

# RNNPB による自然言語列と動作列の意味的結合と人間ロボットインタラクション

村瀬 昌満<sup>†</sup> 尾形 哲也<sup>†</sup> 谷 淳<sup>‡</sup> 駒谷 和範<sup>†</sup> 奥乃 博<sup>†</sup>

<sup>†</sup> 京都大学大学院 情報学研究科 知能情報学専攻 <sup>‡</sup> 理化学研究所 脳科学総合研究センター

## 1. はじめに

人間同士の協調において、言語は使われる文脈によって大きく意味を変えうる、非常に強力なシンボルとなっている。しかし、逆に言語の意味が文脈に強く依存するという性質によって生じる記号接地問題のために、ロボットが言語をそのまま扱うのは非常に困難である。

杉田ら [1] は、ロボットの動作用と言語用の二つの神経回路モジュールを用い、動作と言語の対応を学習している。しかし、あらかじめ動作を分割して与えているので、記号接地問題の一部である、連続した動作をどのように言語に対応させるかという問題は解決していない。

これは、言語と現実世界の対応が多対多であることに起因する。現実世界は連続であるため、どのように切り分けて言語に対応させるかは一意ではない。また、言葉が表す意味は文脈によって変わりうるため、言語をどのような動作に対応させるかも一意ではない。

そこで、本研究では、連続した現実世界の事象 (sensory-motor flow) を、予測誤差に基づいた分節化を行い、言語との対応を学習させる。また、文脈による変化を扱うために、現実世界と言語の双方を力学系モデルで扱い、これらを結合することで対応を学習させる。具体的には、ロボットの動作を、予測誤差に基づいて分節化し、生成、認識する神経回路モジュールを構築し、同様に言語を生成、認識する神経回路モジュールを構築した。神経回路モジュールには、谷らによって提案された Recurrent Neural Network with Parametric Bias (RNNPB)[2] を用い、入力層の一部である PB 層をバインディングすることで、動作から言語、言語から動作の双方向の変換を実現した。

## 2. RNNPB による時系列データの分節化

### 2.1 RNNPB

RNNPB は、現状態を入力として与え、次状態を出力する予測器である。RNNPB は PB 層と呼ばれる入力層を持っており、PB 層への入力値 (PB 値) を変更することで、異なるシーケンスを生成することができる。また、RNNPB を認識に用いることで、認識対象の時系列データを生成するような PB 値を得ることができる。

### 2.2 複数のシーケンスへの分節化

本研究では軌道ではなく、ダイナミクスに基づいた分節化を行う。すなわち、安定した予測が可能な区間を単一のシーケンスであると見なし、分節化を行う。そこで、与えられた分節化数に応じて分節化を行う、次のような分節化手法を開発した。

- 1) 初期化: 与えられた時系列データを、与えられた分節化数の区間に均等に分割する。
- 2) RNNPB の学習: 各区間について RNNPB の学習を行い、結合重み、PB 値を更新する。
- 3) 誤差の算出: 各区間について、予測値を求め、その区間における予測誤差の最大値を求める。

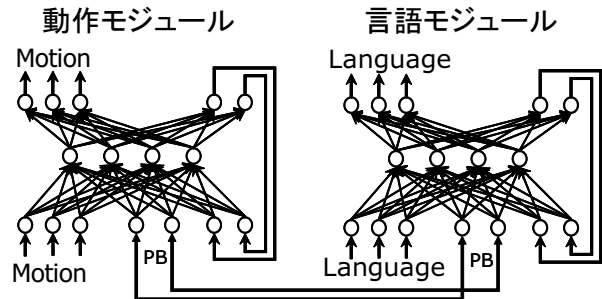


図 1: PB 層がバインディングされた RNNPB の構造

- 4) 区間幅の更新: 3) での予測誤差が隣接区間のものよりも大きければ区間幅を減少、小さければ区間幅を増加させる。
- 5) 全体の予測誤差が小さくなるまで 2) から 4) を繰り返す。

単一の PB 値で予測可能なシーケンス (= 単一のダイナミクス) では、予測が成功するのでエラーが小さくなるが、複数のダイナミクスから成るシーケンスでは、予測に失敗するのでエラーが大きくなる。そのため、上記の手順により、区間の境界はシーケンスの境界に近付き、ダイナミクスに基づいた分節化が可能となる。

