

修士論文

頻出な部分列を抽出する  
スパイキングニューラルネットワーク  
—間隔にゆらぎをもつシンボル列の扱い—

平成30年度修了

三重大学大学院工学研究科

博士前期課程 電気電子工学専攻

澤村 将周

# 目次

第1章	はじめに	1
1.1	音声認識システムの構築	1
1.2	単語の抽出	2
1.3	ゆらぎを持つシンボル列からの頻出な部分列の抽出	3
第2章	スパイクニューラルネットワークを用いた手法	5
2.1	ネットワークの構成と動作	5
2.2	学習方法	10
2.3	ユニットの動作	11
第3章	間隔にゆらぎをもつシンボル列を扱う手法	13
3.1	間隔にゆらぎを持つシンボル列を扱う際の問題点	13
3.1.1	一定の入力間隔を前提としたネットワーク	13
3.1.2	間隔にゆらぎをもつ入力を扱う際に適さない学習法	15
3.2	提案手法	18
3.2.1	入力の間隔に適応する方法の提案	18
3.2.2	間隔のゆらぎを考慮した学習法の提案	19
第4章	実験	21
4.1	実験方法・条件	21
4.2	実験結果・考察	24
第5章	結論	26
	謝辞	27
	参考文献	28



# 目 次

1.1	日本語音声データの一例 . . . . .	3
2.1	ネットワークの構成 . . . . .	6
2.2	「ほん」の抽出が完了したネットワーク . . . . .	7
2.3	使用したデータの一例 . . . . .	8
2.4	「ほ」が入力されたタイミング . . . . .	8
2.5	「ん」が入力されたタイミング . . . . .	9
2.6	「で」が入力されたタイミング . . . . .	9
3.1	ゆらぎの無い場合の出力ユニットの内部電位 . . . . .	14
3.2	ゆらぎをもつ場合の出力ユニットの内部電位 . . . . .	14
3.3	頻出でない部分列を抽出した際の結合荷重の変化 . . . . .	16
3.4	ゆらぎの無いシンボル列から抽出を行った場合の結合荷重の変化 . . . . .	17
3.5	従来手法と提案手法の窓関数の比較 . . . . .	20
4.1	結合荷重の変化 . . . . .	25

# 表 目 次

4.1	シンボルの長さ 2 の部分列の出現頻度 . . . . .	22
4.2	ネットワークの各パラメータ . . . . .	22
4.3	提案手法での学習の各パラメータ . . . . .	23
4.4	既存手法での学習の各パラメータ . . . . .	23
4.5	提案手法によって抽出された部分列 . . . . .	24
4.6	既存手法によって抽出された部分列 . . . . .	24

# 第1章 はじめに

## 1.1 音声認識システムの構築

今日、情報処理が担う役割は大きなものとなっており、情報処理技術のさらなる発展が期待されている。情報処理技術の中でも音声処理技術の発展はめざましく、音声合成技術を用いた音声ガイダンス付きカーナビゲーションシステム、音声認識技術を用いた音声ダイヤル機能付き携帯電話などさまざまな製品が開発されるようになってきた。近年、高品質な音声合成や高精度な音声認識を実現するために必要な様々な技術の研究開発が行われている [1-6]。

音声認識は、機械に音声を与え、機械に音声の内容などを認識させることである。音声の内容を認識させるには自然言語処理技術を用いており、その自然言語処理技術には言語の最小単位である単語をシステムが自動で抽出する必要がある。既存の音声認識では、識別器を用いて単語を抽出している。この識別器は、人間が単語のデータベースを用意し、そのデータベースを識別器に学習させることで作成している。

より高精度な音声認識を実現させるためには、膨大な数の単語を学習させる必要がある [7]。しかし、このような単語のデータベースの構築を手動で行うには数が膨大で、大きな労力が必要となる。また、新たな単語を抽出させるためにはシステムに再学習させる必要があり、それも大きな労力が必要である。古樋は、実際の人間の子供に日本語音声かつ英語字幕の映画を見せ、偶発的語彙力習得について検証をした [8]。検証の結果、1本の中で頻度の高い単語の学習をする傾向があることが分かった。このような、人間の子供が音声を聴いているうちに頻出な単語を覚えるようなことを、機械ができることが望ましい。そこで、本研究は音声データから頻出な単語を自動的に取り出すことを目標とする。

## 1.2 単語の抽出

音声データはシンボル列（時系列データ）であり，単語はシンボル（音素）で構成された部分列である．そこでシンボル列から頻出な部分列を抽出し，抽出された部分列が単語かどうか判定することで頻出な単語を取り出すことをめざす．

音声データから頻出な単語を自動的に取り出すには，シンボル列から頻出な部分列を抽出する際に以下の3つの課題を解決する必要がある．

条件1 ストリーミング形式で入出力を行い，シンボルが与えられるたびに学習する．

条件2 さまざまな長さの部分列を複数個学習する．

条件3 音声データがもつ特徴である間隔のゆらぎへ対応する．

本研究は，子供が単語を覚えるように，音声データから頻出な単語を自動的に取り出すことをめざしている．そのため，人間が音声を聞くように機械にストリーミング形式で音声データを与え，機械が学習と同時に部分列を抽出する条件1が必要である．また条件2は，取り出される単語の種類は多く，長さはさまざまであるので，抽出する部分列の数に制限がない必要がある．そして条件3は，音声データの各シンボルの出現間隔は一定ではないため，それに対応する必要がある．図1.1は日本語の音声データの一例を示している．横軸は時間を意味し，各要素の出現タイミングは水平軸上の黒点で表している．このように音声データの各シンボルは一定の間隔から少しずれて出現する．日本語の場合，平均発話間隔はひらがな1文字単位の間隔で145msであり，標準偏差27.8msのゆらぎをもっている [9]．日本語の単語はシンボルの間隔によって意味が異なる場合があり，シンボルの間隔が重要である．そのため，このシンボルの間隔を考慮し，間隔のゆらぎに影響を受けない処理が必要である．

シンボル列から頻出な部分列を抽出する手法がいくつか提案されている [10–14]．桜井らの手法は Apriori アルゴリズムを拡張した手法 [10] で，類似した部分列を頻出な部分列として上位  $k$  個を抽出することができる．この手法の場合，類似した部分列を抽出する際，対象となるデータの要素の個数や順序をあらかじめ知る必要がある．よって，ストリーミング形式で入力を行う場合に，対象となるデータの

要素の個数や順序を入力時に全て把握することはできないため、条件1に対応することが難しいと考えられる。他の手法として、シンボル列の順序関係を学習する神経回路モデル [11–14] などがある。文献 [12] の田中らの手法は、青木・青柳による連想記憶モデル [13] を参考に基づいたものである。この連想記憶は記憶できるパターン数に制限がある。よって、条件2のさまざまなパターンの部分列を学習することが困難である。文献 [14] の森田らの手法はスパイクニューラルネットワークを用いた教師なし学習に基づいた手法である。ニューラルネットワークを用いることによって過去の入力履歴を保持しなくても良いため、条件1のストリーミング形式の入出力に対応することができる。さらに、ネットワークを自動で成長させることで、条件2のさまざまな長さの部分列を複数個学習することができる。しかし、森田らの手法は、入力シンボルの出現間隔は一定であることを前提としていたため、条件3の音声データがもつ特徴である間隔のゆらぎへ対応することができていない。

