RF-004

時系列シンボルから頻出な部分列を抽出する ニューラルネットワークに関する一考察

A discussion on neural networks that extract frequent sub-sequences from time-series

森田 賢太† 高瀬 治彦‡ 森田 直樹†
Kenta Morita Haruhiko Takase Naoki Morita

1. はじめに

人工知能とはコンピュータ上などで人間の知能を再現しようとする技術である．文章の内容を理解させる自然言語処理技術や，収集されたデータの中から特徴などを抽出する機械学習などがある．

本研究の目的は，情報を取捨選択し，記憶するメカニズムをコンピュータ上で実現することである．具体的には，文献 1 のように子供が映画 1 本の中で頻度の高い語彙の学習を行うことの実現である．本研究はその第 1 歩として，ある期間に入力された情報の中から何度でも出現する情報を抽出するメカニズムをコンピュータ上で実現することである．

本研究に類似する，時系列信号から特定の信号を抽出する研究について述べる．特徴的なイベントの並びを注目時系列イベントパターンとして発見する方法[2]や，連続した入力パターンの順序関係を認識する神経回路モデル[3][4][5]などがある．我々の目的を詳しく述べると，絶え間なく入力される時系列シンボル列の中から，最近頻出の順序列パターンを抽出し，抽出した順序列を一定期間記憶できるようにすることである．

頻出な順序列パターンを抽出する方法に文献[2]の手法がある．この手法は，頻出な順序列の抽出を行う際には分析対象となるデータのすべての順序を知る必要がある．我々の目指すシステムは，時系列シンボルを絶え間なく入力し，頻出順序列の抽出を行うものであるため，入力の度に過去のデータ全体を使用して計算することが問題である．また，途切れることのない入力により,順序列を学習する方法として文献 3，文献 4 および文献 5 の手法がある．文献 3 の手法は，エルマンネットと呼ばれる単純再帰型ニューラルネットワークである．入力層の一部に中間層の状態を戻すことにより，過去の状態を入力とすることができる．この構造により，過去の入力履歴を保持する事ができるため，順序列を学習する事ができる．我々の目的は，最近という特定の範囲から頻出の順序列を抽出したいため，すべての過去の入力歴を保持するのは問題である．文献 5 の手法は，青木・青柳による連想記憶モデル[4]を参考にしたものである．これはニューラルネットワークを用いて現在の入力パターンとその前の入力パターンを連想記憶により記憶する方法である．この手法は連想記憶を用いているため，記憶できる順序列パターンの数に制限がある．

上記をふまえ本稿では，ニューラルネットワークを基にしたモデルを提案し，頻出な部分列を抽出することを試みる．このモデルでは，強化学習に基づいた学習を行うことで過去の入力履歴を保持しなくても良いようにし，ネットワークを自動で成長するようにすることで，記憶できるパターン数の制限を回避する．以下，２章では抽出したい頻出時系列シンボルについて説明し，３章でこれを自動抽出するためのモデルを提案する．４章で実験を通じ提案法の有効性を検討し，５章でまとめる．

†東海大学大学院　情報通信学研究科
‡三重大学大学院　工学研究科

2. 頻出時系列パターンの抽出

本章では，我々がめざす頻出時系列パターンの抽出について説明する．時系列シンボルとは，あらかじめ定められたシンボルを時間ごとに 1 つずつ順に並べた列である．この時系列シンボルを提示することで，頻出な部分列を抽出することを目的としている．ここで，頻出とは厳密な意味ではなく，最近頻出であることとし，部分列とはシンボルとその次のシンボルの時間が決まった間隔で出現したシンボル列である．たとえば図 1 は，時系列シンボルであり，下の数字は現在からの経過した秒数で，上の英字はそのときの出現シンボルである．図 1 において，１秒間隔でシンボル列を見ると，現在から 30 秒前の時間内では，X の次に 1 秒後に Y が出現することが 4 回ある．このようにある間隔で見たときに，X のあとに Y が出現する事を「XY」のシンボル列とする．X と Y の出現の間隔が 2 秒の物もあるが，これは，＿ (無入力)とのシンボル列である「X＿」や「＿Y」として捉えるため、別のシンボル列としてとらえる．図 1 の時系列シンボル列の中では，「YX」や「ZA」よりも「XY」のシンボル列が多く出現している．そこで，シンボル列を提示したときに，「XY」というシンボル列を抽出することが我々の目的である．

