

4D-3

ネットワーク構造の記述が容易な

汎用ニューラルネットワークシミュレータ

齋藤 真紀 西川 克彦 原田 裕明
富士通研究所

1. はじめに

現在、パターン認識にニューラルネットワークを用いる試みが盛んに行われているが、これらの研究にはニューラルネットワークシミュレータが不可欠である。パターン認識に用いるニューラルネットワークシミュレータには次のようなことが望まれる。

- (1)複雑なネットワークを構築するため、ネットワーク構造の変更が容易。
- (2)学習法の試行錯誤が容易。
- (3)認識結果等動作状況の分かり易い表示。

そこで筆者らは、これらの事項を満たすため、対話処理を中心とし、ネットワークの構築等を容易に行うことができるシミュレータを開発した。これについて報告する。

2. シミュレータの特徴

- (1)複雑なネットワークの構築を、対話的にかつ容易に行うため、簡単なネットワーク記述言語を設計しインプリメントした。
- (2)学習法の追加や将来の拡張性を考慮し、シミュレータの動作を記述する部分を動作定義データとし、本体と分離した。
- (3)表示部は、Xウィンドウを用いプログラムの可搬性を高めると共にマンマシン・インタフェースを向上させた。また、ソケット結合を用いネットワーク透過型としたので、制御部を表示部と異なるCPU パワーの大きなワークステーション上で動かすという使い方を可能とした。

3. シミュレータの構成

本シミュレータは、可搬性の面からC言語を用いて作成した。その構成を図1に示す。

対話部は、動作定義データを元に実験者と対話を行い、ネットワークの構築等シミュレーションの各種条件を設定する部分であり、学習部は、対話部の設定を元に学習・認識を行う部分であり、表示部は、実験の経過や結果をグラフィカルに表示する部分である。

シミュレータに使用するデータは、上記の動作定義デー

タを含め3種ある。入力データは、学習・認識で用いるデータであり、ネットワークデータは、シミュレータで構築したネットワークの構造を保持するデータである。

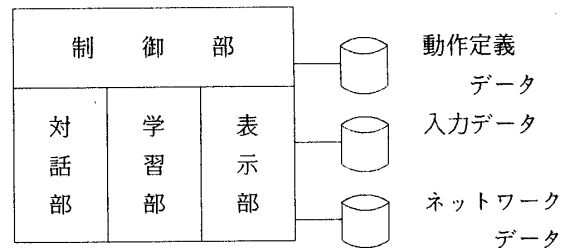


図1 シミュレータの構成図

4. ネットワーク構造の記述

シミュレーションを行う際にはネットワーク構造を定義する必要がある。ネットワークの構造には、

- ・任意の層数、任意のユニット数
- ・任意の層間で任意のユニット結合

等が考えられる。本シミュレータは、これらの構造の記述を容易に行えるように簡単なネットワーク記述言語を用意した。以下にその概要を説明する。

ユニットは複数の層からなっており、各層のユニットの位置は、1~3次元で表現できるようになっているが、以下では2次元を例にとっている。

(1)基本的な結合の設定

N層とM層のユニット間の結合を次のように表現する。

① $N \ell = M \ell$

② $N \ell (X_{N1}, Y_{N1}) = M \ell (X_{M1}, Y_{M1})$

③ $N \ell (X_{N1} - X_{N2} - S, Y_{N1}) = M \ell (X_{M1}, Y_{M1})$

①は、N層とM層の全ユニットが結合している場合、②はN層の (X_{N1}, Y_{N1}) のユニットとM層の (X_{M1}, Y_{M1}) のユニットが結合している場合、③は、N層の (X_{N1}, Y_{N1}) , (X_{N1+S}, Y_{N1}) , (X_{N1+2S}, Y_{N1}) , ... (X_{N2}, Y_{N1}) のユニットとM層 (X_{M1}, Y_{M1}) のユニットが結合している場合である。s を省略した時は、1となる。