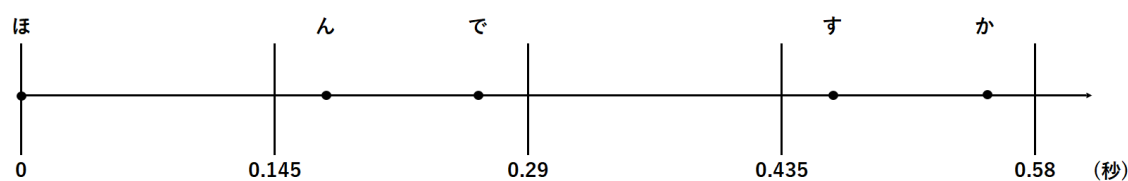


図 1.1: 日本語音声データの一例

### 1.3 ゆらぎを持つシンボル列からの頻出な部分列の抽出

本稿では、音声データから頻出な単語を取り出すための第1歩として、音声データの特徴をもつシンボル列から頻出な部分列の抽出を目的とする。具体的には日本語の音声データから音素を取り出し、それぞれの音素を出現タイミングどおりにストリーミング形式でシステムに与え、システムが頻出な部分列を抽出を行う。目的に対して、本稿ではスパイクニューラルネットワークを用いた手法 [14] を拡張することで取り組む。この手法は、前節で述べた3つの条件のうち2つを解決しているため、本稿では出現間隔のゆらぎに焦点を当てる。そのため、本稿



では抽出したい部分列は一つとし，その部分列の長さは2と限定し，議論を単純化する．

以下に本稿の構成を示す．第2章ではスパイキングニューラルネットワークを用いた手法について述べる．第3章では間隔にゆらぎをもつシンボル列から頻出な部分列を抽出する手法の提案を行う．第4章では提案した手法についての有効性を議論する．第5章では本研究についてまとめる．

## 第2章 スパイキングニューラルネットワークを用いた手法

この章は拡張対象である文献 [14] で提案された既存手法について説明する。このモデルは、スパイキングニューラルネットワークと自己組織化を組み合わせた手法である。ネットワークは入力としてシンボルを一定間隔で一つずつ受け取る。初期段階では、出力層のユニットはランダムに発火するが、シンボルを与えるたびに教師なし学習をして、頻出なシンボルの部分列が入力されたときに特定のユニットが一つだけ発火するようになる。

2.1節では、ネットワークの構成と動作について、2.2節では学習法について、2.3節ではユニットの動作について述べる。

### 2.1 ネットワークの構成と動作

ネットワークは、シンボルブロックと出力ブロックの2つのブロックで構成されている。図2.1はネットワークの構成図を表している。図2.1はネットワークの構成図を表している。シンボルブロックには2つのユニットの層があり、出力ブロックには複数のユニットの層がある。

シンボルブロックには各シンボルに対応する入力線がシンボルの種類と同じ数ある。この入力線は二つのユニットと鎖状に接続されている。入力線にシンボルが与えられると、信号は接続しているユニットに伝搬される。信号を受け取ったユニットは発火し、接続しているユニットへと信号を伝搬させる。信号の伝搬は入力側からの一方向であり、各伝搬には一定の時間がかかる。

シンボルブロックのユニットは出力ブロックのユニットへと接続している。出力ブロックの各ユニットは、シンボルブロックの1層目と出力ブロックの一つ前の層のユニットと全結合している。ただし、出力ブロックの1層目は、シンボル

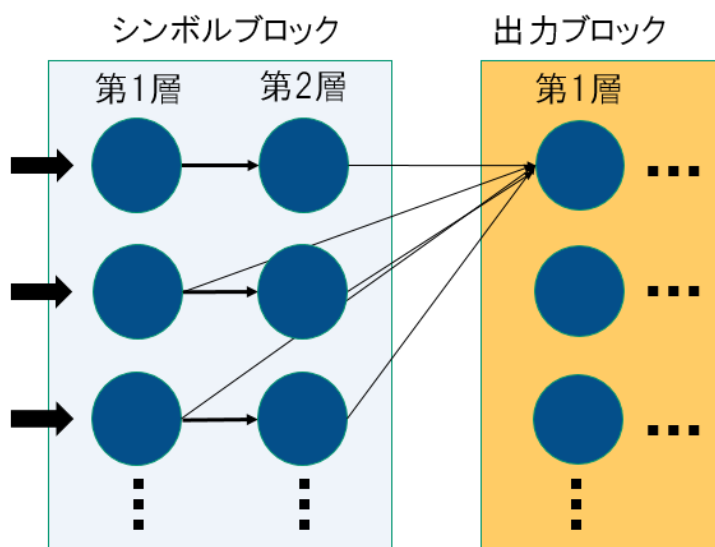


図 2.1: ネットワークの構成

ブロックの1層目と2層目と接続している。シンボルブロックのユニット間の伝搬同様に、出力ブロックのユニットへの信号の伝搬にも一定の時間がかかる。

出力ブロックのユニットへの接続は結合荷重によって重みづけされ、初期状態では全て同じ値になっている。また、初期のネットワークの出力ブロックのユニットは、さまざまな部分列に対して入力されるシンボル列に依存した発火をする。シンボルがネットワークに与え続けられると、学習により頻出な部分列が与えられたときに特定のユニットのみが発火するようになり、ネットワークが頻出な部分列を抽出した状態となる。その状態になったときの出力ブロックのユニットは結合荷重が最大値に達している2つの結合から接続されている。この2つの結合の接続元は、異なる層から1つずつ選ばれる。

図 2.2 は頻出シンボル「ほん」を抽出できるように学習されたネットワークを示している。具体的に図 2.2 に図 2.3 のシンボル列の一部「ほんで」が順に入力された場合の、ネットワークの動作を説明する。図 2.4～図 2.6 はシンボル列「ほんで」が順に入力されたネットワークの状態を示している。シンボルブロックの各行のユニットは「ほ」、「ん」、「で」の各シンボルに対応している。シンボルの入力は左

側から行われ、シンボルブロック内における信号の伝搬されるタイミングは入力  
のタイミングと同時に一定の間隔で行われる。シンボルブロックから出力ブロッ  
クへと信号が伝搬されるのは、シンボルブロックのユニットが2つ発火したときで  
あり、伝搬のタイミングは入力のタイミングと同時に行われる。また出力ブロッ  
ク1層目のユニットは上から順に抽出が完了した「ほん」、「です」に対応したユ  
ニットとなっている。図2.4は最初のシンボルである「ほ」が入力された時、シン  
ボルブロック1層目の「ほ」に対応するユニットが入力情報を保持する。次に図  
2.5の「ん」が入力された時、シンボルブロックの第1層の「ん」に対応するユニ  
ットに入力情報が保持される。さらに、シンボルブロックの第1層に保持されてい  
た「ほ」の入力情報の信号は同じタイミングでシンボルブロックの第2層の「ほ」  
に対応するユニットに伝搬される。そしてシンボルブロックの「ほ」、「ん」のユニ  
ットが保持する入力情報が出力ブロックの「ほん」のユニットへ信号が伝搬され、次  
に図2.6の「で」が入力されたタイミングで発火する。このように出力ブロックの  
ユニットが発火する状態となることで頻出な部分列が入力されたことがわかる。

