中間報告書

創域理工学部　情報計算科学科　4 年
学籍番号 : 6322045
砂川恵太朗

提出日 : 2025 年 8 月 27 日

1　はじめに

近年，深層学習を筆頭とする機械学習技術は目覚ましい進歩を遂げ，特に大規模言語モデル（LLM）は，自然言語処理の分野において人間を凌駕する性能を示すまでになった．しかしその一方で，その成功は膨大な計算資源とデータ量を前提としており，生物の脳が持つ圧倒的なエネルギー効率や柔軟性とは未だ大きな乖離がある．また，現在の LLM の多くはテキストという単一のモダリティに特化しており，我々が日常的に経験するような，複数の感覚情報を統合して世界を認識する能力については限界を抱えている．

この課題を克服し，真に汎用的な知能へと至るためには，マルチモーダルな情報を扱え，かつ生物学的な妥当性を備えた新しい計算モデルの探求が不可欠である．そこで本稿では，数ある感覚の中でも特に外界理解の根幹をなす「視覚」情報処理に着目し，脳の情報処理原理に基づいた新たなネットワークモデルを提案する．

ここで，脳の情報処理原理として，カール・フリストンが提唱した自由エネルギー原理（Free Energy Principle）[2] を取り上げる．これは，知覚や学習といった脳の広範な機能を統一的に説明する理論的枠組みとして注目を集めている。この原理によれば、生物は環境からの感覚入力と内部モデルの予測との間の不一致（予測誤差）を最小化することで、世界の不確実性を減らし、適応的に行動している。この自由エネルギー原理を、生物学的に妥当性の高い神経回路モデルとして具体化したものが予測符号化（Predictive Coding: PC）である。予測符号化は、大脳皮質の階層的な情報処理様式、特にトップダウンの予測信号とボトムアップの誤差信号の相互作用を巧みに説明できることから、有力な脳の計算モデルと見なされている。

しかし、従来の予測符号化モデルの多くは、トップダウンの予測に基づく生成モデルか、ボトムアップの信号処理に基づく識別モデルのいずれかに特化しており、脳が持つ柔軟な情報処理能力を完全に再現するには至っていなかった。こうした背景のもと、Oliviers らによって提案された双方向予測符号化（Bidirectional Predictive Coding: bPC）[3] は、生成と識別の両方の情報処理を単一のエネルギー関数を最小化する過程で自然に両立させる画期的なモデルである。bPC は、教師あり学習における高い分類性能と、教師なし学習における優れた表現学習能力を同時に達成し、人間の視覚認知システムが持つ二重の機能、すなわち「世界がどう見えるかを予測する能力（生成）」と「見えたものが何かを判断する能力（識別）」を統合的に実現するモデルとして大きな可能性を秘めている。

本研究では、この bPC モデルの生物学的妥当性をさらに一歩進めることを目指す。具体的には、神経活動を非同期的なスパイクの発生としてモデル化するスパイキングニューラルネットワーク（SNN）上に bPC を実装する。SNN は、その時間ダイナミクスやエネルギー効率の観点から、実際の生物学的ニューロンの振る舞いをより忠実に模倣した計算モデルである。bPC を SNN の枠組みで実現することにより、bPC が仮定する計算プロセスを、より人間の視覚ネットワークに近い、生物学的に現実的な形で検証することが可能となる。

予測符号化を SNN へ実装する試みはこれまでにもいくつか報告されているが、bPC の持つ双方向の予測誤差計算と学習則を SNN 上で効果的に実現するためには、ネットワークアーキテクチャの選定が極めて重要となる。多くの SNN モデルの中から、本研究では、予測誤差を陽な形で表現するエラーニューロンをネットワーク内に組み込んだ Spiking Neural Coding Network[4] のアーキテクチャを採用する。このモデルは、bPC が定義するトップダウンおよびボトムアップの予測誤差をネットワーク上の特定のニューロン活動として直接的に表現できるため、bPC のダイナミクスやヘブ則に基づく学習ルールを自然かつ忠実に組み込む上で最も適していると判断した。本稿では、このアーキテクチャに基づき、bPC の SNN 実装を提案し、その有効性を検証する。

