

疑似ベイジアンネットを用いた 認知モデルのプロトタイピング手法の提案

Proposal of a Prototyping Method of Cognitive Models using Quasi Bayesian Networks

一杉裕志¹Yuuji Ichisugi¹

¹ 産業技術総合研究所 人工知能研究センター 脳型人工知能研究チーム

¹ National Institute of Advanced Industrial Science and Technology(AIST) AIRC

Abstract: We propose Quasi Bayesian Networks as a tool for efficient prototyping of cognitive models based on restricted Bayesian networks.

1 概要

制限付きベイジアンネットを用いた認知モデルのプロトタイピングを効率的に行うための手法として、疑似ベイジアンネットを用いる方法を提案する。一般にニューラルネットなど大規模な機械学習技術を用いて認知モデルの設計を行う際には、ハイパーパラメタの調整など非本質的な作業を伴う上、意図した動作をしない場合の原因の切り分けにも困難が伴う。本手法では、ベイジアンネットにおける確率値の0と非0のみを区別するよう簡略化した大脳皮質のモデルを用いることで、本質的なモデル設計作業に専念でき、効率的にプロトタイプを設計できる。実際にこの方法を用いて、脳の言語野や視覚野の機能の一部についての認知モデルの実装を現在試みている。

2 はじめに

脳のアーキテクチャを模倣した汎用人工知能を実現するには、視覚、言語、思考、運動などの個々の認知機能をつかさどる特徴的な領野のネットワークの意味を解明する必要がある。

大脳皮質のすべての領野は似通った解剖学的構造をしていることから、共通の動作原理で動いていると思われる、その有力な候補の1つに「大脳皮質がベイジアンネット [2] である」とする仮説がある [7][10]。この仮説が正しいならば、大脳皮質をつかさどる様々な認知機能は、領野のネットワーク構造を模したベイジアンネットでも再現できるはずである。

しかし、一般にニューラルネットやベイジアンネットのような機械学習技術を用いて複雑な認知機能を実現しようとする、たとえ小規模なプロトタイプであっても、ハイパーパラメタの調整など非本質的な作業が

伴う。また、モデルが意図したとおりに動作しない場合、その原因としてはモデルの設計自身の問題の他に、実装上の問題や局所解・過適合の問題などが考えられるため、本当の原因の追及には大きな手間がかかる。

そこで本論文では、ベイジアンネットを用いた認知モデルの本質的な設計作業に専念できるプロトタイピング手法として、疑似ベイジアンネットと呼ぶ、確率値の0と非0のみを区別するように簡略化したベイジアンネットを用いる方法を提案する。疑似ベイジアンネットは学習機能を持たず、条件付確率表の値を設計者が手で与える。

疑似ベイジアンネットを用いたプロトタイピングは飛行機設計における風洞実験に相当する。模型の飛行機は飛ぶことはないが、気流の乱れなどを調べて、その形状が実際に飛ぶ飛行機になるための必要条件を満たしているかどうかを検証することができる。認知モデルのプロトタイプ自体が実用に使えることは決してないし、プロトタイプで問題が見つからなかったとしても同じアーキテクチャを持った実用規模のシステムで問題が起きない保証はない。しかしながら、プロトタイピングにより潜在的な問題を早期に発見できるため、実用レベルのシステムの完成を加速することができると思われる。

以下の章では、認知モデル設計の目的、制限付きベイジアンネットと疑似ベイジアンネットの概要、認知モデルのプロトタイピングの事例について述べていく。

3 認知モデル

3.1 認知モデルと認知アーキテクチャ

人間や知能を持った動物の認識、学習、言語理解、運動制御、行動計画などの個別の機能の実現メカニズム

に関する仮説を単純化して表現したものを、ここでは認知モデルと呼ぶ。

また、部品である個々の認知モデルを1つに結合し、ある環境の中である目標の達成に向けて知的に動作するエージェントにするための設計図を認知アーキテクチャと呼ぶ。

筆者は、ベイジアンネットを用いて大脳皮質の視覚野、言語野、運動野、前頭前野などがつかさどる機能の認知モデルを設計しようとしている。個々の認知モデルが実用レベルの性能で動作するならば、それらを脳全体のアーキテクチャ（脳の認知アーキテクチャ）にしたがって結合し、さらに海馬や大脳基底核などの大脳皮質以外の計算論的モデルとも結合することで、最終的には脳全体の機能が再現できると考える。