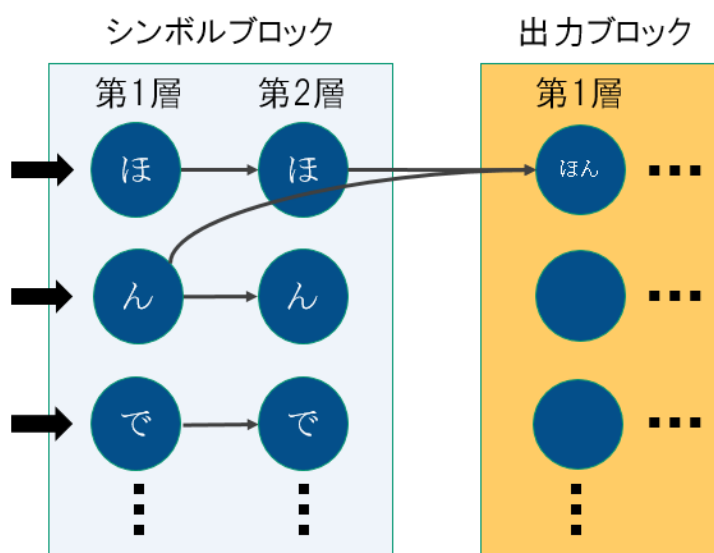


図 2.2: 「ほん」の抽出が完了したネットワーク

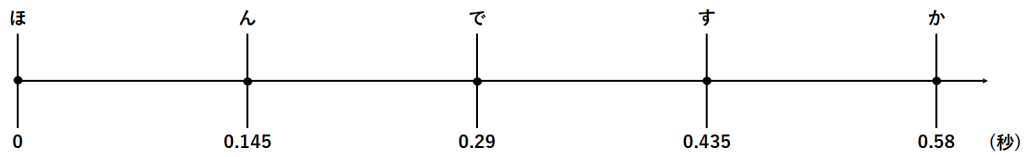


図 2.3: 使用したデータの一例

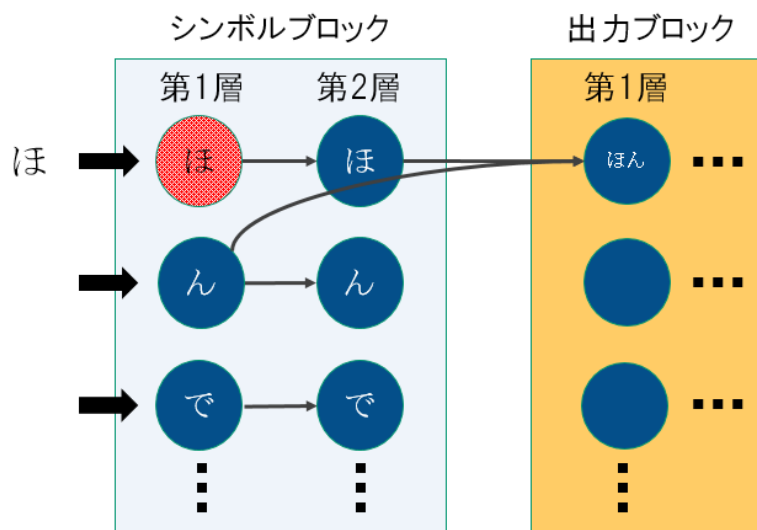


図 2.4: 「ほ」が入力されたタイミング

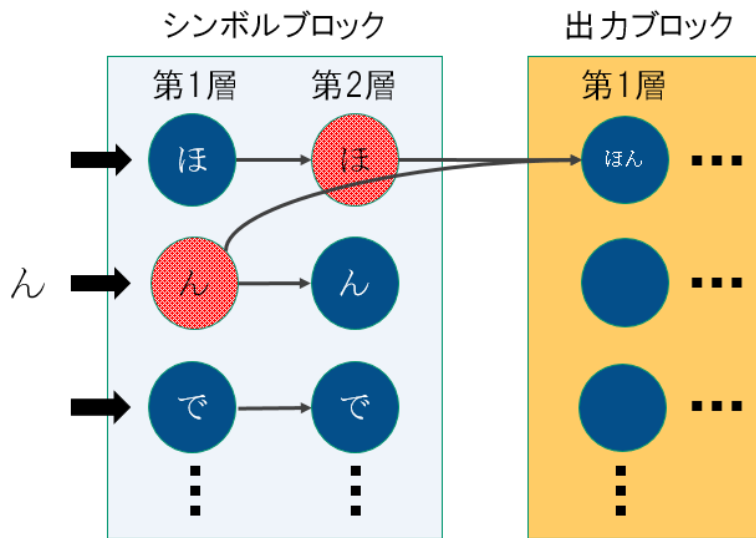


図 2.5: 「ん」が入力されたタイミング

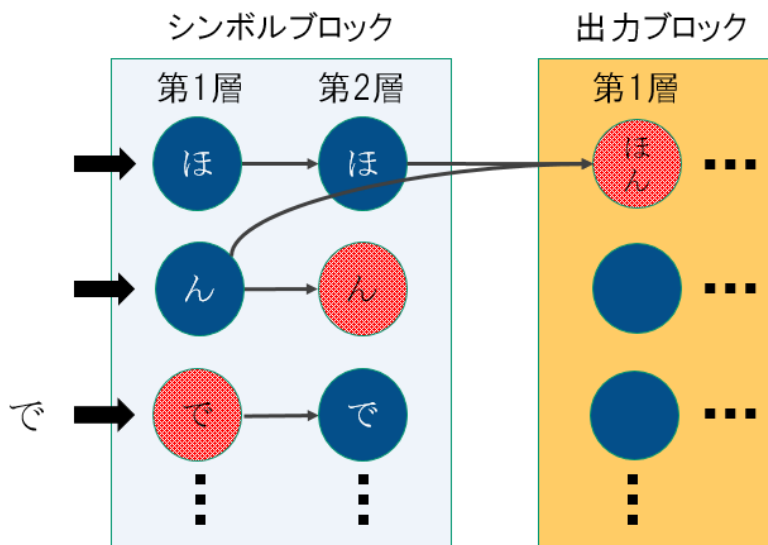


図 2.6: 「で」が入力されたタイミング

## 2.2 学習方法

このネットワークは教師なし学習により，頻出部分列が与えられたときにのみ発火するユニットが自動的に作成される．学習は出力ブロックの各ユニットに接続する結合荷重すべてに対し行われる．結合荷重の値は，教師なし学習である Spike-timing dependent synaptic plasticity (STDP) [16] に基づいた学習によって調整される．STDP 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方法である．接続元のユニットが発火した後に接続先のユニットが発火した場合，その結合荷重の値は増加する．一方，接続先のユニットが発火した後に接続元のユニットが発火した場合，結合荷重の値は減少する．結合荷重の増減量は，ユニットの発火時刻の差で決定され，式 (2.1) に従う．

$$W_{ij} \leftarrow \begin{cases} W_{ij} + A_+ \exp\left(\frac{-x}{\tau_s}\right) & (\text{if } x > 0) \\ W_{ij} - A_- \exp\left(\frac{x}{\tau_s}\right) & (\text{if } x < 0) \\ W_{ij} & (\text{if } x = 0) \end{cases} \quad (2.1)$$