このような頻出時系列シンボル列の抽出にあたって，いくつか注意しなくてはならない点がある．1 つ目は抽出したいシンボル列の出現間隔が決まっていないことである．図 1 の例では，シンボル列「XY」ペアの間には，0 から 2 つの関係ないシンボル A や B などをはさんでいる．このような状況において，頻出のシンボル列を抽出するためには，抽出対象のシンボル列の頻度だけでなく，その「XY」のシンボル列の出現間隔が一定ではないため、どのシンボル列を学習するべきか捉えにくいことを考える必要がある．2 つ目は各シンボルの出現が決まった瞬間に来ないことである．たとえば，11 秒前のところを見ると，シンボル X は 11 秒ちょうどに入力されていない．このように各シンボルは決まった瞬間に入力されるとは限らない．

本稿ではこれら 2 つの注意点を考慮にいれて，頻出時系列シンボル列の抽出を目指す．第一歩として時系列の注意点である１つ目の抽出したいシンボル列の出現間隔が決まっていないことを特に考慮したうえで，空白で単語として

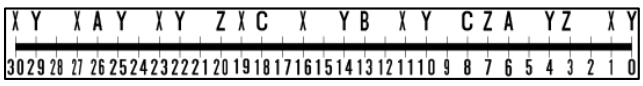

図 1　入力時系列シンボルの例

区切られた時系列シンボル列から頻出の単語の抽出を目指す．さらに簡単のため，次のように単語長は 2 シンボルに固定する．

…, ZA, , X, Y, , A, X, , Z, Y, , X, Y, , B, C, , X, Z, , X, Y, , C, A, , X, Y.

なお，頻度のもっとも高いものだけでなく，頻度が上位のシンボル列を指定数だけ抽出する．

3. 頻出時系列パターンの抽出

3.1 概要

頻出シンボル列の抽出を実現するために，時系列シンボルを入力として受け取り，頻出のシンボル列が入力されたときにのみ反応するニューラルネットワークを教師なし学習を用いて構築する．入力線はシンボルの種類数分用意し，シンボルの入力はシンボルに対応する入力線にのみ与える．出力は抽出したいシンボル列数分だけ用意し，それぞれ異なるシンボル列に対して反応するものとする．

図 2 は，シンボル列「XY」の抽出に成功したときの，入力と出力の関係を表したものである．図中左側は，「XY」の X を入力した時のようすであり，図中右側は，「XY」の Y を入力したようすである．シンボル X を入力した際には出力が行われず，X を入力した後にシンボル Y を入力すると出力が行われる．

このような状態を作るためのニューラルネットワークの構成，学習方法，出力ユニットについて説明する．

3.2 モデルの構成・動作

提案するモデルは，シンボルが入力された時間とその順序を保持するシンボル層と，頻出シンボル列を抽出する出力層の 2 層からなる．

シンボル層は各シンボルに対応する入力ユニットが複数ある．同じシンボルに対応する入力ユニットは一列に接続されており，その接続する個数は抽出したいシンボル列の長さに対応できる個数である．シンボルの入力は一方向からそのシンボルに対応するユニットに行い，信号の入力を受けたユニットは一定の時間間隔で後続のユニットにその信号を伝搬する．

シンボル層のすべての入力ユニットは出力層のすべての出力ユニットに接続しており， 信号の伝搬に一定時間の遅延をかける．入力ユニットの出力ユニットへの信号入力は，シンボル層のユニットに抽出したいシンボル列の長さと同じ個数のシンボルが入力されたときに行う．

