

日本語推論システム lightblue の開発環境構築に向けて

佐伯 小遥 (指導教員: 戸次 大介)

1 はじめに

自然言語推論 (Natural Language Inference; NLI) は自然言語処理における重要なタスクである。計算言語学的なアプローチに基づく自動推論では、推論結果を出力するだけでなく、その推論の妥当性を保証する証明図を出力することができるため、説明性において利点がある。

lightblue[1] は計算言語学に基づく自動推論システムで、意味の理論に依存型意味論 (Dependent Type Semantics; DTS)[2] を採用している。lightblue に依存型理論 (Dependent Type Theory; DTT)[3] の定理自動証明器 wani[4,5] を接続することで、照応や前提束縛などを含む複雑な言語現象の解析を実現している。

近年、lightblue の推論の精度を向上させる研究が進められている中で [6,7], 解析結果や推論結果の視認性の高い可視化が行われていないため、結果の詳細な把握が困難である等の課題が生じている。可視化環境の実装によってこの課題が解決すれば、解析結果を実装のデバッグや理論の見直しに十分活用することが可能となる。そこで、本論文では日本語推論システム lightblue の開発環境である express [8] の改良に取り組む。

2 背景

2.1 日本語推論システム lightblue

lightblue¹ は関数型プログラミング言語 Haskell で実装された、統語論と意味論の分析を統合した日本語推論システムである。

統語理論に組合せ範疇文法 (Combinatory Categorical Grammar; CCG)[9,10] を、意味理論に依存型意味論を採用しており、日本語の文を入力として受け取ると CCG 統語構造や DTS 意味表示を出力する。

加えて、入力文に対して、意味表示の型が依存型理論における type であることを調べる型検査器、そして定理証明器 wani を組み合わせて自然言語推論を行い、それらが成功している場合にはその証明図を出力するという特徴を持つ [11]。この出力される証明図によって、lightblue による推論結果、型検査結果の妥当性が保証される。

2.2 課題: 文法開発における可視化

このように lightblue の実装は進んでいるが、文法開発は複雑な過程である。高度な専門性を要するばかりか、解析結果の分析に時間がかかるなど非効率な側面もあるため、文法開発過程の可視化が課題となっている。文法開発において、lightblue の解析結果を分析・評価することによって、CCG や DTS の理論の修正、lightblue や定理自動証明器 wani の実装の見直しがされることが理想である。しかしながら、現在 lightblue には、文法開発のために解析結果を分析しやすいような可視化環境が用意されていないため、実装のデバッグや理論の見直しに解析結果を活かすことが難しい。たとえば現在以下のような課題がある。

- 統語構造や型検査証明図、推論証明図を一度に全て出力しているため、分析対象の解析結果にアクセスしづらい
- 統語構造や証明図の全体図が画面に入り切っておらず、構造を把握しづらい
- デバックを支援する機能がないため、開発効率が低い

lightblue の HTML 表示では、解析結果が画面の右や下に広がる形式で提示されており、ユーザーはスクロール操作を通じて全体を確認する必要がある。このような出力形式では、解析結果の全体像を直感的に把握しにくく、結果の詳細な分析においてユーザーの負担が大きくなるという問題がある。既存の lightblue の解析結果の出力の様子は付録図 4 に示す。

3 express の機能

express には以下の 3 つの画面 (付録図 5 参照) がある。

1. JSeM: JSeM を用いた NLI 結果の可視化
2. Parsing: 入力文の統語解析、型検査結果の可視化
3. ChartParsing: チャートパーシングの計算過程の可視化

メニュー画面を用意したことによって分析対象の解析結果に直接アクセスすることが可能になった。

本研究で行った主な実装のうち、4 つの機能・特徴をそれぞれ説明する。

3.1 入力文の統語・意味情報の可視化

入力文に対して、石嶋ら [8] の実装した CCG 統語構造の一部の折り畳み/展開機能に加えて、DTS 意味表示の一部の折り畳み/展開機能も実装した (図 1 参照)。これら一部の折り畳み/展開機能はそれぞれ - ボタンと + ボタンで行う。初期状態は親ノード以外全て畳まれている状態で表示され、+ ボタンを押すとその一つ下の子ノードが展開される。- ボタンを押すと、その子ノードが折り畳まれる。