3.2 認知モデルを表現する道具としてのニューラルネットとベイジアンネット

ニューラルネットワークは古くから人間の認知機能の原理に関する仮説を表現するための道具として使われている。例えば Hopfield network は連想記憶のモデル、Neocognitron[1] は変形つよい文字認識のモデルである。Neocognitron は Convolutional neural network[5] として工学的に発展し、画像認識等において高い性能を発揮している。

一方で、大脳皮質がベイジアンネットであるという仮説に基づいて、脳の様々な認知機能や現象を説明する動きもある。ベイジアンネットはニューラルネットワークとは違い、(1) 特徴量を値ではなく値の確率分布で表現する、(2) 情報が入力から出力への一方向ではなく、一部の確率変数に与えられた観測値をもとに全体の尤度を最適化することで推論する、といった特徴があり、より本物の脳に近い認知機能を実現できる可能性がある。実際にある種のベイジアンネットを脳の認知機能やニューロンの応答特性の再現に用いる例としては、視覚野の腹側経路・背側経路・注意のモデル [8] がある。

3.3 認知モデルが示すべきこと

認知モデルは、認知科学的妥当性、神経科学的妥当性、工学的有用性を備えていることが望ましい。

認知科学的妥当性を示すためには、対象とする認知機能が正しく実現されていることをまず示す必要がある。例えば視覚野であれば脳と同様の頑健さで画像認識ができていなければならない。それに加え、脳と同様の錯覚を起こすこと、同様の発達過程を示すこと、損傷時に脳と同様の不具合を示すことなどが示されれば、より正しい認知モデルとしての説得力が増すだろう。

神経科学的妥当性を示すためには、知られている脳内の現象に対して何らかの説明と予測を与えることが必要である。単に機能的な入出力関係を模倣できるだけでなく、各領野に固有のニューロン応答や解剖学的特徴が再現・予測できれば、その領野の正しいモデルとしての説得力が増す。

工学的有用性を示すためには、性能の高さを示す必要がある。例えば認識精度、計算コストの小ささ、学習データの少なさなどの指標がある。

役に立つ情報処理技術の実現が目標であれば、本来は工学的有用性さえ示せばよいのだが、一般にそれには長い時間がかかる。認知科学的妥当性と神経科学的妥当性を示すことは、認知モデルが将来脳と同じ性能を出す潜在的可能性があることを示す、有望さの証拠としての意味がある。

疑似ベイジアンネットを用いた認知モデルは、プロトタイプとはいえ、この中のいくつかは示すことができる。その例は6節で述べる。

4 制限付きベイジアンネット

4.1 条件付確率表の制約

大脳皮質は、規模が大きくなっても計算量の爆発が起きないという特殊な性質を持ったベイジアンネットであろうと筆者は考えている。通常のベイジアンネットは、親ノードの数が増えると、確率伝搬アルゴリズム [2] などの近似推論の1ステップに必要な計算量が指数関数的に増えるという問題がある。ベイジアンネットのネットワークの形や条件付確率表の形に強い制約を入れれば計算量の爆発を抑えることは容易だが、モデルとしての表現力が弱くなり、実際のタスクで高い学習精度を出すことはできない。したがって、計算効率と表現力をいかに両立するかが課題となる。

筆者がこれまで実装してきた BESOM Ver.3 では、条件付確率表に入れた制約を前提に確率伝搬アルゴリズムを最適化することにより、推論動作を高速化していた [11]。しかし制約が強すぎるためモデルとしての表現力が弱く、パターン認識に応用した場合に十分な認識精度が出ないという問題があった。また、視覚野や言語野の機能のごく一部を表現することも困難だった。そこで現在は別の種類の条件付確率表のモデルの導入を検討している。具体的にはORノード、ゲートノード、排他ノードの3種類のモデルである (図1)。

疑似ベイジアンネットを用いたプロトタイプングの目的の1つに、これらの3種類のノードの有効性を確認することがある。具体的には、視覚野や言語野などが持つ様々な認知機能を少ないパラメタのネットワークで表現できる表現力を持っていること、そのネットワー

