# Blockly を利用したタグ付きコーパス検索パタン構築ツール

岡田魁人
岡山大学大学院自然科学研究科
pdrj11bt@s.okayama-u.ac.jp

竹内孔一
岡山大学大学院自然科学研究科
takeuc-k@okayama-u.ac.jp

## 1 はじめに

本研究はブラウザ上でプログラムを組み立てることができる Blockly を利用してタグ付きコーパスから目的とする表現を取り出すための言語パタンの構築を補助するツールについて記述する．基本的な言語パタン抽出の応用としてコンコーダンサや質問応答などでの内部処理が挙げられる．まず，コンコーダンサについて説明する．コンコーダンサとは，コーパスを分析するソフトのことである．自然言語処理の分野において，語の特徴や傾向性を明らかにするためにコーパスが利用される．コーパスを利用するには，大量の電子資料や専用マニュアルを読む必要があり，直接読むことは学習コストが高くなるなどの問題がある．そこで，語の出現状況をわかりやすく示すコンコーダンサを使うことが一般的である．テキストコーパス内の文字列や単語を検索し，検索単語を中心として，前後の文脈とともに示される KWIC 形式のように表示するコンコーダンサは既に数多く存在する [1][2]．しかし既存のコンコーダンサでは，表層上の文字列や単語検索以外に，例えば同義語や類義語，同じ意味を示す文などの柔軟な検索に対応していない．コンコーダンサ以外に，日本語のタグ付きコーパスを柔軟に検索できるツールはほとんど存在しない [3][4]．図 1 に「走る」を検索した場合の KWIC の例を示す．

| | | |
|---|---|---|
| 太郎は公園を | 走っ | ていた時 |
| その時 | 走り | 去った新幹線はとても速く |
| 背中に悪寒が | 走っ | た |
| 山間を道が | 走っ | ている |

**図 1** 「走る」を検索した場合の KWIC の例

一方，テキストに対して，意味役割付与システムから得られる解析結果を用いれば，述語の概念など意味タグをもとにした検索が実現できる．例えば，竹内研では意味役割付与解析システム (ASA[1)]) が構築されている．ASA の結果を用いて，「表情の出現」という概念を持った動詞を検索をすると，図 2 のように述語の概念分類に沿った例文を取り出すことが可能である．

| | | |
|---|---|---|
| 背中に | 悪寒が | 走る |
| 顔に | 恐怖が | 走った |
| 怒りが | 顔に | 出る |
| 疲れが | 顔に | 滲む |

**図 2** 「表情の出現」という概念を検索した場合の提案システムの表示例

次に，質問応答システムにおいて言語パタンを利用する研究事例が挙げられる．IBM が開発した質問応答システム Watson では，あらかじめ質問に対する答えを構文のパタンとして構築している [5]．意味役割が付与されたコーパスに対し，KWIC のみではなく，柔軟に検索できるツールがあれば，パタン構築を素早く仕上げることができる．

よって本研究では前節の背景を踏まえて，意味役割付与されたタグ付きコーパスに対し，柔軟な検索に対応し，コンコーダンサの機能を持ち，さらに質問応答システムに見られる前処理をも行えるような多機能ツールを構築する．本システムは 2 つのモジュールから構成されており，1 つがユーザインタフェースの部分，もう 1 つはパタンマッチの部分である．本論文では，前者のユーザインタフェースの構築について記述する．

## 2 関連研究

関連研究として，「茶器」[3] がある．「茶器」は，品詞，文節，係り受けといった統語情報などタグ付けされたコーパスを柔軟に検索することができ，関

1） http://www.cl.cs.okayama-u.ac.jp/study/project/asa/asa-scala/

係データベースを用いて検索システムとして実装したものがあり [6]，ドイツ語の Treebank を対象にした同様のものがある [7]．また，木構造のタグ付与データに対して検索するツールが提案されている [8]．最終的には品詞，文節等のチャンク，係り受け等の統語情報を含むタグ付けコーパスに対して，柔軟な検索機能を備えるだけでなく，各種統計解析機能や辞書とコーパスの連携，タグ付けエラーの修正などの機能を持ったタグ付きコーパスの検索/管理システムを目指しているものである [9]．

