには、自分の条件に合わせてチューニングが必要ですよ。例として以下をやってみます。

注1: Windows+VirtualBoxで実行。CPU: Core i7, メモリは4Gバイト。

ディープ・ラーニング・プログラミング・ワールド

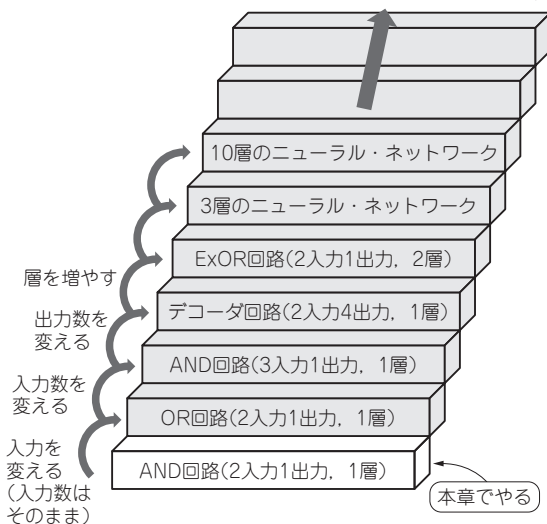


図1 ディープ・ラーニング・プログラミング・ワールドへの階段  
本章ではまず1段目を解説

- AND回路ではなく他の違う学習データを用いてニューラル・ネットワークを使いたい
- 学習データをファイルから読み込めるようにしたい
- 学習済みのニューラル・ネットワークに新しいデータを入力して結果がどのようなになるのか見たい。

実は、ここが結構大変なのです。上記を行うには、ある程度プログラムの構造を理解しておく必要があります。多くの本ではディープ・ニューラル・ネットワークの構造の理解のために、原理をしっかりと説明していますが、皆さんが改造するための方法を詳しく解説しているものは少ないようです。

特集では、この「改造する部分」を作れるように説明をしていきたいと思います。プログラムの説明は急に難しく感じるかもしれませんが、頑張りましょう。頑張りましょうと言っておいてなんですが、難しくてよく分からないと、読んでいても面白くないでしょうから、プログラムの説明を飛ばして使い方から学んで