クにおいて、パラメタをデータから学習可能になりそのようなことなどを確認するのが目的である。

4.2 3種類のノードの振る舞い

以下に3種類のノードの振る舞いについて説明する。すべてのノードは2値ノードである。

ORノードは、noisy-ORモデル[2]として知られているものと同じである。親との間の結合の重みがすべて1であれば、親ノードのどれか1つが1であるときにのみ、自分の値は1になる。

ゲートノードは、複数の制御ノード C_k と、1個の上流ノード U を親ノードとして持つ。結合の重みがすべて1であれば、制御ノードのすべてが0かつ上流ノードが1のときのみ、自分の値は1になる。

排他ノードは、複数の親ノードのうち、2個以上の値が1のときに自分の値は0、それ以外の場合は1になる。可変の結合パラメタは持たない。ネットワークの状態を推定する際には観測値として1を与えることで、親ノードの中の高々1つだけが活性化するような制約を与えることができる。これにより複数の2値ノードを1つ多値ノードのように振る舞わせることができる。

なお、これらのノードはすべて、ネットワークの形を変換することによる推論の高速化の手法[3]が適用可能である。親ノードの数が m 個の時、親ノードの数が高々定数個で深さが $O(m)$ の形のネットワークに変換することにより、 $O(m)$ で近似推論・学習の1ステップが実行できるようになる。この手法により計算量の爆発の問題は回避できる。(ただし、実装による確認は、ORノード以外は現時点では行っていない。)

5 疑似ベイジアンネット

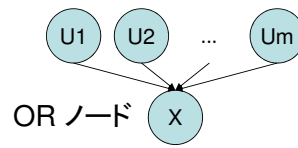
5.1 疑似ベイジアンネット QBN と QBC

通常のベイジアンネットは区間 $[0, 1]$ の値を取る確率値を扱うが、疑似ベイジアンネットでは、確率値の0と非0の区別のみを行うように簡略化して定義する。通常のベイジアンネットでは一部の確率変数の観測値が与えられたときに尤度が最大になる残りの変数の値の組(MPE, Most Probable Explanation)を求める、という推論を行うことができるが、疑似ベイジアンネットに対しては、それは行えない。その代わりに、観測変数と隠れ変数の値の同時確率が非0になる値の組み合わせをすべて求めることになる。

条件付確率表に制約のない一般の疑似ベイジアンネットをQBN(Quasi Bayesian Networks)、制約のあるものを現在仮にQBC(Quasi Bayesian Cognitive circuits, 疑似ベイジアン認知回路)と呼んでいる。

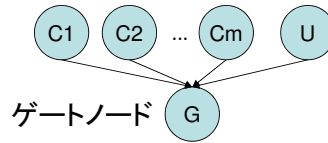
$$P(X = 0 | u_1, \dots, u_m) = \prod_{k=1}^m (1 - w_k)^{u_k}$$

$$P(X = 1 | u_1, \dots, u_m) = 1 - \prod_{k=1}^m (1 - w_k)^{u_k}$$



$$P(X = 0 | c_1, \dots, c_m, u) = 1 - (1 - w_u)^{-u} \prod_{k=1}^m (1 - w_k)^{c_k}$$

$$P(X = 1 | c_1, \dots, c_m, u) = (1 - w_u)^{-u} \prod_{k=1}^m (1 - w_k)^{c_k}$$



$$P(R_X = 0 | X_1, \dots, X_m) = 1 \text{ (} X_k \text{のうち2つ以上が1の場合)}$$

$$P(R_X = 1 | X_1, \dots, X_m) = 1 \text{ (} X_k \text{のうち高々1つが1の場合)}$$

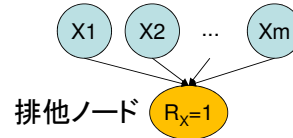


図1: 制限付きベイジアンネットへの導入を検討中の3種類のノード。変数はすべて0か1の値を取る2値ノード。結合の重み w_k, w_u は区間 $[0, 1]$ の値を取る。ここでは 0^0 は $\lim_{x \rightarrow +0} x^0$ すなわち1とする。