# 3 言語パタンマッチシステム

本提案システムは，ユーザインタフェース部分に Blockly[2] を採用して，言語パタン構築モジュールを実装する．Blockly を採用することで，言語パタンをより直感的に組むことができる．

## 3.1 項構造ベースの言語パタンマッチ

・X は Y の「著者」だ
　=>X は Y を「書いた」
・X の「親」は Y だ
　=>X は Y の「子供」だ
・Y は X の「店員」だ
　=>Y は X で「働いて」いる

**図 3** 項構造と動詞への言い換え例

本提案システムにおける言語パタンマッチについて説明する．前提として，ASA により意味役割付与されたコーパスを検索の対象とする．語には，語そのものに関係性を表すようなものがある [10]．例えば図 3 のような構文パタンがある．

この例からわかるように，ある一文から作者と作品名を抜き出すことや，係り受けの関連を取り出すこと，また，単純に文字列や単語そのものも含めてユーザが構築した言語パタンをもとに対応したパタンマッチ処理を実行することを本論文では言語パタンマッチと定義する．

## 3.2 言語パタンマッチシステムの機能

本提案システムには様々な機能が必要となる．様々な視点からコーパスの検索が可能であること，および検索結果の表示や再加工を行えるよう適切なデータやプログラムに変換する処理を行う必要が生じる．検索要求は，単語の綴りそのもの，単語の一部，単語の原形，品詞などの文法情報，意味タグが付与された構文構造 (prolog の木に分解) [11]，係り受けの関係，およびそれらを組み合わせたものであり，以下言語パタンと述べる際にはこれらを含めたものとする．下記に，システムに求める機能について記述して整理する．

**検索**
　ユーザが構築した言語パタンに応じた適切な検索が行える．

**言語パタンの構築**
　視覚的に言語パタンを組むことができる．ユーザが必要とする言語パタンを組み合わせて実現できるような多種の検索要素を提供でき，単語や品詞だけでなく，ASA により付与される概念フレームなど意味的なタグについても検索できる検索要素も提供できることが好ましい．

**言語パタンの保存**
　上記の機能で構築した言語パタンを保存できる．保存した言語パタンは txt ファイルとしてダウンロードができるため，ユーザ間で共有することができる．

**言語パタンの読み込み**
　ユーザが組み立てた言語パタンを読み込む．これにより他の人が作成した言語パタンも利用できる．

**検索結果の出力**
　検索結果を表示するだけでなく，検索結果を例えば二次加工がしやすい json 形式や csv 形式のファイルに書き込み，ダウンロードができる．

## 3.3 言語パタンマッチの全体像

提案システムの実装には，npm (Node Package Manager) を用いる[3]．ここで，システムの処理の流れを視覚化するためフローチャートを図 4 に示す．以下ではフロントエンドとバックエンドの各処理について説明する．

### 3.3.1 フロントエンド: ユーザによる操作の処理

ユーザの操作項目についての説明をする．最初に解析対象とするテキストデータを読み込ませる．

2） https://developers.google.com/blockly

3） https://www.npmjs.com/

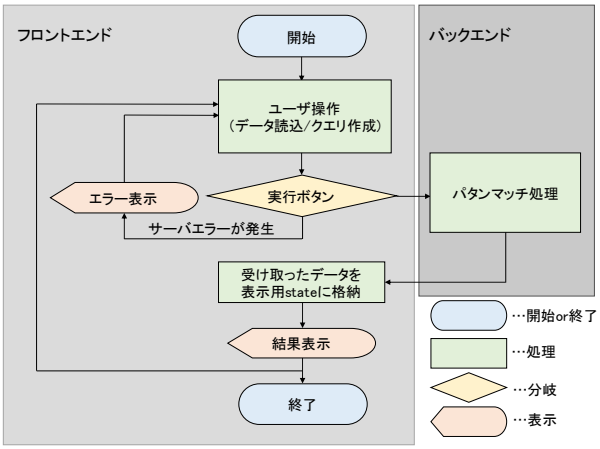


**図 4**　提案システムのフローチャート

解析対象は複数のファイルを登録することができ，チェックボックスで切り替えることができる．次に，クエリ作成を行う．クエリとは，検索要求を組み合わせて構築する言語パタンのことである．クエリはデータ読み込みから作成することもできる．実行ボタンを押すことで言語パタンマッチから表示に必要なデータ加工までの処理が実行される．