ユーザは express にアクセスすると、入力文とビーム幅 (複数の解析結果の出力のうち、上位いくつ出力するか) をフォーム形式で指定することで入力文の統語構造と意味表示を beam 数だけ得ることができる。この実装により、展開したい部分構造を、他の構造を折りたたんだままの状態でも展開できるため、より効率的に調査したい部分構造に到達できるようになった。

3.2 型検査証明図の可視化

入力文の意味表示に対応する型検査クエリの型検査証明図を表示する機能を実装した。型検査クエリ (Type Check Query) とは、証明の対象となる文の意味表示が型として整合であるかという問いのことである。3.1 節と同様に、- ボタンと + ボタンを実装したこと

¹<https://github.com/DaisukeBekki/lightblue/>

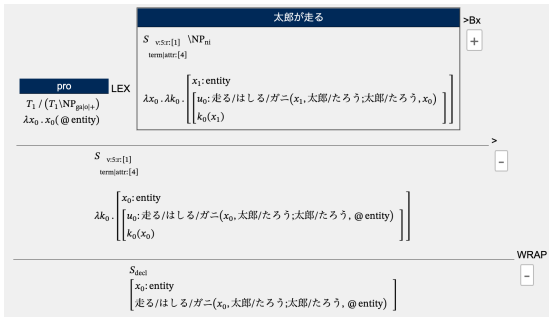


図 1: 「太郎が走る」の解析結果の一部を折り畳んだ構造

よって、図 2 のように証明図の一部を折り畳み/展開することが可能となり、型検査の詳細なプロセスが、視覚的に明瞭になり、全体像を素早く把握できるように改善された。

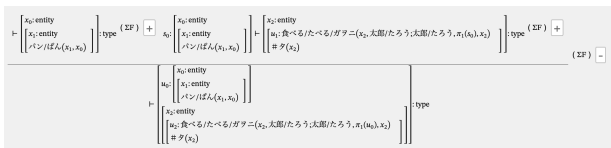


図 2: 「太郎がパンを食べた」の型検査証明図の一部を折り畳んだ証明図

3.3 型検査失敗時の表示

型検査は、証明ができない場合は失敗とみなされる。本研究では、入力文の型検査に失敗したとき、証明が失敗した型検査クエリを特定しやすくするインタフェースを実装した。この実装結果を図 3 に示す。lightblue の型検査クエリ (図 3 の下部の Type Check Query) は統語解析結果の数だけ存在し、型検査は型検査クエリそれぞれに対して行われる。そこで、複数の型検査クエリをタブ形式で切り替えて確認する機能を追加した。加えて、型検査に失敗している、つまり型検査の証明図 (Type Check Diagram) が見つからなかった型検査クエリに対しては、タブの色を赤色にするよう実装した。この実装により、型検査に失敗しているときにそれがどの型検査クエリによるものなのかを特定しやすくなり、デバッグの効率化が見込まれる。

3.4 推論証明図の可視化

石嶋ら [8] は, JSeM (Japanese semantic test suite: 日本語意味論データセット)[12] の各データに割り当てられた ID を入力するとその JSeM データの前提文と仮説文の統語構造を可視化する機能を実装した。本手法では、前提文と仮説文の統語構造の可視化に加え、推論結果の証明図の表示機能の実装を実現した。

この実装では、推論結果の証明図は、型検査証明図と同様に一部を折り畳み/展開することができる。これによって、一目で推論証明図の構造がわかるようになり、証明の流れや文法規則が直感的に理解しやすくなった。推論証明図の可視化結果は付録図 6 に示す。