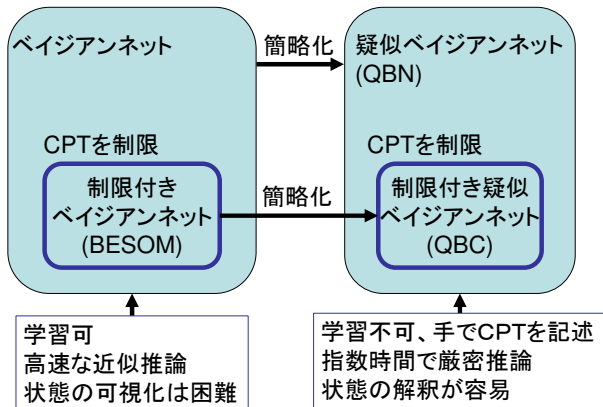


図2: 疑似ベイジアンネットワークと制限付きベイジアンネットワークの関係。QBN と QBC はどちらも確率値の0と非0の区別のみを行う簡略化されたベイジアンネットワークである。BESOM と QBC はどちらも条件付確率表 (CPT, Conditional Probability Table) に制限を入れたベイジアンネットワークである。

QBN を認知モデルの記述の道具に使うこともできるが、実際に筆者が構文解析器を記述してみたところ、多くの問題が発生した。例えば親ノードの数に対して指数関数的なサイズの条件付確率表を定義する必要がある。モデルのプロトタイピングの目的の1つに制限付きベイジアンネットワークの結合パラメタ数の爆発が起きないことを確認があるが、QBN ではそれを確認することができない。また、我々がこれまで開発してきた制限付きベイジアンネットワーク BESOM では各ノードを特徴量の組を並べた関連データベースと解釈することが可能だが、QBN を用いた記述では関連データベースのような書き方にはならず、関連した情報が分散して記述され、可読性・保守性に難点があった。

そこで、制限付きベイジアンネットワークと直接的に対応がつかない制限付き疑似ベイジアンネットワーク QBC を定義した。一般のベイジアンネットワーク、制限付きベイジアンネットワーク、疑似ベイジアンネットワーク QBN と QBC の間の関係は図2のようにまとめられる。

以下に続く節で QBN と QBC の概要を説明する。

5.2 疑似ベイジアンネットワーク QBN

条件付確率に制約のない疑似ベイジアンネットワーク QBN がどのようなものかを、以下に具体例で説明する。5つの確率変数の同時確率を以下のように因子分解して表現するベイジアンネットワークを考える。

$$\begin{aligned}
 &P(N_1, N_2, N_3, N_4, N_5) \\
 &= P(N_4|N_3)P(N_5|N_3)P(N_3|N_1, N_2)P(N_1)P(N_2)
 \end{aligned}
 \tag{1}$$

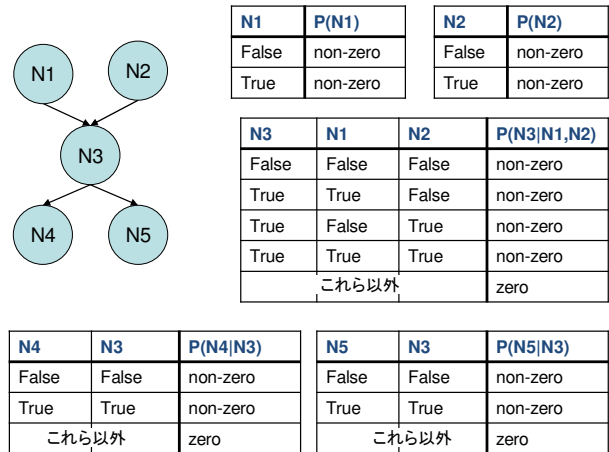


図3: QBN の例。

さらに、条件付確率表が図3で定義されるとする。

疑似ベイジアンネットワークが与えられれば、すべての変数の値の同時確率が非0になる組み合わせ（これを疑似ベイジアンネットワークの解と呼ぶ）の集合を求めることができる。図4は、図3の疑似ベイジアンネットワークのすべての解である。式1の左辺（同時確率）が非0になるのは右辺の各因子（各条件付確率）がすべて非0になる時に限られる。例えば図4の解2の値を式1の右辺の各条件付確率の式に代入して得られる値 $P(N_4 = True|N_3 = True)$, $P(N_5 = True|N_3 = True)$, $P(N_3 = True|N_1 = False, N_2 = True)$, $P(N_1 = False)$, $P(N_2 = True)$ はすべて、図3の条件付確率表で非0と定義されている。

