実テキストにおける数量表現の含意関係認識に向けて

小谷野 華那（指導教員：戸次大介）

1 はじめに

実テキストを対象とした意味解析において、複数文書における記述の差分を自動で計算するタスクには一定の社会的ニーズがある。近年では、このタスクに対して、含意関係認識器 ccg2lambda[2] を用いたアプローチが試みられている [5]。

現在、戸次研究室は日本電気株式会社と共同研究を行なっており、捜査文書内のテキストの意味解析に取り組んでいる。この研究では、被疑者と被害者の供述調書が与えられたときに、意見の食い違いを自動で計算することを目指している。例えば被疑者が「私は、酒を飲んでいた男を鉄パイプで 2、3 回殴った」と供述しており、被害者が「私が酒を飲んでたら、後方から男が鉄パイプで 10 回以上殴ってきた」と述べているとする。このとき、意見が食い違っている点としては、殴った回数であり、また被害者のみが「後方から」と述べている。このような食い違いを自動で抽出することを目指す。これを計算するには、双方向の含意関係を判定すること、そして含意関係の計算を行ったときに証明できなかった項の情報が有益だと考える。

2 先行研究

本研究では、含意関係の判定のために ccg2lambda を用いる。ccg2lambda は、前提文と仮説文が与えられたときに、構文解析、意味解析、定理自動証明を行うことで、前提文が仮説文を含意しているかを判定する。日本語の CCG 構文解析器として、depccg[6] や jigg[3] があげられる。

前提文として「私は鉄パイプで殴られた」、仮説文として「私は鉄パイプで後方から殴られた」を入力文とすると、文が単語ごとに区切られたのち、構文解析によって木構造で表現される。そして各単語に対応する λ 式を合成していくことで、文の意味を表す論理式が導出される。最後に 2 つの論理式の含意関係を定理証明支援器 Coq[4] を用いて判定を行う。この例では前提文が仮説文を含意するかを計算すると unknown という出力になり、前提文と仮説文を交換すると yes という結果が出力される。

Yanaka ＋ [5] は ccg2lambda で含意関係を判定する際に証明できなかった項を抽出することで、前提文と仮説文の差分を抽出している。上記の前提文・仮説文を入力文とすると、サブゴール（未証明項）として「後方」・「から」に対応する論理式が出力される。

捜査文書には数量表現を含む文が頻繁に出現する。例として年齢表現を含む文があげられる。前提文として「22 歳の男がいる」、仮説文として「22 、3 歳の男がいる」の含意関係を計算すると yes となるべきである。しかし、ccg2lambda でこれを計算すると unknown と出力されてしまう。

ccg2lambda が数量表現を含む文を処理したときの問題点は 2 つある。1 つ目は、意味テンプレートにおいて助数詞の λ 式が定義されていないことにより、文の意味を表す論理式に助数詞の情報が含まれていないことである。数量表現を処理する際に助数詞の情報が付与されていないと、以下のような問題が生じる。

例えば、「22 人の男がいる」と「22 歳の男がいる」の 2 文について含意関係を計算した場合に、unknown と出力されるべきところで yes と出力されてしまう。このような問題を解決するために、論理式に助数詞の情報を付与する必要がある。

2 つ目の問題点は、木の合成の順番である。ccg2lambda で「22、3 歳」を処理したとき、[22、][3 歳] という順番で合成されている。しかし、「22、3」の部分で、数字は「22 または 23」を表し、数字の単位が「歳」であるという情報を持つ論理式になるべきである。現状の論理式では「22 かつ 3」を表す論理式になっているため、出力される論理式を適切なものに書き換える必要がある。

3 研究目的

本研究では、数量表現を含む文間の含意関係認識を正しく行えるようにすることを目的とし、構文木・意味表示の書き換えを行う手法を提案する。また、書き換えた意味表示において含意関係の計算を正しく行うための数字の扱いについて検討する。

4 数量表現のための ccg2lambda の改良

数量表現を含む文の含意関係認識を正しく行うために、以下の点について ccg2lambda を改良した。

4.1 構文木の書き換え

ccg2lambda が出力する構文木は、数字どうしが先に合成されるのではなく、図 1（次項）のように「22、」・「3 歳」がそれぞれが先に合成されている。本来、数字どうしが先に合成され、「22 または 23」という情報を持ち、その数字の単位が「歳」であるとすべきである。構文木をそのように書き換えるために、本研究では、Tsurgeon[1] を用い、図 1 の構文木は図 2（次項）のように書き換えることで、数字どうしが先に合成されるようにした。ccg2lambda の処理の流れは図 3（次項）の通りである。

4.2 意味表示の書き換え

書き換えた構文木に対応して意味表示を付け加える必要があるため、意味テンプレートの加筆修正を行った。ccg2lambda での助数詞の扱いは、⟨ 数字 ⟩ + ⟨ 助数詞 ⟩ の統語範疇が NP である時と NP/NP、S/S である時によって扱いが異なっている。本研究では、⟨ 数字 ⟩ + ⟨ 助数詞 ⟩ の統語範疇（NP 等）を問わず、⟨ 助数詞 ⟩(x) = ⟨ 数 ⟩ という情報を持った意味表示に書き換えを行った。書き換えた意味表示は図 2 の通りである。