具体的な動作例をシンボル列「XY」の抽出に成功したときの「XY」の入力で述べる．入力する「XY」のシンボル X と Y の間隔は 1 秒である．図 3 の場合は，シンボル 6 種，シンボル列の長さ 2 種分，抽出したいシンボルは 3 種類である．入力層の各行は上から X, Y, Z, A, B, C に対応している．シンボルの入力は左側から行い，シンボル層の伝搬にかかる時間は 1 秒である．またシンボル層から出力層の伝搬はシンボル層のユニットが 2 つ発火したときであり，伝搬にかかる時間は 1 秒である．また，「XY」の抽出に成功したユニットは 13 とする．図 3 の構造にて発火のようすを表す．図 4 は何も入力されていない状態である．この状態でシンボル X を入力すると，図 5 ように X に反応するユニット 1 が発火する．この場合シンボル層において発火しているユニットは 1 つのため，シンボル層は出力層に伝搬を行わない．図 6 はシンボル X を入力してから 1 秒後にシンボル Y を入力した時である．Y の入力により，Y に対応するユニット 3 が発火する．また，ユニット 2 はユニット 1 からの伝搬により発火する．シンボル層に入力されたシンボル数が X と Y の 2 個になったため，シンボル層から出力層へ信号が送られる．図 7 はシンボル列 XY の Y を入力してから 1 秒後のようすである．ユニット 13 はユニット 2 と 3 からの伝搬により発火する．ユニット 2 は「XY」の X に対応するユニットであり，ユニット 3 は「XY」の Y に対応するユニットである．このようにシンボル層がシンボル列に対応し，出力ユニットを発火させる状態になることで頻出のシンボル列がきたことがわかる．次節ではこの状態になる手法を述べる．

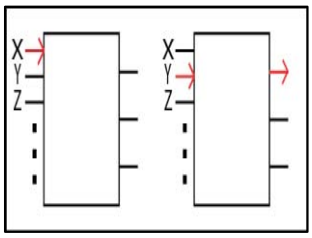

図 2 抽出成功時の入力

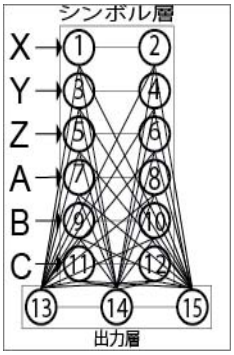

図 3 ネットワークの構成

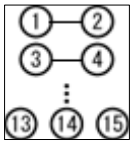

図 4 入力前

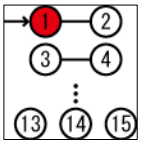

図 5 X を入力

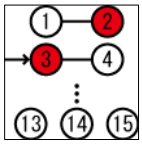

図 6 Y を入力

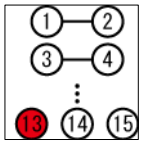

図 7 入力後

3.3 学習方法

3 章 2 節のような頻出シンボル列を抽出した状態を，ニューラルネットワークの強化学習に基づく自己組織化で実現する．強化学習の方法は，頻出パターンに対して強化を，そうでないパターンに対して減衰を行う．この方法により層間の結合荷重を調整することで，出力層の各ユニットが頻出のシンボル列の入力に呼応して反応する自己組織化を行う．

このネットワークを，ニューラルネットワークの代表的な強化学習の手法である Hebb 則[6]に基づいて学習する．本提案のモデルではこの Hebb 則をシンボル層と出力層間の結合に対して行う．Hebb 則では二つのユニット間の結合をそれらの発火時刻に基づいて調整する．前ユニット（シンボル層ユニット）が発火した後に後ユニット（出力層ユニット）が発火した場合，式(1)に従い，その結合荷重を一定値増やすことで強化する．

$$W \leftarrow W + A_+ \tag{1}$$

具体的な強化例を説明する．「XY」の入力により図 3 のユニット 2 とユニット 3 がほぼ同時に発火する．このシンボル層の発火により，出力ユニットであるユニット 13 が発火した場合，ユニット 13 につながるユニット 2 とユニット 3 の結合荷重が共に上がる．

本提案の学習方法は Hebb 則に 2 点工夫を加えることによ

り，頻出シンボル列を反応する自己組織化を実現することができる．１点目の工夫は結合荷重の強化の制限である．結合荷重に上限をつけ，結合荷重をそれ以上あげない．強化により結合荷重の上限を達したとき，他の同じ順序位置の入力ユニットと出力ユニットとの結合荷重を0にする.具体的には式(2)に従う．

$$\begin{cases} W \leftarrow W_{max} \ (\text{if } W_{max} < W) \\ W_{else} \leftarrow 0 \ \ (\text{if } W_{max} < W) \end{cases} \tag{2}$$