疑似ベイジアンネットワークは定量的な確率計算を捨てているが、ベイジアンネットワークの本質的な特徴の一部は引き継いでいる。解を定めるための制約条件は、各ノードが持つ条件付確率表において局所的に定義される。一部の確率変数に観測値が与えられれば、その情報がネットワーク全体に伝搬し、ネットワーク内の各ノードの制約をすべて満たす値の組み合わせのみが解として出力される。すなわち、真のベイジアンネットワークが持つ(1) 確率分布を扱う、(2) ネットワーク全体を最適化する、という2つの特徴のうち前者は大幅に簡略化しているが後者はそのまま残したものが疑似ベイジアンネットワークであると言える。

図3の例では確率変数の取り得る値として True, False という2つの記号を用いたが、相互に区別可能でさえあれば一般にどのような記号を用いてもよい。例えば構文解析のモデルを記述するのであれば、NP（名詞句）や VP（動詞句）といった記号を用いることで、定義や解の可読性を上げることができる。また、確率変数は2値である必要はなく、多値でも構わない。

	N1	N2	N3	N4	N5
解1	False	False	False	False	False
解2	False	True	True	True	True
解3	True	False	True	True	True
解4	True	True	True	True	True

図 4: QBN のすべての解の例。

5.3 制限付き疑似ベイジアンネット QBC とその実装

4章で述べた制限付きベイジアンネットを疑似ベイジアンネットとして表現したものが制限付き疑似ベイジアンネット QBC である。

QBC を効率的に記述できるようにするため、Java プログラムのソースコード中に埋め込んで記述できる DSL(ドメイン固有言語, domain-specific language) を実装した。記述言語の詳細な仕様は別の機会に述べることにするが、ここでは簡単に特徴を説明する。

DSL は Java 8 の標準的機能の範囲内で、enum, default method の機能を使って実装されており、比較的記述が簡潔になる上、確率変数や値の名前のスペルミスがコンパイルエラーとして検出できるという特徴を持つ。(Eclipse などの統合開発環境内であれば記述時にスペルミスを検出できる。) また、繰り返し構造を持つネットワークを Java のプログラムで半自動生成することも容易である。

DSL で定義された QBC は、トランスレーターによりまず QBN に変換され、その後、全解探索が行われる。探索には、簡単な枝刈りを行うだけの実装と、SAT ソルバ¹ を用いる実装の 2 つがある。いずれの実装でも解の探索にはノード数に対して最悪指数時間がかかるが SAT ソルバを用いれば比較的大きなネットワークも扱えると思われる。

5.4 QBC を使う利点と欠点

認知モデル設計に QBC を使う利点は以下のようにまとめられる。

1. 記号処理に近い動作をしつつも、ネットワークの状態を最適化するというベイジアンネットの本質的特性は残している。
2. 機械学習特有の局所解・過適合やオーバーフロー・アンダーフロー、NaN などの問題が起きない。
3. 決定論的に動作するためモデルが意図した通りに動かないときの原因の究明が容易。

```
net.tableNodeList.add(tableNode(P13, table(
    row(N1,N11,N12),
    row(N2,N21,N22),
    row(N11,N111,N112),
    row(N12,N121,N122),
    row(N21,N211,N212),
    row(N22,N221,N222),
    row(I,_,_)));
net.tableNodeList.add(tableNode(J15, table(
    row(J2,
        I,I,I,
        or(J3,J4),_,_,
        --,
        0,I,I,
        0,I,I),
    row(J3,
        __,I,I,
        I,I,__,
        I,
        I,0,I,
        I,0,I),
    row(J4,
        __,or(J2,J3),_,_,
        I,I,I,
        --,
        I,I,0,
        I,I,0)
)));
```

図 5: QBC を記述する DSL を用いた記述例。通常の Java のソースコード中に記述を埋め込むことができる。この例は 6 章で述べる構文解析器の定義の一部である。