(2)複雑な結合の設定

より複雑な構造を実現するために、ユニットの集合をサブ領域として定義し、サブ領域間の結合により全体の構造を表現する。具体的には、次のようにする。

図2にあるN層、M層のサブ領域 A_i, A_j を

$$A_i = (w_{Ai}, h_{Ai}), A_j = (w_{Aj}, h_{Aj})$$

と定義する。そしてこのサブ領域間の結合を

$$N \ell A_i (x_{N1}, y_{N1}) = M \ell A_j (x_{M1}, y_{M1})$$

と定義する。ただし、 $(x_{N1}, y_{N1}), (x_{M1}, y_{M1})$ は、 A_i, A_j の左上の座標である。このサブ領域を用いると図3のような複雑な結合も、

$$N \ell A_i (x_{N1} - x_{N2} - s_{Nx}, y_{N1} - y_{N2} - s_{Ny}) = M \ell A_j (x_{M1} - x_{M2} - s_{Mx}, y_{M1} - y_{M2} - s_{My})$$

と簡単に定義できる。

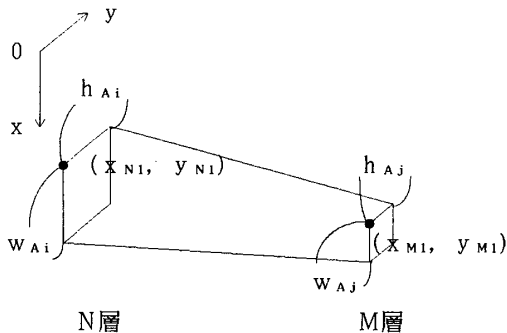


図2 サブ領域

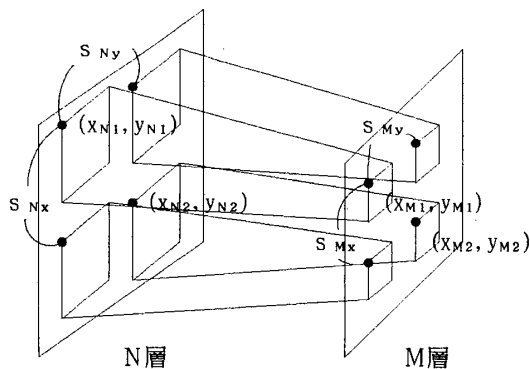


図3 ネットワークの構造例

5. シミュレータの実行

シミュレータを起動すると、対話部は動作定義データに基づき、シミュレータ動作環境を設定するための各種質問例えば、層の数、各層のユニットの数等(図4)を発する。実験者はこれに答えていくことにより、容易にシミュレーションを実行することができる。図5は、バックプロパゲーションを実行した場合の表示例である。

```

学習法を選択してください。1.BP法 2.競
1
:
層の数を入力してください。
3
各層のユニット数を入力してください。
99 40 53
:
入力データのファイル名を入力してください。
gazoul.dat
:
    
```

図4 シミュレータとの対話例

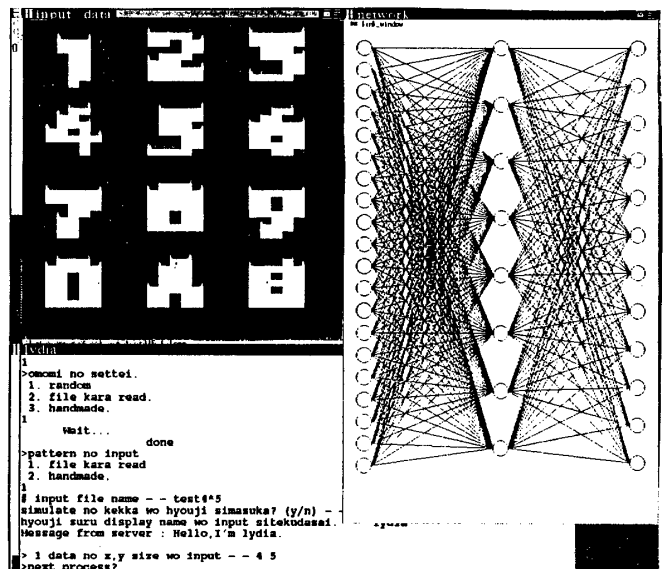


図5 シミュレータ表示例

6. おわりに

ネットワーク構造の記述が容易に行える汎用ニューラルネットワークシミュレータについて報告した。現在、学習アルゴリズムとしてバックプロパゲーション、競合学習を実装している。今後は高速化及び表示部の改良を行い、より使い易いシミュレータにしていく予定である。

【参考文献】

(1)D.E.Rumelhart, D.Zipser: "Feature Discovery by Competitive Learning", Parallel Distributed Processing vol.1, pp.151-193.
 (2)D.E.Rumelhart, G.E.Hinton, and R.J.Williams: "Learning Internal Representations by Error Propagation", Parallel Distributed Processing vol.1, pp.318-362.