$W_{ij}$  がユニット  $j$  から  $i$  に接続するユニットの結合荷重の値を示している． $x$  はユニット  $j$  が発火してからユニット  $i$  が発火するまでの経過時間である． $A_+$ ， $A_-$  は定数で， $\tau_s$  は時定数であり， $\tau_s$  が大きいほど結合荷重の変化は急激になる．

既存手法 [14] の学習は，頻出部分列を抽出するために，STDP に 2 点の工夫を加えている．1 点目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ようにしている．結合荷重が上限に達したとき，式 (2.2) に従う．

$$\begin{cases} W_{ij} \leftarrow W_{max} & (\text{if } W_{max} < W_{ij}) \\ W_{else} \leftarrow 0 & (\text{if } W_{max} < W_{ij}) \end{cases} \quad (2.2)$$

$W_{max}$  は事前に決めるパラメータであり結合荷重の上限となる。学習により結合荷重の値が上限に達したとき、結合荷重の接続元の層において、その結合荷重以外の接続先のユニットへ接続している結合荷重の値を 0 にする。この工夫によって変更された結合荷重はその後学習により値を変化させない。

2 点目は結合荷重の下げ方である。接続元のユニットが発火していない場合、以下の式 (2.3) に従って指数関数的減衰を行う。

$$W_{ij} = W_{ij} - D_- \exp\left(-\frac{t}{\tau}\right) \quad (2.3)$$

$D_-$  は定数であり、 $\tau$  は時間的減衰での時定数である。発火していない時間である  $t$  により減少量が変わり、発火していない時間が長いほど結合が下がる。

## 2.3 ユニットの動作

シンボルブロックにユニットは信号を受け取ると必ず発火をするが、出力ブロックのユニットは Leaky Integrate-and-Fire (LIF) モデル [15] に基づいて発火する。発火動作は、式 (2.4)、式 (2.5) に従う。

$$I_t = \sum_j W_{ij} V_i(t) \quad (2.4)$$

$$\begin{cases} V_i(t) \leftarrow V_i(t) + (V_r - T_i(t)) \exp\left(-\frac{t}{\tau_r}\right) + I_t(t) \\ V_i(t) \leftarrow V_r \text{ (if } \theta < V_i(t)) \end{cases} \quad (2.5)$$

LIF は内部電位  $V_i(t)$  を時間ごとに加算していき、しきい値  $\theta$  を超えたときに発火する。LIF は入力刺激を積分し発火する様子だけでなく、内部電位の漏れによ



る減衰も定式化している。  $V_i(t)$  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。  $V_r$ ,  $\tau_r$ ,  $\theta$  は、事前に決めるパラメータである。  $V_r$  はユニットの内部電位の初期値である。  $\tau_r$  は時定数でありこの数値が高いほどユニットの内部電圧が下がりにくい。  $\theta$  はしきい値であり、ユニット内部の電圧が超えた時に発火し、溜まっていた電圧が下がりユニット内部の電圧は  $V_r$  の値へともどる。

出力ブロックのユニットは各層で側抑制で接続されている。出力ブロックのユニットが発火した場合、同じ層で他のユニットが発火しないように、その時刻に出力ブロックへ与えられる信号はすべてカットされる。

# 第3章 間隔にゆらぎをもつシンボル列を扱う手法

本章では，既存の手法を拡張して，間隔にゆらぎを持つシンボル列を扱う際に起こる問題点を2点述べる．そして，それぞれの問題点について対応する手法を提案する．

## 3.1 間隔にゆらぎを持つシンボル列を扱う際の問題点

### 3.1.1 一定の入力間隔を前提としたネットワーク

既存手法では，図2.3に示すようなシンボル列を対象としており，ニューラルネットワークに与えるシンボル列のシンボルの出現間隔は一定であることを前提としている．そして，シンボルが与えられる間隔は一定であり，ネットワークの伝搬時間はその一定間隔と同等になっている．そのため，シンボルブロックのユニットから出力ブロックのユニットへの入力は，常に2つの信号が同時に与えられる．

音声データの各シンボルの出現間隔は一定でないため，このニューラルネットワークにシンボルをそのまま与えた場合，シンボルブロックのユニットからの出力ブロックへの入力は，同時に2つの信号が与えられるとは限らない．既存手法では，2つの信号が同時に与えられることを想定しているため，図3.1のように同時に2つの信号が与えられ出力ユニットの内部電位がしきい値を超える．しかし間隔にゆらぎをもつ場合，図3.2のように，2つの信号がずれて伝搬され，出力ユニットの内部電位がしきい値を超えることができず，発火することができないことが考えられる．

