

ニューラルネットを組み込んだ学習システムの

5K-6

デザインパターンによる実現

田中 宏輔* 坂本 憲広** 牛島 和夫*

*九州大学大学院システム情報科学研究科

**九州大学医学部附属病院医療情報部

1. はじめに

データ収集、蓄積技術の進歩に伴い、様々な分野で大規模なデータベースが構築されている。データベースからの知識発見(KDD)^[1]は、このような大量のデータを解析し、人間によって有用な知識を得るための技術である。KDD処理は多段階の処理から構成されており、ユーザは状況に応じてこれを反復的に行う必要がある。なぜなら、データの性質や解析目的によって、どのような手法が有効であるかという明確な指標が存在しないためである。それゆえ、KDD処理の一部として学習システムを用いる場合、手法の変更や拡張が容易に行えるような高い柔軟性が必要となる。

我々はデータ発掘手法を容易に組み込むことのできる柔軟性を備えた決定木学習システムを開発している^[2]。本研究ではKDDツールの研究の一環として、ニューラルネットの1つである階層型ネットワークをデータ発掘手法として組み込んだ学習システムを開発する。

従来のニューラルネットを利用した学習システムは、柔軟性に重点を置いておらずシステムの一部の変更や拡張が困難であるという問題点がある。そこでこの問題点を考慮し、柔軟性を備えたニューラルネットによる学習システムを実現する。高い柔軟性を実現するため、オブジェクト指向技術におけるデザインパターンを利用し、学習システムの各構成要素を再利用性の高い小さなオブジェクトに分解して設計を行う。

2. デザインパターン

デザインパターンとは、オブジェクト指向に基づくソフトウェア設計において頻りに現われるパターンに名前を付け、体系化したものである。本研究ではE.Gammaらが提案したデザインパターンカタログ^[3]に基づいて設計を行う。これは、オブジェクトの生成、構造、振舞いに関する23個のパターンを規定したもので、一貫した記述法に従い各パターンを明確かつ実践的に記述している。本文中のBUILDERパターンやSTRATEGYパターンはデザインパターンの1つである。これらのパターンを組み合わせることで、柔軟性の高いシステムを容易に構築することができる。

Development of a Neural Network Learning System with Design Pattern.

Kousuke Tanaka*, Norihiro Sakamoto** and Kazuo Ushijima*

*Graduate School of Information Science and Electrical Engineering, Kyushu University.

**Department of Medical Informatics, Kyushu University Hospital.

3. 設計

3.1 設計方針

柔軟性や拡張性を備えた学習システムを設計するためには、システムの中で変更が予測される部分(ホットスポット)^[4]を見極めることが重要である。本研究では、まず学習システムに対する要求分析を行いホットスポットを同定する。その後、得られたホットスポットの設計に対してデザインパターンを適用する。種々の学習システムの分析や調査によって同定したホットスポットを、以下の4つの項に分けて述べる。

データセットの入力・生成部: データセットはニューラルネットの学習に用いる学習データの集まりである。ニューラルネットの種類により様々な形式のデータセットを扱う必要がある。その入力方法にも柔軟性が必要である。

ユニットの振舞い: ニューラルネットの構成単位をユニット(またはニューロン)と呼ぶ。ユニットの振舞いである、出力値の計算方法・しきい値閾数・重みの更新方法を、ニューラルネットの種類や学習アルゴリズムによって柔軟に変更しなければならないという要求がある。

学習アルゴリズム: 学習アルゴリズムとはニューラルネットの学習の手順を表したものである。学習アルゴリズムにおけるホットスポットは、ユニットの重みとしきい値の更新方法、学習の終了条件の2つである。

ニューラルネットの構造: 1つのニューラルネットに対し複数の学習アルゴリズムが存在する場合がある。ゆえにニューラルネットはそれらを柔軟に変更できるような構造を備えておくべきである。

3.2 デザインパターンを適用した設計

本節では、前節で提示した各ホットスポットに対する具体的な要求を挙げ、適用するデザインパターンとその理由を示し、設計方法について述べる。

データセットの生成の際、任意の型の学習データを扱うために、学習データの種類に依存しないオブジェクトの生成過程を備えておくべきである。そこで、学習データの生成過程をカプセル化するBUILDERパターンを適用する。

学習アルゴリズムを設計する際、ユニット・ユニットの層・ネットワークのそれぞれのレベルで変更が予測される。そこで学習アルゴリズムを複合オブジェクトとして設計する。STRATEGYパターンを適用することで、アルゴリズムをカプセル化しそれらを交換可能にする。