2 関連研究

2.1 予測符号化

自由エネルギー原理は、生物（特に脳）が、環境からの感覚入力に対して驚き（サプライズ）を最小化するように、内部状態を更新し続けるシステムであると仮定する。しかし、サプライズそのものは直接計算することが困難であるため、その上限である変分自由エネルギー（Variational Free Energy: VFE）$\mathcal{F}$ を代わりに最小化する。VFE は以下のように定義される。

$$\mathcal{F} = \underbrace{D_{KL}[q(\mu|o)||p(\mu)]}_{\text{Complexity}} - \underbrace{\mathbb{E}_{q(\mu|o)}[\ln p(o|\mu)]}_{\text{Accuracy}} \tag{1}$$

ここで、o は感覚入力（観測）、μ はその感覚入力を引き起こした世界の外的状態（隠れ状態）を表す。$p(o|\mu)$ は観測が隠れ状態からどのように生成されるかを示す生成モデル、$p(\mu)$ は隠れ状態の事前分布である。$q(\mu|o)$ は、脳が持つ世界の隠れ状態の近似的な事後分布（信念）である。

この式の第一項（Complexity）は、事後的な信念 $q(\mu|o)$ が事前知識 $p(\mu)$ からどれだけ離れているかを示すカルバック・ライブラー（KL）ダイバージェンスであり、モデルの複雑さを表す。第二項（Accuracy）は、現在の信念 $q(\mu|o)$ の下で、感覚入力 o をどれだけうまく説明できるか、すなわちモデルの正確さを示す。脳は、この複雑さと正確さのバランスを取りながら、VFE を最小化するように内部状態（信念 $q(\mu|o)$）とモデルパラメータ（シナプス荷重）を更新する。

予測符号化は、この VFE 最小化を神経回路で実現するための具体的な計算プロセス（プロセス理論）として広く受け入れられている。PC では、上位の神経層が下位の層の活動を予測し、その予測誤差を最小化するようにニューロン活動とシナプス荷重が更新される。この予測誤差の最小化が、VFE の勾配降下法による最小化と等価であることが示されている。本研究で比較対象となる各 PC モデルのエネルギー関数は、この FEP の枠組みから以下のように理解できる。

生成的予測符号化：古典的な PC は、脳を生成モデルとして捉える。すなわち、高次の概念（例：「猫がいる」）から低次の感覚情報（例：網膜に映る像）をトップダウンで予測する。この枠組みは、VFE の Accuracy 項、すなわち $\mathbb{E}_q[\ln p(o|\mu)]$ を最大化することに主眼を置いている。ガウス分布を仮定すると、この対数尤度の最大化は二乗誤差の最小化と等価になる。生成的予測符号化のエネルギー関数は、これを階層的なネットワークで具体化したものである。

$$E_{\text{gen}}(x, W) = \sum_{l=1}^{L-1} \frac{1}{2} ||x_l - W_{l+1} f(x_{l+1})||^2 \tag{2}$$

ここで、x_l は第 l 層のニューロン活動であり、VFE における信念 $q(\mu|o)$ の一部に相当する。$W_{l+1} f(x_{l+1})$ は、上位層 x_{l+1} から下位層 x_l へのトップダウン予測である。各層における予測誤差の二乗和である E_{gen} を最小化することは、階層的な生成モデルが感覚入力をうまく説明できるように内部状態を最適化するプロセスであり、FEP における推論（Inference）に対応する。

識別的予測符号化：一方、脳は感覚入力から高次の特徴を抽出する、ボトムアップの判別的な処理も行う。これは、感覚入力 o から隠れ状態 μ を推論する過程 $p(\mu|o)$ に焦点を当てたものと解釈できる。識別的予測符号

化のエネルギー関数は、このボトムアップの予測を行うモデルである。

$$E_{\text{disc}}(x, V) = \sum_{l=2}^{L} \frac{1}{2} ||x_l - V_{l-1} f(x_{l-1})||^2 \tag{3}$$

この式では、下位層 x_{l-1} が上位層 x_l の活動を予測する。これは、従来の順伝播型ニューラルネットワークで行われる計算を、予測誤差最小化の枠組みで再定式化したものと見なせる。このモデルは、入力データからラベルを予測するような教師あり学習タスクで高い性能を発揮するが、トップダウンの生成プロセスを持たないため、データの生成や欠損情報の補完などは困難である。