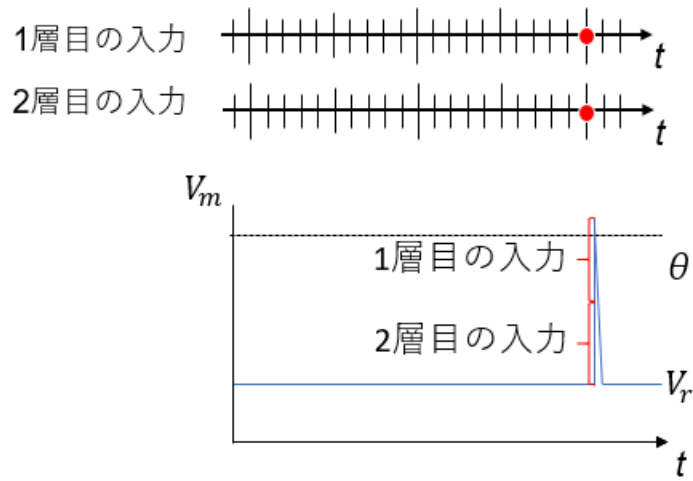


図 3.1: ゆらぎの無い場合の出力ユニットの内部電位

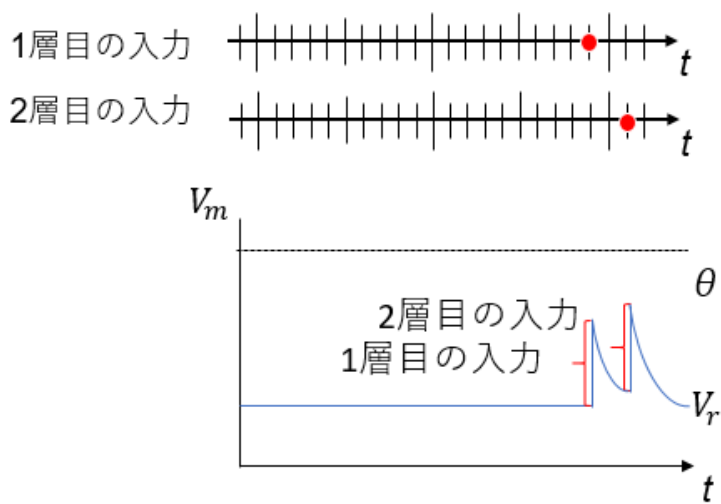


図 3.2: ゆらぎをもつ場合の出力ユニットの内部電位

### 3.1.2 間隔にゆらぎをもつ入力を扱う際に適さない学習法

2.2節で述べたとおり，既存手法はSTDPに基づいた学習を用いている．STDPは，接続元と接続先のユニットのそれぞれの発火タイミングに基づいて，結合荷重の値を調整する．この調整による値の増減量は，接続元と接続先のユニットの発火タイミングの差により決定される．

既存手法の場合，出力ブロックのユニットへ接続しているユニットは同時に発火しているため，増減量は一緒である．しかし，シンボルの出現間隔が一定でない場合，その接続元のユニットの発火は同時でないため，それぞれの結合間の発火タイミングの差は同じではなく，結合荷重の値の変化量は同じでない．STDPの窓関数は指数関数的に単調増加・減少しているため，発火タイミングが少しずれた場合も変化量が大きく異なってしまう．

図3.3は発火タイミングのずれによって頻出では無い部分列が抽出された場合に接続している2つのユニットの結合荷重の変化を表している．このように抽出された部分列に接続している結合荷重の変化量が大きくずれてしまい異なるタイミングで最大値に到達している．その結果頻出では無い部分列を抽出している．間隔にゆらぎの無いシンボル列から抽出を行った場合，図3.4のようなずれの無い変化となる．よって，発火タイミングが少しだけずれた場合も，変化量がほぼ同じになることが望ましい．

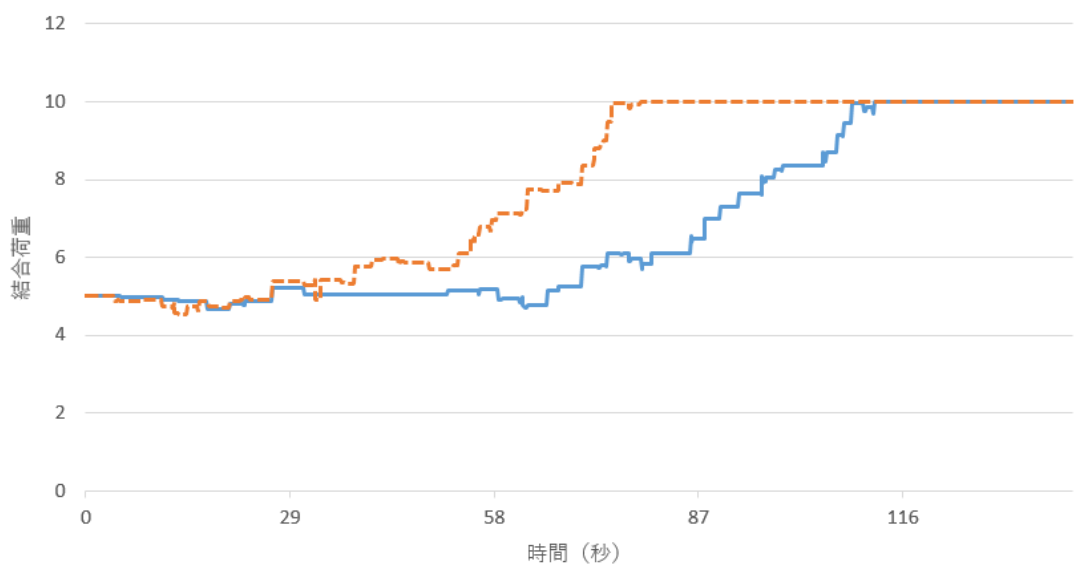


図 3.3: 頻出でない部分列を抽出した際の結合荷重の変化

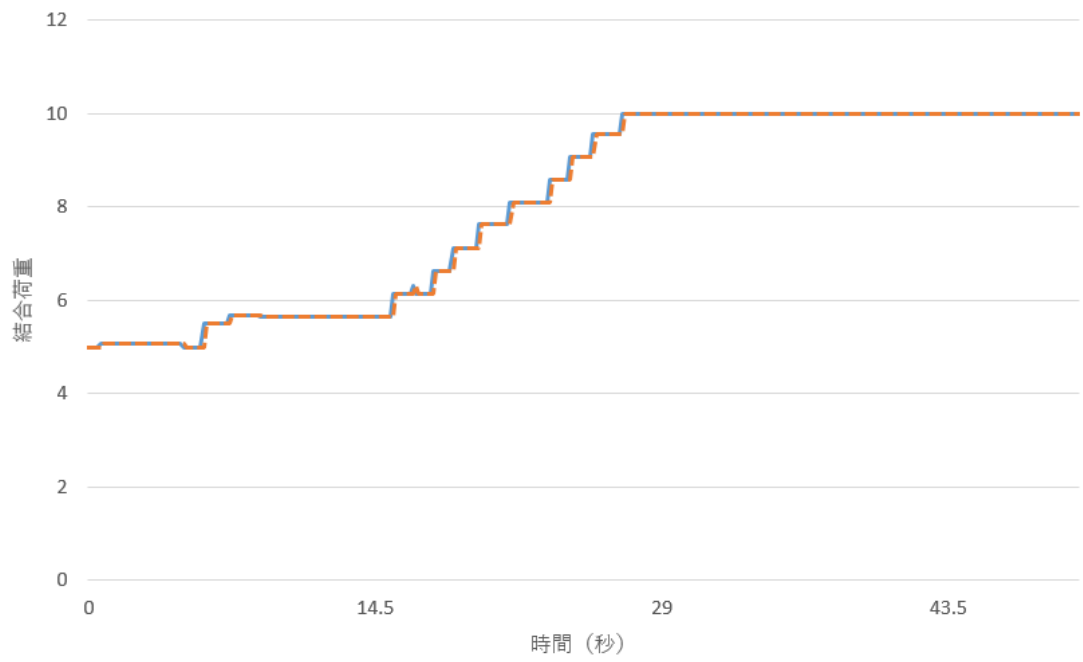


図 3.4: ゆらぎの無いシンボル列から抽出を行った場合の結合荷重の変化

## 3.2 提案手法

3.1.1 節と 3.1.2 節で議論したように、既存手法では一定の入力間隔を前提としている点と学習法が間隔のゆらぎによる影響を考慮していないという点の 2 つの問題点がある。本節では、これらの 2 点についてそれぞれに対する手法を提案する。