## 3. 言語列と動作列の対応関係の獲得

### 3.1 動作モジュールと言語モジュールへの入力

動作列を学習する RNNPB (動作モジュール) では、入力として、ロボットのセンサ、モーターの値である sensory-motor flow を用いる。

また、言語列を学習する RNNPB (言語モジュール) では、入力として、Elman ら [3] と同様に、文章の単語をビット表現したものを与える。これは、各単語を入力層の各素子に対応させ、入力する単語の素子に 1 を、それ以外の素子に 0 を与えることで行う。

### 3.2 PB 層をバインディングした RNNPB

言語列と動作列の対応関係を学習するために、これら二つの RNNPB の PB 層をバインディングしたものをを用いる (図 1)。

#### 3.2.1 学習

動作列と、それに対応する  $N$  個の文章から成る言語列が与えられたとき、以下のように学習を行う。

- 1) 動作列を  $N$  個に分節化する。
- 2) 分節化された動作列の  $i$  番目の区間と  $i$  番目の文章をそれぞれのモジュールで学習させる。この時、互いの PB 値が等しくなるようにバイアスをかけながら PB 値を更新する。
- 3) 2) を  $i = 1, \dots, N$  について行い、これを繰り返す。

この学習の結果、対応する動作列と言語列をそれぞれのモジュールで認識させると、等しい PB 値が得られる。すなわち、PB 値を中間表現として、動作列と言語列のバインディングが可能となる。

Semantic combination between linguistic and behavioral sequences using RNNPB for human-robot interaction: Masamitsu Murase (Kyoto Univ.), Tetsuya Ogata (Kyoto Univ.), Jun Tani (RIKEN), Kazunori Komatani (Kyoto Univ.), and Hiroshi G. Okuno (Kyoto Univ.)

表 1: 学習に用いた動作パターンと文章

動作パターン	文章
(1) 黄 赤 青 白(遅い)	Move to red slowly. Move to blue slowly. Move to white slowly.
(2) 黄 赤 青 赤(遅い)	Move to red slowly. Move to blue slowly. Move to red slowly.
(3) 黄 赤 黄 赤(速い)	Move to red fast. Move to yellow fast. Move to red fast.
(4) 黄 赤 白 赤(速い)	Move to red fast. Move to white fast. Move to red fast.
(5) 黄 青 赤 黄(遅い)	Move to blue slowly. Move to red slowly. Move to yellow slowly.
(6) 黄 青 白 黄(遅い)	Move to blue slowly. Move to white slowly. Move to yellow slowly.
(7) 黄 青 黄 赤(速い)	Move to blue fast. Move to yellow fast. Move to red fast.
(8) 黄 青 白 赤(速い)	Move to blue fast. Move to white fast. Move to red fast.

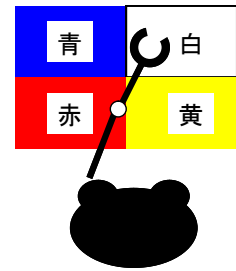


図 2: 色の配置

### 3.2.2 生成

学習後、与えられた言語列を言語モジュールで認識し、得られた PB 値を動作モジュールに入力することで、その言語列に対応する動作列を生成することが可能となる。

同様に、与えられた動作列を動作モジュールで認識し、得られた PB 値を言語モジュールに入力することで、対応する言語列を生成することが可能となる。このとき、与えられた動作列を任意の数に分節化し、切り分けられた各動作列を動作モジュールで認識し、得られた各 PB 値から対応する言語列をそれぞれ生成することで、与えられた動作列を任意の数の文章で表現することができる。

## 4. 言語列と動作列の対応獲得実験

### 4.1 実験条件

実験には、RobovieIIs (図 3) を用いた。今回は、卓上 4 色領域間腕移動タスクを行うため、頭部 3 自由度のうち pitch 軸と yaw 軸、左腕 4 自由度のうち、肩の roll 軸と肘の pitch 軸、および頭部 CCD カメラを用いた。動作モジュール用 RNNPB への入力として、用いた関節の角度 (4 次元)、CCD カメラ画像中の各色の割合 (4 次元) を 0.1 秒毎に取得し、0 から 1 の値になるように正規化したものを与えた。