A+，Wmax は事前に決めるパラメータであり，A+は増加量，Wmax は結合荷重の上限となる．また，W は増加させる結合荷重であり，Welse は W の結合荷重が上限に達したときの他の同じ順序位置の結合過重である．例えば，ユニット 2 とユニット 13 の結合荷重が上限を超えた場合，ユニット 13 につながるユニット 4,6,8,10,12 の結合荷重が 0 に下がる．ある一つの出力ユニットからの結合荷重が上限に達した個数とシンボル列の長さの個数と一致したとき，シンボル列の抽出でき，そのシンボル列にのみ反応することができる．２点目は工夫を結合荷重の下げ方である．入力層ユニットが発火していない場合，以下の式(3)に従って指数関数的減衰を行う．

$$W \leftarrow W + A\_ \cdot e^{\frac{t}{\tau}} \tag{3}$$

A_は定数であり，τは学習則での時定数である．発火していない時間である t により減少量がかわり，発火していない時間が長いほど結合が下がる．

Hebb 則では出力ユニットが発火しないと結合荷重が下がらないため，まったく入力されていない入力ユニットの結合荷重は下がらない．よって結合荷重を指数関数的に減少することにより入力されてない入力ユニットの結合荷重を下げる事が出来る．

3.4 出力ユニット

出力層の各ユニットは，シンボル層にあるすべてのユニットからの信号を結合荷重で重み付けして受け取る．また，出力層内の各ユニットは互いに接続しており，側抑制を行う．本手法では，ユニットの発火判定・側抑制の発動は，常に出力層ユニットの番号順に行うこととする．従って，ユニット番号 n のユニットが発火することで，ユニット番号 n+1 以降のユニットに側抑制の刺激を与える．ユニットの発火動作は，Leaky Integrate-and-Fire (LIF)モデルに基づいており，式(4)に従う．

$$\begin{cases} V(t) \leftarrow V(t) + \left(Vreset - V(t)\right)e^{-\frac{t}{\tau m}} + Vinput \\ V(t) \leftarrow Vreset \qquad \left(if \ \theta < V(t)\right) \end{cases} \tag{4}$$

LIF は内部電位 V(t)を時間ごとに加算していき，閾値θを超えたときに発火する．LIF は単純に積分発火するだけでなく，内部電位の漏れもおこなう．Vinput は入力層からの信号に結合荷重を掛けたものの総和および側抑制からの入力である．Vreset，τm，θは，事前に決めるパラメータである．Vreset はユニットの内部電位の初期値である．τm は時定数でありこの数値が高いほどユニット内部の電圧が下がりにくい．θは閾値であり，ユニット内部の電圧が超えた場合，発火し溜まっていた電圧が下がり，ユニット内部の電圧は Vreset の値へともどる．

LIF モデルの特徴である積分発火と内部電位の漏れの量により，シンボルの入力がずれても，出力ユニットは発火することができる．積分発火により 0.9 秒後でも 1.1 秒後に入力されても，内部電位が下がりきらなければ，発火する事ができる.内部電位の漏れの量によりシンボルの入力のずれを対応することができる．しかし，内部電位の漏れは少なすぎるとシンボル列の抽出に失敗する.たとえば図 2 において，内部電位の漏れが少ないとき，「X○」と「○Y」のシンボル列が頻出だと，出力ユニットの発火により，入力ユニット 2 と 3 の結合荷重が上がる．このように出力ユニットの発火頻度が高いと，「XY」の頻度に関係なく「XY」に反応するネットワークになってしまう．これは内部電位の漏れが少ないと,閾値を超える事が多くなるため，「X○」と「○Y」の入力時に出力ユニットが発火する事が多いからである.よってシンボル列として抽出するには出力ユニットの発火頻度を押さえる必要がある.

発火頻度を押さえる方法としては即抑制を用いる事や結合荷重の上限を低く設定する方法などがある.しかし,即抑制は発火を押さえたいユニットではないユニットが発火する必要があり,どのように構成するかが難しい.また,結合荷重の上限を下げると,反応したいシンボル列の入力ユニットが反応しても内部電位が閾値を超えないため使う事ができない.そのため,ユニット自身に頻度を押さえる仕組みを与える.具体的には出力ユニットの時定数τm を小さくすることである.時定数τm を小さくすると,蓄積していた内部電位の減少量が増加する.従って,内部電位の漏れが多くなる.そのため,積分発火モデルである LIF モデルの出力ユニットは発火しにくくなる.よって,「X○」の X に反応することが減り,シンボル列を学習できる確率が上がる.しかし,時定数を下げすぎると内部電位の漏れが多いため,シンボルの時間のずれを吸収できなくなる可能性がある.よって時定数のパラメータを適切にする必要がある.