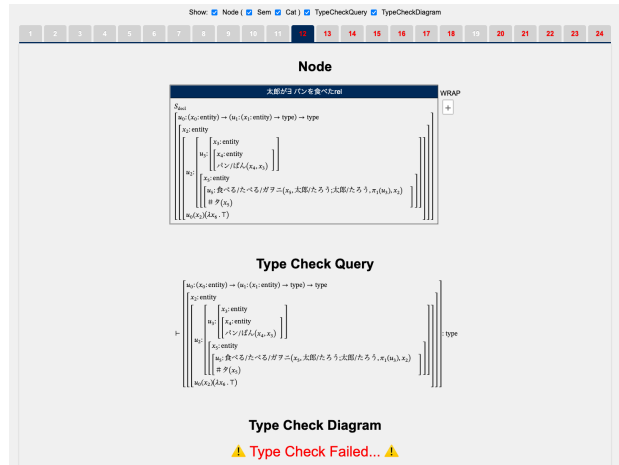


図 3: Parsing 画面の解析結果表示

4 おわりに

本研究では、Web アプリケーションフレームワーク Yesod を用いて、GUI を備えた lightblue 開発システム、express を改良した。本システムを用いることで、チャートパーズング過程や型検査、NLI の証明図の可視化が可能になった。加えて、文法開発に関わるユーザの解析効率や解析結果の視認性を上げるような、証明図一部の折り畳み/展開機能や、型検査失敗時にはそのことを表示する機能を実装した。

謝辞

本研究の一部は、JSPS 科研費 JP23H03452 の助成を受けたものである。

参考文献

[1] Daisuke Bekki and Ai Kawazoe. Implementing variable vectors in a CCG parser. In *Logical Aspects of Computational Linguistics. Celebrating 20 Years of LACL (1996-2016)*, pp. 52-67, Berlin, Heidelberg, 12 2016. Springer Berlin Heidelberg. / [2] Daisuke Bekki and Koji Mineshima. Contextpassing and underspecification in dependent type semantics. In Stergios Chatzikyriakidis and Zhaohui Luo, editors, *Studies of Linguistics and Philosophy*, pp. 11-41. Springer International Publishing, 2017. / [3] Per Martin-Löf. *Intuitionistic Type Theory*, Vol. 17. Bibliopolis, 1984. / [4] Hinari Daido. DTS の部分体系を用いた定理自動証明器への等号型の導入. PhD thesis, お茶の水女子大学, 2022. / [5] Hinari Daido and Daisuke Bekki. Development of an automated theorem prover for the fragment of dts. In the 17th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS17), 2020. / [6] 富田朝, 谷中暁, 戸次大介. 言語学的に妥当な日本語 CCG ツリーバンクの構築と評価. 言語処理学会第 30 回年次大会 (NLP2024), 2024. / [7] 松原舞, 富田朝, 戸次大介. CCG 統語解析器 lightblue と定理証明器 wani による JSeM Verbs データセットの自動推論. 言語処理学会第 31 回年次大会 (NLP2025), 2025. / [8] 石嶋暁, 戸次大介. Yesod による日本語 CCG パーザ開発環境の構築. 人工知能学会全国大会論文集 第 35 回全国大会 (2021), 2021. / [9] Mark Steedman. *Surface Structure and Interpretation*. The MIT Press, Cambridge, 1996. / [10] Mark Steedman. *The Syntactic Process*. MIT Press, 2000. / [11] Daisuke Bekki and Miho Satoh. Calculating projections via type checking. In *Proceedings of the TYTTLES Workshop on Type Theory and Lexical Semantics, ESSLLI*, 2015. / [12] Ai Kawazoe, Ribeka Tanaka, Koji Mineshima, and Daisuke Bekki. An inference problem set for evaluating semantic theories and semantic processing systems for japanese. In the 17th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS12), pp. 67-73, 2015.