¹ライブラリ Sat4j を使用。 <http://www.sat4j.org/>

4. 専用記述言語 (DSL) の利用によりモデルの記述が容易。
5. 変数の値に意味のある文字列が使えるのでモデルと解の可読性が高い。

一方で、真のベイジアンネットと比較して、以下の制限がある。

1. データから条件付確率表を学習することは想定していない。条件付確率表の値は設計者がすべて手で与える。
2. 推論には最悪指数時間を必要とするため、極端に大規模な回路設計はできない。
3. 確率値の0と非0区別のみをするため、一番もっともらしい解を1つ選択する、という本来のベイジアンネットなら可能な推論動作をさせることはできない。

6 認知モデルのプロトタイピングの事例

6.1 構文解析器の実現

QBC を用いた認知モデルの例として、確率文脈自由文法 (PCFG, Probabilistic Context-Free Grammar)²の構文解析器のプロトタイプについて説明する。

PCFG は文脈自由文法を拡張したもので、非終端記号から生成規則を選択するための確率値が決められている。文の生成過程で条件付確率を使うという点で、PCFG はベイジアンネットと共通点を持っている。実際、工学的な動機から、普通のベイジアンネットを用いた PCFG の構文解析器の構築が試みられた [4]。しかし、パラメタ数や推論にかかる時間は構文解析対象の単語数の上限に対し指数関数的に増えるものであり、スケーラビリティに問題がある。

筆者は制限付きベイジアンネットを用いて PCFG の構文解析器を効率的に実装できれば、それをさらに発展させることで脳が行う構文解析の認知モデルに近づけるのではないかと考え、そのプロトタイプを QBC を用いて実装した³。

ネットワークの規模が大きくなると記述に間違いが入りやすい上、計算量爆発により解の探索が事実上できなくなる恐れがあるため、プロトタイピング時には問題の本質を保ったまま極力扱う問題のサイズを小さ

²参考: [6]

³なお、制限付きベイジアンネットを用いたとしても、ネットワーク規模が指数関数的に爆発しないように構文解析器を構築するにはいくつかの自明でない工夫を入れる必要があるが、その詳細については本稿の範囲ではないので別の機会に説明する。

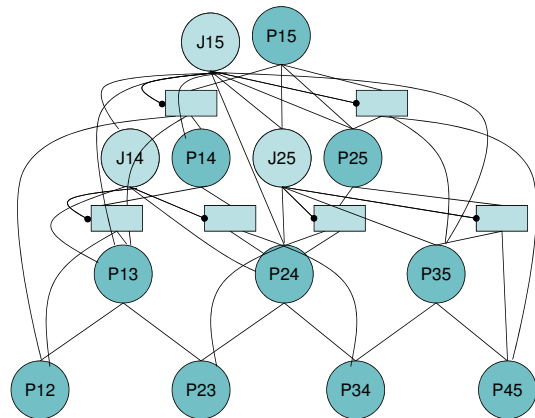


図 6: 構文解析を行う QBC ネットワークを、ネットワーク設計用の記法を用いて図式化したもの。

くする必要がある。今回は、構文解析できる単語の数は4個に固定、文法もあいまいさの生じない単純なものとした。

図6は、構文解析を行うQBCのネットワークである。なお、この図はQBCネットワーク設計用の独自記法を用いており、丸いノードはORノード、四角いノードはゲートノードの行列を表している。QBCを用いた認知モデル設計は、まずこの記法を使ってネットワークの設計を行った後、DSLでネットワーク構造と結合行列の要素を記述し、動かしてデバッグする、という手順になる。この構文解析器のDSLによるQBCの定義サイズは約160行である。

プロトタイプではまだ文の解釈の曖昧さが生じない簡単な文法でしか試していない段階だが、実際に構文解析器として動作するであろうことが確認できた。ネットワークのすべての解を出力させると、可能な文がすべて生成される。単語列を観測データとして与えれば、可能な解釈がすべて解として出力されるはずである。もし単語列が文として受理できない場合は、解の集合は空集合となる。

ネットワークのパラメタ数は指数関数的に爆発することはなく、単語数 n に対して多項式のオーダーで抑えられている。

実装した構文解析器は、(1) 上位層ほど受容野が広い、(2) 上位層ほど抽象的という脳の視覚野や深層学習と共通の特性を持っており、認知モデルとしての神経科学的妥当性の1つと言える。