### 3.2.1 入力の間隔に適応する方法の提案

この節では、3.1.1 項で挙げたシンボルの入力タイミングと信号の伝搬するタイミングを柔軟に設定する方法について議論する。間隔にゆらぎを持つシンボル列を扱う際、入力のタイミングは一定の間隔とは限らない。そこで、単純・確実な方法として、入力のタイミングと信号の伝搬するタイミングをそれぞれ任意に決定できるネットワークを提案する。具体的には、入力のタイミングは間隔のゆらぎを吸収することが可能となるよう細かく設定し、信号が伝搬するタイミングはシンボルの平均間隔に設定する必要がある。

ここで入力のタイミングを細かくした際の実出力ユニットへの影響について考える。出力ユニットで使用している LIF モデルの特徴として積分発火と内部電位の漏れがある。積分発火により少しのずれをもつ入力が行われても、内部電位が下がりきらなければ、発火する事が可能である。また、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、内部電位の漏れの量が少ないと出現頻度が高くない部分列を抽出してしまう場合がある。これは内部電位の漏れが少ないことによって、しきい値を超える回数が増えるため、出現頻度の高い部分列と一部共通部分をもつ入力が行われた時、出力ユニットが発火する事が多いからである。よって部分列として抽出するには出力ユニットの発火頻度は押さえる必要がある。発火頻度を押さえる方法としては側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、側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すぎると、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しきい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が必要である。具体的には出力ユニットの時定数

を小さくすることで、. 時定数を小さくすると、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り、一部共通部分をもつ入力が行われた時、反応することが減り、部分列を学習できる確率が上がる。その際、時定数を下げすぎると内部電位の漏れが多くなり、入力の少し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。

### 3.2.2 間隔のゆらぎを考慮した学習法の提案

この節では、3.1.2 項で挙げた間隔のゆらぎを考慮した学習法に関して議論する。STDP の窓関数は特に発火の前後で結合荷重の値が大きく変化する特徴をもっていた。そこで、学習の進行に影響が大きい発火前後の荷重の変動がユニットの発火時刻の差  $x$  に対してあまり変化しない窓関数を提案する。新たに提案する窓関数を式 (3.1), 式 (3.2) に示す。

$$W_{ij} \leftarrow \begin{cases} W_{ij} + A_+ (\exp(-\frac{x+t_{top}}{\tau_s}) - \exp(-\frac{x+t_{top}}{\tau_m})) & (\text{if } x > 0) \\ W_{ij} - A_- (\exp(\frac{x+t_{top}}{\tau_s}) - \exp(\frac{x+t_{top}}{\tau_m})) & (\text{if } x < 0) \\ W_{ij} & (\text{if } x = 0) \end{cases} \quad (3.1)$$

$$t_{top} = \frac{\tau_s \tau_m}{\tau_s - \tau_m} \ln \frac{\tau_s}{\tau_m} \quad (3.2)$$

$A_+$ ,  $A_-$  は STDP と同様に事前に決めるパラメータであり結合荷重の増減加量となる。また  $\tau_s$ ,  $\tau_m$  は時定数であり、結合荷重の変化量を決定する。 $t_{top}$  は  $\tau_s$ ,  $\tau_m$  によって決定する値であり式 (3.2) に従う。式 (2.1) と式 (3.1) の概形を図 3.5 に示す。図 3.5 に示すように  $x = 0$  の近傍で結合荷重の変動が抑制されていることがわかる。つまり、それぞれの結合間の  $x$  がずれることによる、窓関数の変化量が



最も多い  $x = 0$  の近傍の変化量があまり変化しない。これによって、間隔のゆらぎによって結合間の  $x$  がずれた場合に、結合荷重の値の変化量がほぼ同じとなる。

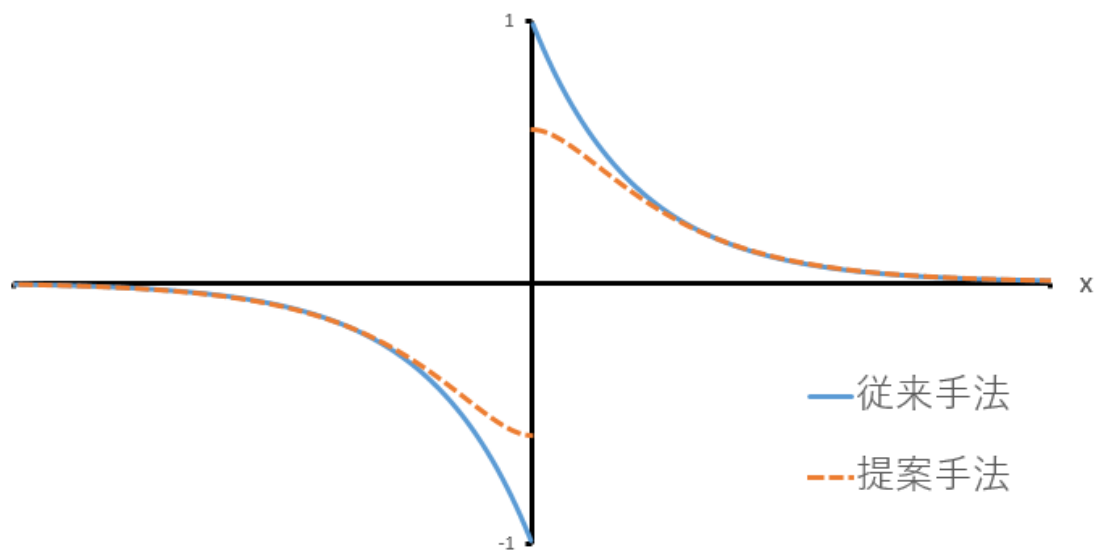


図 3.5: 従来手法と提案手法の窓関数の比較

## 第4章 実験

本章では，提案手法によりニューラルネットワークが出現間隔にゆらぎをもつ系列データから頻出な部分列を抽出することができるのかを確認する．

### 4.1 実験方法・条件

本節では，間隔にゆらぎをもつシンボル列から頻出な部分列を抽出できるかを，音声データから各音素を取り出し，ストリーミング形式で音素を出現タイミング通りにネットワークに与え，ネットワークが頻出な部分列を抽出できるのか既存手法と比較することで確認を行った．

既存手法はさまざまな長さの部分列を抽出することができるため，提案によりネットワークが出現間隔のゆらぎに対応できているかに焦点を当てる．そのため，抽出したい部分列は一つとし，その部分列の長さは2とした．

実験では，音声データとして国立情報学研究所データセット共同利用研究開発センターから提供されている，音声コーパスのうち重点領域研究「音声対話」対話音声コーパス (PASD) [17] の模擬会話音声を使用した．この音声データに対してインターネット上で公開されている汎用大語彙連続音声認識エンジン「Julius」[18] を用いて音素セグメンテーションを行い，64種類のシンボル情報（日本語のひらがな1文字）と，時間情報（前のシンボルの発生から次のシンボルの発声までの間隔）を持つ時系列シンボルデータを作成した．この時系列シンボルデータにおける最も頻出な部分列の出現率は従来手法で抽出可能であった出現率とほぼ同等となる音声データを選択した．時系列シンボルデータを時間情報の間隔ごとに1シンボルずつネットワークへ入力する．ネットワークは，64入力であるのでシンボルブロックのユニットは合計で128個用意した．シンボルの総数は3,003シンボルであった．本実験で用いた時系列シンボルデータの長さ2の部分列の出現頻度は

表 4.1: シンボルの長さ 2 の部分列の出現頻度

部分列	出現頻度 (%)
「はい」	4.03
「です」	1.67
「いえ」	1.57
「ます」	1.43
その他 1.30% 以下	

下記の表 4.1 の通りである.