A 付録

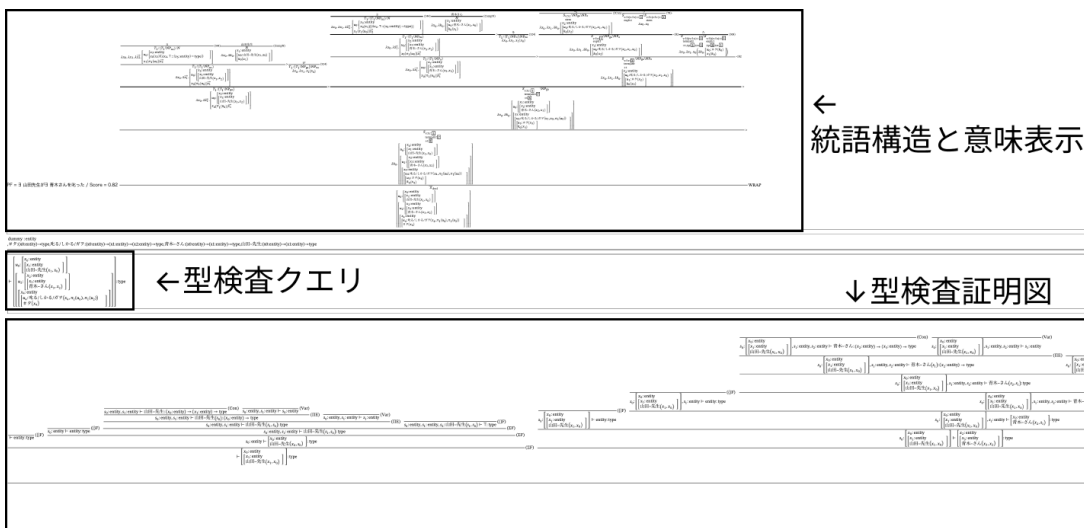


図 4: 可視化前の lightblue による HTML 出力のキャプチャ画面

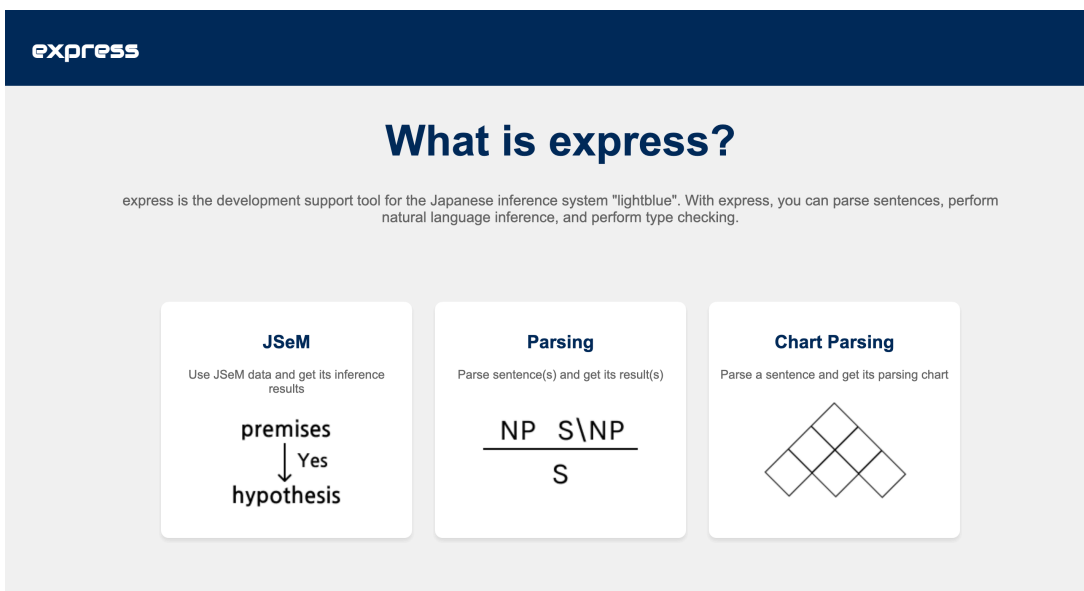


図 5: express のメニュー画面

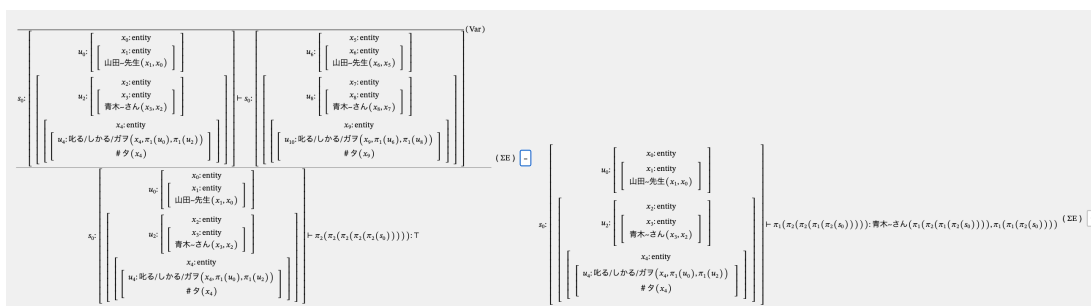


図 6: 前提文「山田先生が青木さんを叱った。」と仮説文「青木さんが山田先生に叱られた。」における推論証明図の一部を折り畳んだ構造