6.2 その他の認知モデル

構文解析器のほかに、現在様々な認知モデルのプロトタイピングに着手している。

言語野に関連したものとしては、我々は組み合わせ範疇文法に基づく日本語文法 [9] の構文解析を行うネッ

トワークの構築を目指している。そのために必要となる要素技術として、2つの前提から1つの結論を推論する推論規則の適用の機構と、単語列と意味表現の間の相互変換の機構の最初のプロトタイプを実装した。

視覚野に関連したものとしては、腹側経路と背側経路の相互作用のモデルとして、視覚刺激の中の物体の位置と形を同時に認識する機構や、複数の物体があったときに手前の物体が奥の物体を覆い隠す状況（オクルージョン）を処理する機構のプロトタイプを実装した。

ベイジアンネットの条件付確率表に強い制約を入れると表現力が落ちる危惧があったが、現在のところ QBC を用いた認知モデル記述に大きな不都合はなく、表現力が十分に保たれていると言えそうである。

現在までに実装した認知モデルはいずれも問題のサイズを極力小さくした簡単なプロトタイプに過ぎないが、今後はより複雑な機能を持ったネットワークに発展させていく予定である。それにより、制限付きベイジアンネットの表現力の検証と、疑似ベイジアンネットを使ったプロトタイピングの有効性の検証を進めていく。

7 まとめと今後

制限付きベイジアンネットを用いた認知モデルの本質的な設計作業に専念できるプロトタイピング手法として、疑似ベイジアンネット QBC を用いる手法を提案した。また実際に本手法が役立つ事例として構文解析器の設計について紹介した。

本稿ではプロトタイピング手法の提案の段階にとどまっている。この手法で設計したプロトタイプがそのまま実用レベルのシステム開発に本当に役立つことを実証することは、今後の課題である。そのためには、学習機能を持った BESOM で、プロトタイプと同じアーキテクチャを実装し、大量の実データから学習して性能を評価する必要がある。

本稿では示さなかったが、疑似ベイジアンネットは、おそらく認知モデルの動作の可視化がしやすいという利点も持っている。一般に大規模ニューラルネットや大規模ベイジアンネットはその認識仮定で具体的に何が起きているのかを正確に可視化することは難しい。疑似ベイジアンネットは人間に理解しやすい記号を用い、認知モデルの本質部分をコンパクトに表現でき、決定論的に動作するため、その動作も可視化しやすいと思われる。大脳皮質ベイジアンネットモデルの研究を行う研究者が多くない理由の1つには、従来のモデルの動作が直観的に理解しにくかったということもあるだろう。疑似ベイジアンネットを用いることで、情報が双方向に伝搬し全体を最適化するというベイジアンネット

の本質的動作をより多くの人に理解してもらえるデモを作ることができるだろう。

謝辞

この成果は、国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務の結果得られたものである。

参考文献

- [1] K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4): 93-202, 1980.
- [2] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, 1988.
- [3] David Heckerman, Causal Independence for Knowledge Acquisition and Inference, In Proc. of UAI-93, 122-127, 1993.
- [4] D. V. Pynadath and M. P. Wellman, Generalized queries on probabilistic context-free grammars, In Proc. of the 14th National Conference on Artificial Intelligence (AAAI '96), pp.1285-1290, 1996.
- [5] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86 (11), 2278-2324, 1998.
- [6] 北研二, 辻井潤一, 言語と計算 (4) 確率的言語モデル, 東京大学出版会, 1999.
- [7] T.S. Lee, D. Mumford, Hierarchical Bayesian inference in the visual cortex. *Journal of Optical Society of America*, A 20(7): pp.1434-1448, 2003.
- [8] S. Chikkerur, T. Serre, C. Tan, T. Poggio, What and Where: A Bayesian Inference Theory of Attention, *Vision Research*, 2010.
- [9] 戸次大介, 日本語文法の形式理論-活用体系・統語構造・意味合成-, くろしお出版, 2010.
- [10] 一杉裕志, 解説: 大脳皮質とベイジアンネット, 日本ロボット学会誌 Vol.29 No.5, pp.412-415, 2011.

- [11] Yuuji Ichisugi and Naoto Takahashi, An Efficient Recognition Algorithm for Restricted Bayesian Networks, In Proc. of 2015 International Joint Conference on Neural Networks (IJCNN), 2015.