提案手法の入力のタイミング, 信号の伝搬するタイミング, 結合荷重の初期値, 出力ブロックのユニット, それらの各パラメータは表 4.2 とした. また学習で使用するパラメータは表 4.3 とした. これらのパラメータは実験により決定した. 既存手法で同程度の出現頻度の抽出を行った場合と比較するため, 時間情報を一定の間隔にした時系列シンボルデータを作成した. その際, 既存手法の結合荷重の初期値, 出力ブロックのユニットのパラメータは表 4.2 と同じ値を用いた. 学習で使ったパラメータは表 4.4 とした. これは提案手法と窓関数の概形が大きく変化しないような値を選択した.

すべてのシンボルをネットワークに与えたあと結合荷重の値を見て, 頻出な部分列にのみ反応する状態となった出力ブロックのユニットが存在するかどうかで頻出な部分列を抽出できたか判断した.

表 4.2: ネットワークの各パラメータ

入力のタイミング	1ms
信号の伝搬するタイミング	145ms
結合荷重の初期値	5
$V_r$	-68mV
$\theta$	-28mV
$\tau_r$	145

表 4.3: 提案手法での学習の各パラメータ

結合荷重の上限値	10
$A_+$	1
$A_-$	1
$\tau_s$	145
$\tau_m$	29
$D_-$	0.5
$\tau$	14.5

表 4.4: 既存手法での学習の各パラメータ

結合荷重の上限値	10
$A_+$	0.5
$A_-$	0.5
$\tau_s$	145
$D_-$	0.5
$\tau$	14.5

## 4.2 実験結果・考察

本節では、4.1節で述べた条件で実験を行った結果と提案手法の有効性について議論する。

シンボルをすべてネットワークに与えた後、出力ユニットへ接続している結合荷重の値を調べ、特定の部分列のみに反応する状態となった出力ブロックのユニットを確認した。表 4.5 は提案手法を用いた場合の各出力ユニットの抽出状態を表している。表 4.6 は既存手法を用いた場合の各出力ユニットの抽出状態を表している。

表 4.5: 提案手法によって抽出された部分列

ユニット番号	抽出した部分列	かかった時間
1	「はい」	51.237 秒
2	×	×

表 4.6: 既存手法によって抽出された部分列

ユニット番号	抽出した部分列	かかった時間
1	「はい」	41.035 秒
2	×	×

提案手法は最初のシンボルを与えてから約 52 秒後に出力ブロックの第 1 層のユニット 1 にて頻出部分列「はい」を抽出した。ユニット 2 では頻出部分列が抽出されていないため、このネットワークは頻出部分列のみを抽出したと考えられる。また、既存手法は提案手法よりも約 10 秒早く抽出が完了している。この約 10 秒間に出現頻度が 4% の部分列「はい」は 3 回出現している。今回の実験では窓関数の概形が大きく変化しないように既存手法のパラメータを設定しているため、窓関数の最大値は既存手法の方が大きい。既存手法と提案手法の窓関数の最大値の差によって 3 回分の変化量に相当する差が出たと考えられる。

図 4.1 は提案手法のユニット 1 に接続する結合荷重の値の変化を表している。この 2 本の線は結合荷重の変化を表し、それぞれ「はい」の「い」がネットワーク

に与えられた瞬間にシンボルブロックで発火するユニットからの結合荷重である。図 3.3 と比較して、2つの結合荷重の変化量はほぼ同じ量であり、結合荷重の値が最大値付近の場合の傾きもほぼ同じ大きさであることがわかる。そして、2つの結合荷重の値が最大値である 10 にほぼ同時に到達していることがわかる。このことから、発火前後の窓関数の変化を抑制することで、ユニットの発火タイミングのずれによる結合荷重の変化量がほぼ同じに抑えられていると考えられる。

以上の結果より、部分列の抽出結果と結合荷重の変化から、ネットワークがシンボルの出現間隔に対応できていることが確認できた。これにより、本提案が有効であると考えられる。

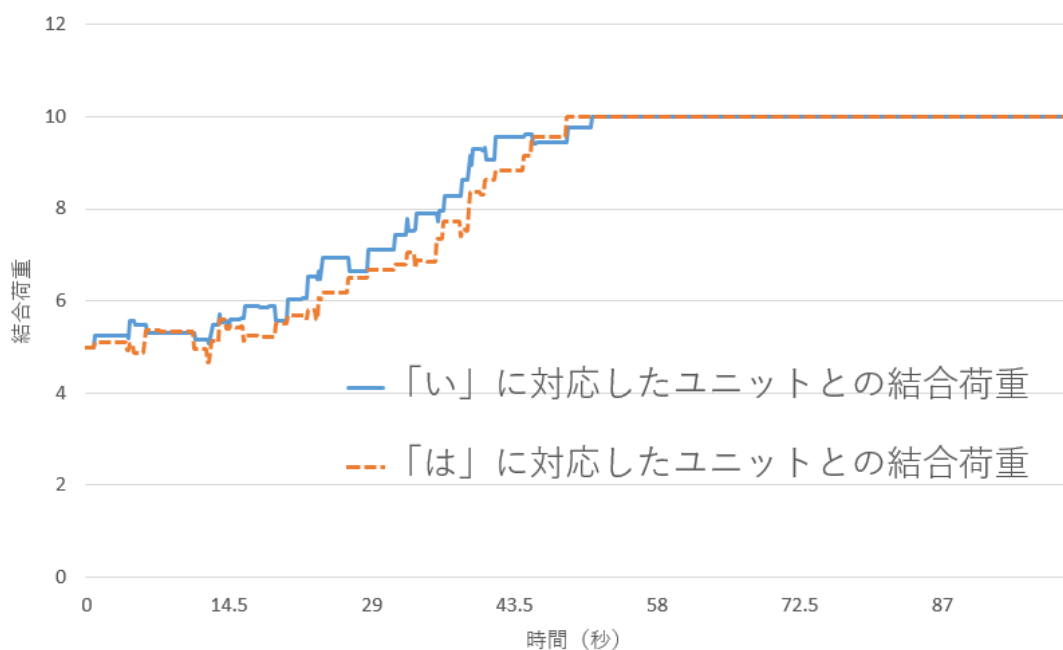


図 4.1: 結合荷重の変化

## 第5章 結論

本稿では、音声データから頻出な単語を取り出す第1歩として、音声の特徴をもつシンボル列から頻出な部分列を自動的に抽出を目的とした。具体的には日本語の音声データから各音素を取り出し、音素を出現タイミングどおりにストリーミング形式でシステムに与え、システムが頻出な部分列を抽出することを目指した。