ハイブリッド予測符号化：ハイブリッド予測符号化は、生成モデルの持つ反復的な推論能力と、判別モデルの持つ高速な順伝播処理を組み合わせることを目指したモデルである。そのエネルギー関数は、E_{gen} に判別的な項を追加した形をしている。

$$E_{\text{hybrid}}(x, W, V) = \sum_{l=1}^{L-1} \frac{1}{2} ||x_l - W_{l+1} f(x_{l+1})||^2 + \sum_{l=2}^{L} \frac{1}{2} ||\text{sg}(x_l) - V_{l-1} f(\text{sg}(x_{l-1}))||^2 \tag{4}$$

このモデルの重要な特徴は、第二項に停止勾配（stop-gradient: sg）演算子が含まれている点である。これにより、ボトムアップの判別経路（V でパラメータ化）は、エネルギー最小化のための反復的な推論ダイナミクスには直接影響を与えず、主にニューロン活動の初期値を設定する役割を担う。つまり、まず判別経路で高速な初期推論を行い、その後、生成経路で時間をかけてその推論を精緻化するという、二段階のプロセスを実装している。しかし、推論中に二つの経路が分離されているため、両者の相乗効果が限定的であるという課題が残る。bPC は、これらのモデルの長所を統合し、単一のエネルギー関数内で双方向の予測を同時に行うことで、この課題を克服することを目指すものである。

2.2 スパイキングニューラルネットワーク

スパイキングニューラルネットワークは，神経活動を精度よくモデル化しようとするネットワークである．サーベイ論文 [1] で整理された、スパイキングニューラルネットワーク（SNN）における予測符号化の実装アプローチを概観する。このサーベイでは、モデルが「予測誤差をどのように表現し、利用するか」という観点から、既存の研究が大きく 3 つのクラスに分類されている。

クラス 1：陽的な誤差ニューロンを持つモデル (Models with Explicit Error Neurons) このクラスのモデルは、予測符号化の理論的枠組みを最も直接的に実装する。すなわち、感覚情報などを表現する「表現ニューロン」とは別に、予測と実際の入力との不一致を計算し、その誤差信号を出力するための専門の「誤差ニューロン」がネットワーク内に陽に存在する。本稿で提案するモデルはこのクラスに分類される。

クラス 2：膜電位を誤差として用いるモデル (Models with the Membrane Potential as a Prediction Error) クラス 2 のモデルでは、独立した誤差ニューロンを設けず、個々のニューロンの膜電位そのものが予測誤差を表現する。ニューロンへの入力（感覚情報）とそのニューロン自身の活動履歴に基づく予測との差が膜電位として蓄積され、この「誤差」がある閾値を超えた場合にのみスパイクを発生させる。これにより、効率的な情報表現を実現する。

クラス 3：暗黙的な誤差符号化を行うモデル (Models with an Implicit Prediction Error Encoding) このクラスでは、予測誤差に相当する信号を陽に定義・計算することはない。代わりに、興奮性と抑制性のシナプス結合の競合的な学習などを通じて、予測可能な（冗長な）情報は抑制され、予測から外れた（新規性のある）情報のみがネットワークを伝播する、という予測符号化と類似した振る舞いが創発的に実現される。

本研究の基盤となるクラス 1 は、予測を生成する表現ニューロン群と、その予測と入力信号との差分を計算する誤差ニューロン群を分離して構成するアプローチである。以下，表 1に SpNCN 以外の関連するモデルを示す.

表 1　クラス 1 に属するモデル

著者	予測	ニューロン	学習則	説明された現象
Wacongne et al.	トップダウン入力	Izhikevich	固定重みと STDP	予測符号化を適用して ミスマッチ陰性電位を 再現する
Lan et al.	トップダウン入力 （学習時のみ）	IF	自由エネルギー原理	誤差逆伝播法を 予測符号化で 置き換える方法
Fraile et al.	側方入力	LIF	固定重み	生物学的に妥当な V1 ネットワーク における明示的な 予測誤差的ニューロン の観察
Lee et al.	トップダウン入力	ADEX	レートベースの ヘブ学習	視覚入力サンプルの 生成/再構成

3　提案手法

3.1　bPC

セクション 2 で予測符号化のエネルギー関数について記したが，トップダウン予測とボトムアップ予測のできる bPC のエネルギー関数は次のようになっている.

$$E(x, W, V) = \sum_{l=1}^{L-1} \frac{\alpha_{gen}}{2} ||x_l - W_{l+1} f(x_{l+1})||_2^2 + \sum_{l=2}^{L} \frac{\alpha_{disc}}{2} ||x_l - V_{l-1} f(x_{l-1})||_2^2 \tag{5}$$