#### 3.3.2　バックエンド: タグ付与と検索処理

本論文ではテキストを解析するシステムとして意味役割付与システム ASA を利用する．ユーザがまず最初に検索したいテキストをシステムに入力した場合，ASA による解析を行った後，係り受け，形態素，述語の概念フレーム，意味役割をすべて Prolog 形式の探索木に変更する [11]．次にフロントエンド側で言語パタンが構築され，検索要求がだされたときに，検索言語を受け取り prolog の機能を利用して，マッチしたテキスト部分をフロント側に送る．検索の際，文，係り受けのチャンク，形態素の単位で探索木を構成しているが，今のところ，検索結果の単位は文の集合を返している．

### 3.4　言語パタンマッチシステムの拡張性

今回実装するシステムの言語パタン構築部分は，バックエンドで処理する言語処理ツール用にカスタマイズしている．よって，他の言語処理ツールを適用する場合にはそれに合わせたタグ体系に合わせることで，複雑な解析も Blocly を利用することで手軽に行えるようになる．

## 4　動作確認実験

本提案システムについて，以下で示すデータセットを用いて動作実験を行う．動作実験の目標として，図 3 の例として述べた著者と作品の組み合わせを抜き出すこととする．

### 4.1　検索対象例文

検索対象例分として図 5 に示すように，著者と作品名を文で表した例文を 10 文用意した．例にもあるように「図書館戦争を書いた有川浩」など連体修飾の例も入れている．また，パタンにマッチしない文も入れている．

泥棒に財布を盗まれる。
有川浩が図書館戦争を書いた。
昨日、友達と喧嘩した。
今日、新しいコンピュータを買うつもりだ。
羅生門は芥川龍之介の小説だ。
親に好きな靴を捨てられた。
明日、学校で誕生日パーティーが開かれる。
大学の隣に、新しいスーパーが建つ。
初めて先生に褒められた。
図書館戦争を書いた有川浩

**図 5**　検索対象例文

### 4.2　言語パタン

Blockly を用いて実現した Prolog で構築した言語パタンを図 6 に示す．author ブロックとして `author(_author,_work)` という述語を宣言し，緑のブロックで囲われているエリアに述語の定義を記述する．_author と_work は項であり，検索結果が入る．author ブロック内では author を取り出す具体的な構造が各ブロックで記述されている．例えば「type (X0, verb)」は品詞のタイプとして verb のものを探索木から取り出し，その要素を変数 X0 にセットする．また，semantic は概念フレーム，main は主辞，role は意味役割を表す．ここで定義した author ブロックを利用して，さらに他の言語パターンを取り出す際に利用することができる．

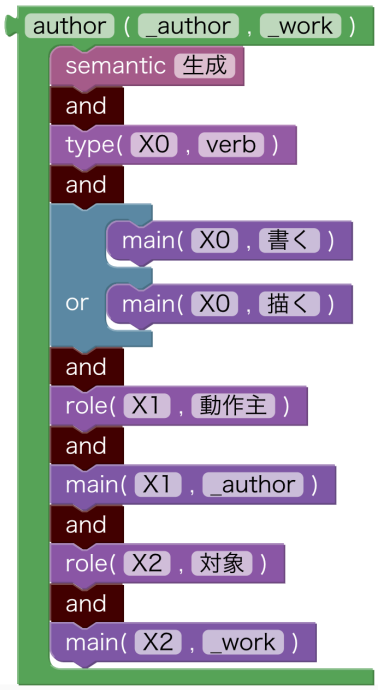


**図 6**　著者作品を抽出するための Prolog 言語パタン

### 4.3　動作結果

上記のデータセットに対してパタンマッチした結果の例を図 6 に示す．図 6 では，検索にマッチした文を表示していて「書く」や「〜の小説だ」といったパタンにマッチした文が取り出されている．単に文が取り出されているわけでなく内部が項構造として構築されているため，変数である_author には「有川浩」「芥川龍之介」，_work には「図書館戦争」「羅生門」が取り出されており，必要であれば，その部分だけをとりだすこともできる．