実験は、図 2 に示すように、机が赤、青、黄、白の 4 色の領域に分けられた環境で行った。学習時に、RobovieIIs はあらかじめプログラムされた軌道に従って「黄 赤 青」のように左腕、頭部を動かす。このとき得られた sensory-motor 流を動作モジュールに、この動作に対応する文章 (“Move to red slowly. Move to blue slowly.” など) を言語モジュールに入力し、学習を行った。

学習に用いた動作パターンとそれに対応する文章の組を表 1 に示す。ここで「遅い」と書かれたパターンは、2 秒かけて次の色領域に移動し「速い」と書かれたパターンは、1 秒かけて次の色領域に移動している。表に示すように、move, to, red, blue, yellow, white, slowly, fast の計 8 単語からなる言語を用いた。この場合、(5)、(6) のように、同じ “Move to yellow slowly.” であっても、実際の動作は異なるものとなる場合が存在する。つまり、言語と動作の対応は 1 対 1 の関係ではない。

また、動作モジュールの RNNPB の各層のサイズは、入出力層 8、中間層 50、文脈層 8 とし、PB 層は 4 とした。言語モジュールの RNNPB の各層のサイズは、入出力層 9、中間層 20、文脈層 9 とし、PB 層は 4 とした。

これらの動作、言語の組を学習後、言語から動作、動作から言語への変換を行った。

### 4.2 実験結果・考察

腕の初期位置が黄であるときに、言語モジュールに “Move to white slowly.” を入力し、対応する動作の生成

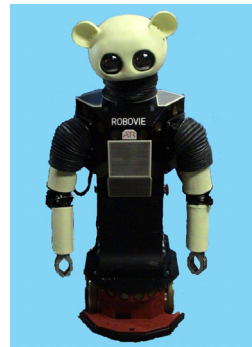
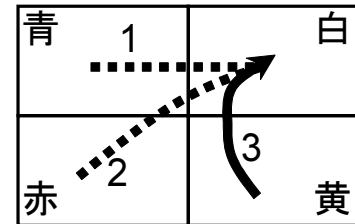


図 3: RobovieIIs



1: 学習済みの軌道(青 白)  
2: 学習済みの軌道(赤 白)  
3: 未学習の軌道(黄 白)

図 4: 言語から動作の生成例

を行ったときの軌道を図 4 の 3 に示す。学習データには青 白、赤 白の移動(図 4 の 1, 2)のみで、黄 白への移動は学習データに含まれていないが、適切な軌道が生成されていることが分かる。

次に、動作列から言語列の生成を行った。黄 赤 青へ速く移動するという動作を入力し、文章を生成させた。その結果、文章の数として 2 を指定した場合、適切な文章である、“Move to red fast. Move to blue fast.” が生成された。さらに、文章の数として 1 を指定した場合、“Move to blue slowly.” が生成された。動作全体を一つに見なした場合、“Move to blue” と表現するのは適切であり、また、動作に要する時間は fast の 2 倍となっているため、slowly と出力されたのも適切であると言える。

## 5. おわりに

多対多の関係である言語と動作の対応の学習において、予測誤差に基づいて動作列の分節化を行い、言語列と対応させて学習する手法を開発した。卓上色領域間腕移動タスクにおいて、言語から動作の生成、動作から言語の生成が可能であることを検証した。また、未学習の動作も生成可能であること、同じ動作列に対して生成させる文章の数を変えると、その文章の数で表現可能な適切な文章が出力されることを確認した。

謝辞 本研究は科研費若手研究 A (No. 17680017) 栢森情報科学振興財団設立 10 周年記念特別研究助成、理研の支援を受けた。

### 参考文献

- [1] Sugita *et al.*: “Learning Semantic Combinatoriality from the Interaction between Linguistic and Behavioral Processes”, *Adaptive Behavior*, 13, 1, pp.33–52, 2005.
- [2] Tani *et al.*: “Self-Organization of Behavioral Primitives as Multiple Attractor Dynamics: A Robot Experiment”, *IEEE Trans. on systems, Man, and Cybernetics PartA: Systems and Humans*, Vol. 33, No. 4, pp.481–488, 2003.
- [3] Elman: “Distributed representations, simple recurrent networks, and grammatical structure”, *Machine Learning*, 7, pp.195–225, 1991.