ここで，W_l はトップダウンの重み，V_l はボトムアップの重み，f は活性化関数である．α_{gen} と α_{disc} はスカラーの重み定数であり，ボトムアップとトップダウンの予測誤差の大きさの違いを考慮するために必要なものである．重み定数は学習可能な精度パラメータとみなすことができるが，実装を簡素化するために調整され，一定に保たれている．hybridPC とは異なり，停止勾配は適用されない．学習の各試行において，ニューラル活動はまず勾配降下法（ニューラルダイナミクス）によって E を最小化するように更新される.

$$\frac{dx_l}{dt} = -\nabla_x E = -\epsilon_l^{gen} - \epsilon_l^{disc} + f'(x_l) \odot \left(W_l^\top \epsilon_{l-1}^{gen} + V_l^\top \epsilon_{l+1}^{disc} \right) \tag{6}$$

ここで,

$$\epsilon_l^{gen} := \alpha_{gen}(x_l - W_{l+1}f(x_{l+1})), \quad \epsilon_l^{disc} := \alpha_{disc}(x_l - V_{l-1}f(x_{l-1})) \tag{7}$$

はそれぞれ層 l のニューロンのトップダウンとボトムアップの予測誤差を表す．f' は関数 f の微分を表し，$\odot$ は要素ごとの積を表す．ニューラル活動を更新した後，重みは勾配降下法によって E を最小化するように更新される．

$$\Delta W_l \propto -\nabla_{W_l} E = \epsilon_{l-1}^{gen} f(x_l)^\top, \quad \Delta V_l \propto -\nabla_{V_l} E = \epsilon_{l+1}^{disc} f(x_l)^\top \tag{8}$$

このように，エネルギー関数に対してある層 l での学習を考えると，上下の層の予測誤差だけで学習を進めることができるので，局所的計算で学習を成り立たせることができる．

3.2 SpNCN

3.2.1 漏れ積分発火モデル

SpNCN のニューロンモデルには，漏れ積分発火モデル (Leaky Integrate-and-Fire, LIF) を使用している。このモデルは次の式で表される．

$$\tau_m \frac{\partial \mathbf{v}^l}{\partial t} = -\gamma_m \mathbf{v}^l(t) + R_m \mathbf{j}^l(t) \tag{9}$$

ここで、R_m は対応する膜抵抗（システム全体の任意の単一セル)、τ_m は膜時定数（具体的には $\tau_m = R_m C_m$ として設定され、C_m は膜容量)、γ_m は漏れの強さを制御する係数である。上記の式は、実際には、抵抗器から漏れが生じる単純な抵抗器-コンデンサー (RC) 回路を表している。電流 $\mathbf{J}^l(t)$ は、抵抗器と並列に配置されたコンデンサーによって時間とともに積分されていく。式を使用して漏れ積分ニューロンのダイナミクスをシミュレートするために、微分方程式をオイラー法で近似し、電圧 $\mathbf{v}^l(t)$ の値を計算する。

$$\mathbf{v}^l(t + \Delta t) = \mathbf{v}^l + \frac{\Delta t}{\tau_m}(-\gamma_m \mathbf{v}^l(t) + R_m \mathbf{j}^l(t)) \tag{10}$$

これに従って $\mathbf{v}^l(t)$ が増えていき，ある閾値 $\mathbf{v}_{thr}$ を超えるとシナプスが発火してデジタル値 1 が送られる．その後 $\mathbf{v}^l(t)$ はリセットされ，一定時間入力を拒否した後，元の挙動に戻る．

3.2.2 学習則

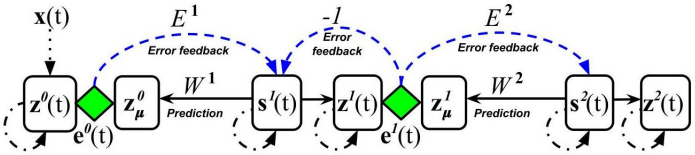

図 1　SpNCN のネットワーク例

LIF によって出力されたスパイク列 $s^l(t)$ は，トレース $z^l(t)$ に平滑化される．

$$z^l(t) = z^l(t) + \frac{\partial z^l(t)}{\partial t}, \text{ where, } \frac{\partial z^l(t)}{\partial t} = -\frac{z^l(t)}{\tau_{tr}} + s^l(t) \tag{11}$$