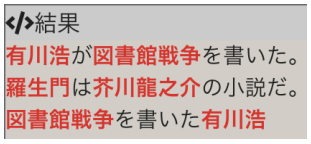


**図 7**　動作実験のスナップショット

## 5　おわりに

本論文では意味役割や概念フレームなどがテキストに探索木として付与されている場合にユーザが構築した言語パタンをブロックベースのプログラムとして組むことが出来るシステムについて記述した．実装として Blockly のライブラリを用いてブラウザベースの Web アプリケーションとしてシステムを構成した．実装に際して，言語処理の分野でよく利用される次の 2 つの機能の実現を目標とした．1 つはコンコーダンサの機能，そしてもう 1 つは質問応答システムに見られるような構文パタンを検索，処理を行うツールとしての機能である．実際に活用されるツールにするために，上記機能を満たすだけでなく，アプリケーションとして改善していく必要があるだろう．今後は，機能追加に加え，既存機能の改良，英語版対応および UI/UX の最適化を検討している．

## 謝辞

本研究の遂行にあたって JSPS 科研費 19K00552 の支援を受けた．

## 参考文献

[1] 小木曽智信, 中村壮範. 通時コーパス用『中納言』:Web ベースの古典語コンコーダンサー.
[2] 中條清美, 西垣 知佳子アントニ・ローレンス. フリーウェア WebParaNews オンライン・コンコーダンサーの英語授業における活用. 日本大学生産工学部研究報告 B (文系), Vol. 47, pp. 49–63, 2014.
[3] 松本裕治, 高岡一馬, 浅原正幸, 乾健太郎, 橋本喜代太, 投野由紀夫, 大谷朗, 森田敏生ほか. タグ付きコーパスの格納/検索ツール「茶器」. 言語処理学会第 10 回年次大会発表論文集, pp. 405–408, 2004.
[4] 田中良. 多言語対応コンコーダンサー『HASHI』: 日本語と日本語教育と社会言語学の研究を中心に. 2015.
[5] Adam Lally, John M Prager, Michael C McCord, Branimir K Boguraev, Siddharth Patwardhan, James Fan, Paul Fodor, and Jennifer Chu-Carroll. Question analysis: How Watson reads a clue. *IBM Journal of Research and Development*, Vol. 56, No. 3.4, pp. 2–1, 2012.
[6] 工藤拓, 松本裕治. Rdb を利用したタグ付きコーパス検索支援環境の構築. 情報処理学会自然言語処理研究会 2001-NL-144, pp. 135–142, 2001.
[7] Laura Kallmeyer. A query tool for syntactically annotated corpora. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pp. 190–198. Association for Computational Linguistics, 2000.
[8] Kohsuke Yanai, Misa Sato, Toshihiko Yanase, Kenzo Kurotsuchi, Yuta Koreeda, and Yoshiki Niwa. Struap: A tool for bundling linguistic trees through structure-based abstract pattern. In *Proceedings of the 2017 EMNLP System Demonstrations*, pp. 31–36, 2017.
[9] 浅原正幸, 米田隆一, 山下亜希子, 伝康晴, 松本裕治. 語長変換を考慮したコーパス管理システム. 情報処理学会論文誌 Vol.43,No.7, pp. 2091–2097, 2002.
[10] 西山佑司. ひつじ書房, 2003.
[11] 小笠原崇, 竹内孔一. 意味役割付与テキストに対する prolog ベースの探索木による言語パタンマッチシステム構築. 言語処理学会第 27 回年次大会, 2021.