4. 頻出時系列の抽出の確認

この章では,提案手法により頻出の時系列パターンを抽出できるのかを,プログラムを用いた実験により確認する.

時系列シンボルの抽出において,シンボル列の出現間隔とシンボルの入力時間のずれの２点を注意する必要がある.入力時間のずれについては積分発火モデルの LIF により対応できる可能性が高い.そのためもう一つの注意点である,シンボル列の出現間隔に関係なく,頻出のシンボル列を抽出できるかを優先で確認する.

4.1 確認条件・方法

実験では，6 種のシンボル(A, B, C, D, E, F)を使用し，これを 2 シンボル組み合わせた部分シンボル列を 36 種類(AA, AB, AC, ・・・,FE,FF)用意した．これらの部分シンボル列を，空白(無入力)で区切り，ランダムな順序(使用回数は問わない)で連結することで，システムに提示する時系列シンボル列を作成した．これを，1 秒に 1 シンボルずつネットワークへ入力する．時系列シンボル列を作成する際に，各部分シンボル列の生成確率を変化させることで,各部分シンボル列の頻度を調整できる．ネットワークは，6 入力 3 出力として，シンボル層ユニットは合わせて 12 個用意した．これにより，6種のシンボルについて，2秒分の入力をシンボル層で保持できる．入力ユニットから出力ユニットへの伝搬の時間は 1 秒の遅れとした． 結合荷重の初期値と出力層ユニットの各パラメータは，表 1 とした．ま

表 1 ネットワークのパラメータ　表 2 学習のパラメータ

結合荷重の初期値	15
Vreset	-68mV
θ	-10mV
τ m	5 秒

Wmax	30
A+	1.0
A-	0.01
τ	10 秒

た学習の各パラメータは表 2 とした．また，これらのパラメータは試行錯誤で決定した．実験は,提示するシンボル列の出現割合を変化させ提示を行った.

4.2 頻出シンボル列の抽出確認

この節では，提案手法により最近の頻出部分シンボル列を正しく，指定数分だけ抽出できるかどうかを確認する.

実験は，ネットワークに空白も含めて 20 分間分のシンボル列を提示しながら，各出力ユニットが反応する部分シンボル列を調査した．その結果を表 3 に示す．この表は，入力として与えたシンボル列を生成した際の各部分シンボル列の生起確率と，それを学習したネットワークの 3 つの出力ユニットが反応するようになった部分シンボル列を示したものである．シンボル列の右の数字はそのシンボル列のみに反応できるようになった状態の開始からの秒数である．以下，各ケースについて詳細に分析する.

1 番目，2 番目のケースでは，高い頻度の 3 種の部分シンボル列をすべて抽出できた.

3 番目のケースでは，ケース 1 よりもユニット番号 1 番のユニットと 2 番のユニットの抽出が早かったが,3 位の EF について抽出できなかった．提案手法では，側抑制の関係から，ユニット番号 1 番のユニットと 2 番のユニットのどちらかが発火したとき，3 番ユニットは反応する事が出来ない.このケースでは頻出シンボル列の 2 つの割合が多すぎるため,三番目のユニットへの結合荷重の減衰に増加が間に合わず,抽出に失敗した.

4 番目のケースでは，特に頻度の高い AB は正しく抽出できた．もう一つ抽出した DA は，実験の設定では特に頻度が高くなかったが，乱数の関係で，たまたま抽出されたと考えられる.

以上より，提案法によりシンボル列の出現間隔にかかわらず，頻度が高いシンボル列を抽出できることが確認された．なお，シンボル数・抽出シンボル列数・シンボルの入力時刻の揺らぎについては，実験しておらず，今後の検討が必要である．しかし，以下に示すように，いずれも深刻な問題は引き起こさないと考える．まず，シンボル数・抽出シンボル列数については，それぞれシンボル層・出力層のユニットを増やすことで容易に対応できる．シンボルの入力時刻の揺らぎについては，３章４節で検討したように対応できる.