その後トップダウン予測の重み W^l を用いてトップダウンの予測を生成し，誤差を計算する

$$z_\mu^l = W^l \cdot s^{l-1}(t), \quad e^l(t) = (z^l(t) - z_\mu^l) \tag{12}$$

計算した誤差を用いて入力電流を表すと，以下の通りになる．

$$\frac{\partial J^l(t)}{\partial t} = -\frac{\kappa_J J^l(t)}{\tau_J} + \phi_e(-e^l(t) + E^l \cdot e^{l-1}(t)) \tag{13}$$

$$\frac{\partial J^L(t)}{\partial t} = -\frac{\kappa_J J^L(t)}{\tau_J} + \phi_e(E^L \cdot e^{L-1}(t)) \tag{14}$$

先の電圧と同じようにオイラー法で離散化し，計算できるようにする．

$$J^l(t+\Delta t) = J^l(t) + \frac{\Delta t}{\tau_J}\left(-\kappa_J J^l(t) + \phi_e(-e^l(t) + E^l \cdot e^{l-1}(t))\right) \tag{15}$$

$$J^L(t+\Delta t) = J^L(t) + \frac{\Delta t}{\tau_J}\left(-\kappa_J J^L(t) + \phi_e(E^L \cdot e^{L-1}(t))\right) \tag{16}$$

そしてトップダウンの予測重みと予測誤差のフィードバックの学習を勾配降下法で進める．

$$\Delta W^l = e^{l-1}(t) \cdot (s^l(t))^\top, \quad \Delta E^l = \beta(s^l(t) \cdot (e^{l-1}(t))) \tag{17}$$

3.3 統合

ボトムアップの重み V を追加し，誤差ニューロンを生成過程と識別過程に分ける．それに伴って誤差のフィードバックも 2 つに分かれる．

$$\mathbf{z}^l_{gen} = \mathbf{W}^{l+1} \cdot \mathbf{s}^{l+1}(t), \quad \mathbf{e}^l_{gen}(t) = \alpha_{gen}(\mathbf{z}^l(t) - \mathbf{z}^l_{gen}) \tag{18}$$

$$\mathbf{z}^l_{disc} = \mathbf{V}^{l-1} \cdot \mathbf{s}^{l-1}(t), \quad \mathbf{e}^l_{disc}(t) = \alpha_{disc}(\mathbf{z}^l(t) - \mathbf{z}^l_{disc}) \tag{19}$$

$$\Delta W^l = e^{l-1}_{gen}(t) \cdot (s^l(t))^\top, \quad \Delta E^l_{gen} = \beta(s^l(t) \cdot (e^{l-1}_{gen}(t))) \tag{20}$$

$$\Delta V^l = e^{l+1}_{disc}(t) \cdot (s^l(t))^\top, \quad \Delta E^l_{disc} = \beta(s^l(t) \cdot (e^{l+1}(t))) \tag{21}$$

$$\tau_j \frac{\partial \mathbf{j}^1(t)}{\partial t} = -\kappa_j \mathbf{j}^1(t) + \phi_e(-\mathbf{e}^1_{gen} + \mathbf{E}^1_{gen} \cdot \mathbf{e}^0_{gen}(t)) \tag{22}$$

$$\tau_j \frac{\partial \mathbf{j}^l(t)}{\partial t} = -\kappa_j \mathbf{j}^l(t) + \phi_e(-\mathbf{e}^l_{gen} - \mathbf{e}^l_{disc} + \mathbf{E}^l_{gen} \cdot \mathbf{e}^{l-1}_{gen}(t) + \mathbf{E}^l_{disc} \cdot \mathbf{e}^{l+1}_{disc}(t)) \tag{23}$$

$$\tag{24}$$

参考文献

[1] Antony W. N'dri, William Gebhardt, Céline Teulière, Fleur Zeldenrust, Rajesh P. N. Rao, Jochen Triesch, Alexander Ororbia. (2024): Predictive Coding with Spiking Neural Networks: a Survey, https://www.arxiv.org/abs/2409.05386.

[2] Karl Friston, James Kilner, Lee Harrison. (2006): A free energy principle for the brain, Journal of Physiology, Paris 100 (2006) p.70 – 87.

[3] Gaspard Oliviers, Mufeng Tang, Rafal Bogacz. (2025): Bidirectional predictive coding, https://arxiv.org/abs/2505.23415.

[4] Alexander Ororbia. (2023): Spiking neural predictive coding for continually learning from data streams, Neurocomputing Vol.544.