このように、各ホットスポットについてデザインパターンを適用して設計を行った。最終的に7個のホットスポットに対し、延べ8パターンを利用した。

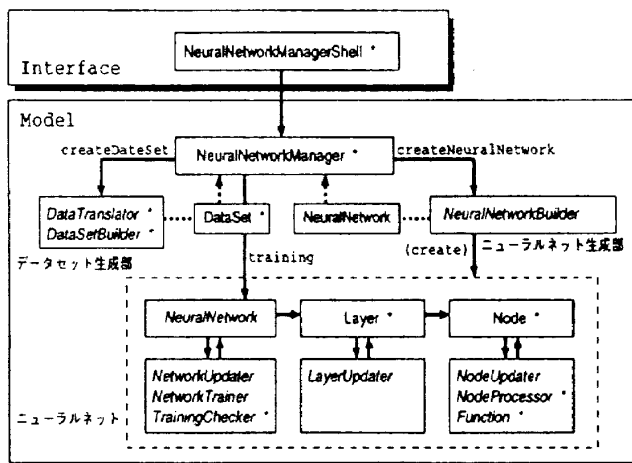


図 1: 学習システムの構成図

4. 実装

以上の設計をもとにして実装した学習システムの構成を図1に示す。実装にはオブジェクト指向言語 Smalltalk を用いた。本学習システムはモデルとインタフェースの2つの部分から構成される。モデルは学習システムの中心部分である。インタフェースはモデルとユーザとの対話的操作を実現する部分である。図1では実線で囲んだ枠内に1つまたは複数のクラス名を記述している。図1でクラス名の後に付いている*は、そのクラスあるいはサブクラスが、ニューラルネットの種類や学習アルゴリズムに依存せず汎用的に利用できることを示している。斜体で書かれているクラス名は抽象クラスを表しており、実際に学習システムを構築する場合これらの抽象クラスから派生したサブクラスを導入して実現する。図中の実線の矢印はクラス間のメッセージの流れを表す。

データセット生成部はメッセージ createDataSet を受け取ると、DataSet クラスのインスタンスを生成し NeuralNetworkManager オブジェクトへ返す。ニューラルネット生成部はメッセージ createNeuralNetwork を受け取ると、オブジェクト生成のメソッドにより、破線の枠内のクラス群をインスタンス化して組み合わせ、NeuralNetwork オブジェクトを NeuralNetworkManager オブジェクトへ返す。

5. 評価

本章では実装した学習システムが設計に対する要求を達成することができたかどうか評価する。

5.1 ユニットの振舞いに関する評価

従来のニューラルネットによる学習システムでは、「クラス継承」によるユニットの設計を行っている。ユニットの振舞いには複数のホットスポットが存在するため、従来の学習システムだと具象クラス数の著しい増加を招く恐れがある。本研究ではユニットの設計において、オブジェクトの構成要素を分解して合成する「オブジェクト合成」による設計を行った(図1のNode、NodeUpdater、NodeProcessor、Functionクラスによる合成)。この設計により、ユニットの構成要素を独立して変更あるいは拡張することが可能になった。

5.2 学習アルゴリズムに関する評価

図1の「ニューラルネット」部を構成するクラス群では、クラスごとに独立して拡張できるようクラス間の依存性を低くした。これにより学習アルゴリズムの自由な拡張が可能になった。今回実装を行った階層型ネットワークの学習システムでは、以下の項目を自由に組み合わせることが可能である。(1)階層型ネットワークの種類として、パーセプトロンもしくはBPN、(2)学習アルゴリズムとして、モーメント法など4種類の学習方法、(3)ユニットのしきい値関数として、線形関数、二値(0,1)関数、シグモイド関数、(4)学習の終了条件として、回数制限もしくは誤差制限。

5.3 ニューラルネットの拡張性に関する評価

拡張性に関する評価として、本学習システムのクラスを利用してニューラルネットの1つであるSOM(Self-Organizing Map)^[5]を試験的に実装し、組み込みにかかった手間、既存のクラスの再利用性、拡張の容易さ、適用したデザインパターンの有効性といった面から評価を行った。

SOMを組み込む際、学習システムの既存のクラスのうち6クラスをそのまま再利用することができた。既存のクラスのメソッドのうち再利用した総数は116個であったのに対し、新たに追加したメソッドの総数は55個であった。デザインパターンの利用によって拡張すべき箇所が明確になり、本学習システム構築時に比べ短時間でSOMを組み込むことができた。以上の評価より、本学習システムが様々な種類のニューラルネットを構築することのできる拡張性を備えていることを示した。

6. おわりに

本研究では階層型ネットワークを利用した学習システムの実装を行った。試験的にSOMを組み込むことにより、ニューラルネットを組み込んだ本学習システムの柔軟性や拡張性を示した。このことから設計したニューラルネットのクラス群をKDDツールのコンポーネントに再利用することも可能であると考えられる。しかしながら、システムの柔軟性とプログラムの実行効率はトレードオフの関係にあり今後検討すべき問題である。今後はこれに加え、KDDの他のシステムとの統合性、適用したデザインパターンの考察、ニューラルネットの応用方法について検討する予定である。

参考文献

- [1] Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, H.: "Advances in Knowledge discovery and data mining," AAAI/MIT Press, 1996.
- [2] 増田剛, 坂本憲広, 牛島和夫: "オブジェクト指向型決定木学習システムの開発," 第55回情報処理学会全国大会 3AE-10, 1997.
- [3] Gamma, E., Helm, R., Johnson, R. and Vlissides, J.: "オブジェクト指向における再利用のためのデザインパターン," ソフトバンク株式会社, 1995.
- [4] Pree, W.: "デザインパターンプログラミング," 株式会社トッパン, 1996.
- [5] Beale, R., Jackson, T.: "ニューラルコンピューティング入門," 海文堂, 1993.