4.3 XML と Coq

ccg2lambda で文を処理した時、形態素解析によって文は単語ごとに分解される。各単語を Coq の述語名として扱うために単語の前に「_」がついた形で XML ファイルに情報が書き込まれるが、数字についても同

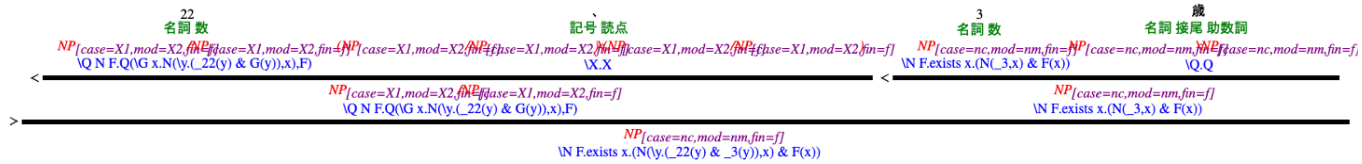

図 1: ccg2lambda の出力

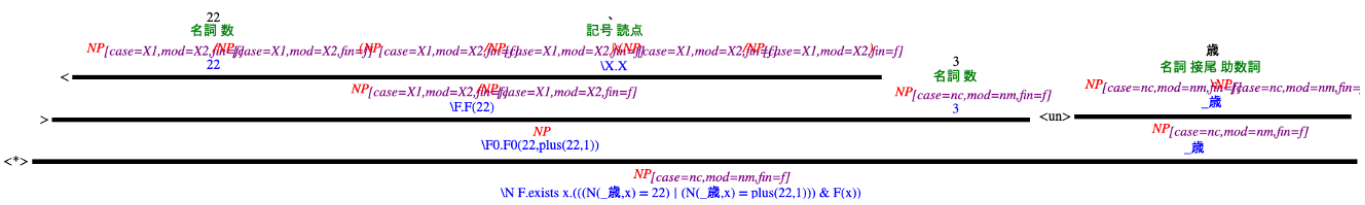

図 2: 修正した構文木と意味表示

様の処理が行われるため、数ではなく述語名として扱われる。しかし、Coq で数字の比較をするために等号や不等号、四則演算を行うためには、数字を述語名ではなく数として扱う必要があり、「_」を取り除かなくてはならない。したがって、XML ファイルに書き込まれている各単語の情報について、数字のみ「_」を取り除くようにプログラムを追加した。これにより、「22 歳」が「22、3 歳」を含意することを証明することができるようになる。

5 現状

現状として、「22、3」といった数字の間に区切り記号が入る表現について構文木・意味表示の書き換えを行い、正しく証明ができるようにした。構文木に関しては数字どうしが先に合成されたのち、助数詞が合成されるようにした。意味表示については、「、」の前後の数字において後ろの数字は前の数字に 1 を足した数であるとすることで、「22、3」から「23」の情報を取得できるようにした。さらに数字と単位の関係がわかるように「⟨助数詞⟩(x) = ⟨数⟩」という形の式に書き換え、「22、3 歳」については「歳 $(x) = 22$| 歳 $(x) = 23$」という式に書き換えを行った。証明については、数字を述語名ではなく数として扱う（「_」を取り除く）ことで、等号や不等号、四則演算を含む論理式の含意関係を証明できるようになった。「23 歳の男がいる。」が「22、3 歳の男がいる。」を含意していると正しく判定できることも確認済みである。

6 おわりに

本研究では、ccg2lambda の数量表現の含意関係認識において、数字の間に区切り記号が入る表現について改良を行った。

今後の課題として、金額の表現（「50 万円」など）に多く見られる算用数字と漢数字が混在している表現や「何回」・「数回」といった、量化を含む数の表現についても含意関係の判定を正しく行えるようにしていく。

図 3: Tsurgeon を加えた ccg2lambda の処理

参考文献

[1] Levy, R. and Andrew, G.: Tregex and Tsurgeon: tools for querying and manipulating tree data structures, *In 5th International Conference on Language Resources and Evaluation*, pp. 2231–2234 (2006).

[2] Mineshima, K., Martínez-Gómez, P., Miyao, Y. and Bekki, D.: Higher-order logical inference with compositional semantics, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 2055–2061 (2015).

[3] Noji, H. and Miyao, Y.: Jigg: A framework for an easy natural language processing pipeline, *Proceedings of ACL-2016 System Demonstrations*, pp. 103–108 (2016).

[4] Team, T. C. D.: *Coq Proof Assistant: Reference Manual: Version 8.9.0*, INRIA (2019).

[5] Yanaka, H., Mineshima, K., Martínez-Gómez, P. and Bekki, D.: Determining Semantic Textual Similarity using Natural Deduction Proofs, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing,*, pp. 681–691 (2017).

[6] Yoshikawa, M., Noji, H. and Matsumoto, Y.: A* CCG Parsing with a Supertag and Dependency Factored Model, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 277–287 (2017).