シンボル列から頻出な部分列を抽出するために、既存手法を拡張することを試みた。この既存手法は、音素の出現が一定間隔であれば部分列を抽出可能であるが、出現間隔にゆらぎがある場合には抽出することができない。そこで、既存手法に対して2つの拡張を提案した。1点目は、信号の入力するタイミングと信号の伝搬するタイミングを任意に決定できるネットワークの提案を行った。2点目は、学習において発火タイミングが少しずれた場合も、結合荷重が大きく変化しない窓関数の提案を行った。

提案手法の有効性を確認するために、模擬会話音声に対し音素セグメンテーションを行い、シンボル情報と時間情報をもつ時系列シンボルデータから頻出な部分列の抽出ができるか確認を行った。その結果、使用した時系列シンボルデータ内で最も頻度の高い部分列「はい」を抽出することができた。また、結合荷重の変化量からユニットの発火タイミングのずれによるSTDPによる変化量がほぼ同じに抑えられていることを確認した。

今後の課題としては、抽出するシンボルの長さを拡張する方法について検討がある。本稿では、議論を簡単化するためシンボルの長さを2としているが、単語を取り出す際にはシンボルの長さはさまざまである。そのため、今回提案した手法が、抽出するシンボルの長さを拡張した際に、3つ以上のシンボルのゆらぎに対応することができるのか検討していく必要がある。

# 謝辞

本研究の遂行および本論文の作成にあたり，懇切丁寧なご指導と御督励を賜った本学工学研究科電気電子工学専攻の高瀬治彦教授，北英彦准教授，川中普晴准教授，博士後期課程の森田賢太様，本学理事・副学長の鶴岡信治教授に感謝いたします。また，日頃熱心に討論していただいた計算機工学研究室，情報処理研究室の皆様方に厚く御礼申し上げます。最後に，本論文をまとめるにあたり，助言，討論，その他お世話になったすべての方々に感謝いたします。



## 参考文献

- [1] A. Hunt and A. Black, Unit selection in a concatenative speech synthesis system using a large speech database, Proc. International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 373–376, 1996.
- [2] H. Zen, K. Tokuda and A. Black, Statistical parametric speech synthesis, Speech Commun., Vol 51, pp. 1039–1064, 2009.
- [3] 橋本 佳, 高木 信二, 深層学習に基づく統計的音声合成, 日本音響学会誌 Vol 73, No. 1, pp.55–62, 2017.
- [4] J. G. Wilpon and C. N. Jacobsen, A study of speech recognition for children and the elderly, Proc. International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 349–352, 1996.
- [5] K. Konno, M. Kato, and T. Kosaka, Speech recognition with large-scale speaker-class-based acoustic modeling, Proc. APSIPA, pp. 1–4, 2013.
- [6] 関 博史, 榎並 大介, 朱 発強, 山本 一公, 中川 聖一, 話者クラスタリングに基づく短時間発話音声認識, 電子情報通信学会論文誌 Vol J100-D, No. 1, pp. 81–92, 2017.
- [7] 鷹見 淳一, 加藤 喜永, 佐藤 奈穂子, 呂 彬, 半自動音声ラベリングシステムによる音声データベース構築作業の効率化, 日本音響学会研究発表講演論文集, pp. 171–172, 2001.
- [8] 古樋 直己, 偶発的語彙力習得と英語力、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.

- [9] 河野 守夫, モーラ, 音節, リズムの心理学言語の考察, 音声研究, Vol 2, No. 1, pp. 16–24, 1998.
- [10] Shigeaki Sakurai, Minoru Nishizawa, Discovery of Various Sequential Patterns within Top-k form Sequential Data, Proceedings of 2014 SCIS&ISIS, pp. 446–451, 2014.
- [11] ELMAN, J.L, Finding structure in time, Cognitive Science, Vol. 14, pp. 179–211, 1990.
- [12] 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.
- [13] 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.
- [14] Kenta Morita, Haruhiko Takase, Naoki Morita, Hiroharu Kawanaka, Shinji Tsuruoka, Stable Extraction of Frequent Sub-sequences from Sequential Symbol Input, Proceedings of International Conference on Knowledge Based and Intelligent Information and Engineering Systems, pp. 1449–1453, 2017.
- [15] Stein RB, A theoretical analysis of neuronal variability, Biophysical Journal, pp.173–195, 1965.
- [16] Teuvo Kohonen, Self-Organized Formation of Topologically Correct Feature Maps, Biological Cybernetics, Vol. 43, No. 1, pp. 59–69, 1982.
- [17] 板橋 秀一, 市川 薫, 河原 達也, 樽松 明, 小林 哲則, 小林 豊, 白井 克彦, 土屋 俊, 鶴丸 弘昭, 新美 康永, 松本 裕治, 山下 洋一, 山本 幹雄, 横田 将生, 重点領域研究「音声対話」対話音声コーパス (PASD).
- [18] Akinobu Lee, Tatsuya Kawahara, Kiyohiro Shikano, Julius - an Open Source Real-Time Large Vocabulary Recognition Engine, EUROSPEECH2001:

the 7th European Conference on Speech Communication and Technology,  
pp.1691–1694, 2001.

## 発表実績

- [1] Masachika Sawamura, Kenta Morita, Haruhiko Takase, and Hidehiko Kita, Extraction of Frequent Sub-sequences from Spiking Neural network: Extraction from Symbol Sequence with the Fluctuation of the Interval, Proceedings of the 10th International Workshop on Regional Innovation Studies (IWRIS2018), pp.87–90, 2018.
- [2] Masachika Sawamura, Kenta Morita, Haruhiko Takase, and Hidehiko Kita, Extraction of Frequent Sub-sequences from Spiking Neural network: Handling with Fluctuation of the Interval between Symbol, Proceedings of The 8th International Symposium for Sustainability by Engineering at Mie University (IS2EMU2018-C), pp.35–36, 2018.
- [3] 澤村 将周, 森田 賢太, 高瀬 治彦, 川中 普晴, 鶴岡 信治, 頻出な部分列を抽出するスパイクングニューラルネットワーク—間隔にゆらぎをもつシンボル列からの抽出—, 第43回東海ファジィ研究会予稿集, pp.16-1–16-4, 2017.
- [4] Kenta Morita, Haruhiko Takase, Masachika Sawamura, Naoki Morita and Hidehiko Kita, Extraction of Frequent Sequential Patterns from Sequence at Uneven Intervals, Proceedings of 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems and 19th International Symposium on Advanced Intelligent Systems, pp.1449–1453, 2018.
- [5] 小野田 憲悟, 澤村 将周, 高瀬 治彦, 川中 普晴, 鶴岡 信治, 時系列情報を処理するためのニューラルネットワーク—SpikePropにおけるパラメータが学習に及ぼす影響の調査—, 地域イノベーション学会誌, Volume 6, p. 34, 2017.