4.3 シンボル列の確認

次に 3 章 4 節で述べたように,出力ユニットが頻繁に発火する場合,「A○」が頻出であった場合,A の第一順序位置に対応する入力ユニットの結合荷重が増加していく.これにより「A○」と「○B」が頻出であった場合,出力ユニットが「AB」を学習してしまう恐れがある.そこで先ほどのパラメータを用いて,出力ユニットの発火を押さえることで,「AB」を学習しないかを確認するため,「AB」「AC」や「BA」「BC」など最初のシンボル,もしくは後のシンボルが同じシンボル列が頻出である時の結果を表 3 に示す.これにより,「A○」と「○Ｂ」が頻出であるとき,入力データにはない「AB」にのみ反応するユニットは作成されていないため,頻出シンボルによる頻出シンボル列の抽出への影響はないと考えられる.

表３　頻出シンボルの抽出確認の結果

	入力シンボル列	出力ユニット		
		1 番目	2 番目	3 番目
1	「AB」10.0% 「CD」10.0% 「EF」10.0% 他各 2.1%	EF(359)	AB(728)	CD(1190)
2	「AB」10.0% 「CB」10.0% 「EF」10.0% 他各 2.1%	AB(326)	CB(509)	EF(1070)
3	「AB」42.0% 「CD」42.0% 「EF」16.0% 他各 0.0%	AB(161)	CD(281)	なし
4	「AB」10.0% 他各 2.5%	AB(458)	DA(947)	なし

表４　シンボル列の確認

入力シンボル列	出力ユニット		
	1 番目	2 番目	3 番目
「A○」15.0% 「○B」15.0% 「AB」0.0% 他各 2.8%	AA(353)	BE(773)	なし

4.4 環境の変化による影響

次に,学習済みのユニットがある状態でシンボル列の出現率を変化させてから,シンボル列を提示させた結果を述べる．シンボル列の出現率を「AB」20%「CD」10%他各 2.0%にして提示を行った.まず,第一出力ユニットで「AB」を学習し抽出済みである状態にした.その後,学習済みである「AB」の出現率を 0%に変更した.「AB」0%「CD」10%「EF」10%他 2.4%でシンボルを提示させた.「AB」の出現率を無くし,「EF」の出現率を上げて提示をした場合,出現しなくなった第一ユニットの結合荷重はすべて 0 に下がり,すべてのシンボル列に発火しなくなった.他の出力ユニットにおいては，変更前から頻出である「CD」の抽出に成功し,途中から出現率を上げた「EF」においても抽出することができた.

5. まとめ

本稿では，絶え間なく入力される時系列シンボルの中から最近頻出の順序シンボル列パターンを抽出できることを目的とし，抽出したいシンボル列のシンボルの出現間隔とシンボルの入力のずれを考慮した頻出シンボルを抽出する

LIF モデルを用いたニューラルネットワークの構造と学習方法を提案した．

シンボルの入力間隔が一定の 2 シンボル長のシンボル列に対して，本システムが出現頻度の高いシンボル列を抽出できることを実験により確認した．

課題としては，シンボルの入力のずれに対応できるパラメータを見つけることである．

参考文献

[1] 古樋 直己，“偶発的語彙習得と英語力、語の頻度、コンテキストの関係：英語字幕付き邦画を用いた場合”，映画英語教育研究：紀要，Vol.14，pp.29-40，2009.

[2] Shigeaki Sakurai, Minoru Nishizawa, “Discovery of Various Sequential Patterns within Top-*k* form Sequential Data”, Proceedings of 2015 SCIS&ISIS，pp．446-451，2014.

[3] ELMAN，J.L，“Finding structure in time”，Cognitive Science，Vol.14，pp．179-211，1990.

[4] Takaaki Aoki，Toshio Aoyagi，“A Possible Role of Incoming Spike Synchrony in Associative Memory Model with STDP Learning Rule”, Progress of Theoretical Physics Supplement, No.161，pp．152-155，2006.

[5] 田中 一穂，矢野 慎一郎，山本 野人，“連続した入力パタンのあいだの順序関係を認識する神経回路モデル—情報の予測・抽象化に向けて”，日本応用数理学会論文誌，Vol.18，No.1，pp．87-105，2008.

[6] 荻原 直洋，塚田 稔，合原 一幸，“文脈構造の記憶の書き込みと情報表現”，電子情報通信学会技術研究報告，NC96-179，pp．183-188，1997.