クラウドソーシングを用いた日本語述語項構造タグ付きコーパスの拡張

阿部 航平 † 河原 大輔 † 黒橋 禎夫 †‡
† 京都大学 ‡ 科学技術振興機構 CREST
{abe, dk, kuro}@nlp.ist.i.kyoto-u.ac.jp

1 はじめに

意味解析の基本的なタスクの一つとして述語項構造解析がある。これは、文章中の各述語に対して「誰が何を何にどうした」のようにそれぞれの格に当てはまる項を同定するタスクである。格には様々あるが主にガ格、ヲ格、ニ格が解析の対象とされている。例として、次の文章に対して述語項構造解析を行うことを考える。

(1) 太郎はジュースを買った。そして帰る前に飲んだ。

この文章では、述語「買った」に関する述語項構造は「太郎がジュースを買う」であり、述語「飲んだ」に関する述語項構造は「太郎がジュースを飲む」である。

述語項構造解析は、係助詞句や被連体修飾詞の項の格を同定する格解析と、存在が明示されていない項の先行詞を同定する省略解析に大別される。上記の例では、「太郎は」と「買った」の関係は格解析され、「飲んだ」のガ格とヲ格は省略解析される。日本語では項の省略が頻繁に起こるが、省略解析の精度は60%弱にとどまっており、重要なタスクの一つになっている。

近年、述語項構造解析に対して深層学習を用いたモデルが多く考案されている。モデルの学習には、人手で述語項構造タグを付与したコーパスが用いられているが、それらの作成には膨大な時間的、金銭的コストがかかる。解析精度向上の一つの方法としてタグ付きコーパスの大規模化が考えられるが、コストが大きな問題となる。本研究ではクラウドソーシングを用いた大規模化の方法を提案し、それを用いたコーパス拡張および解析の結果を示す。

2 関連研究

日本語述語項構造タグ付きコーパスとして、以下の3つが代表的なものとして用いられている。京都大学テキストコーパスおよびNAISTテキストコーパス [1] は毎日新聞の記事をコーパスとして利用し、形態素・構文情報のほか述語項構造や共参照などの意味情報タグを付与している。また京都大学ウェブ文書リードコーパス (KWDLC) [2] では、多様な文体や内容の収集を目指し、ウェブドメインを対象として同様の意味情報タグの付与を行っている。上記のコーパスは数千〜数万文の文章で構成されているが、近年の述語項構造解析では深層学習を活用したモデルが利用されていることから、学習データ量の増加がさらなる精度向上に寄与すると考えられる。しかし、コーパスのタグ付けは少人数の専門家によって行われており、これには膨大なコストがかかるため大規模コーパスを作成するのは難しい。

上記のコーパスを用いた日本語述語項構造解析の研究は長く続いているが、近年は特にフルニューラルのモデルが多く提案され、それ以前の統語的な特徴量を用いたモデルやニューラルとの複合モデルなどを超える精度を達成している。例えば、複数の述語が項を共有することが多いことに着目し文章中の述語について同時学習を行ったモデル [3] や、self-attention 等を用いた中間層で複数述語間の情報共有をさらに改良したモデル [4] による精度向上が見られる。また、entity の概念を学習に取り入れ共参照との同時学習を行ったモデル [5] や、画像生成などで主に用いられている敵対的学習を組み込むことで生コーパスを学習の補強に用いるモデル [6] なども精度向上を達成している。

非専門家による協力を受けて意味的タグ付けを行った研究の例としては以下のものが挙げられる。飯田ら [7] は、一般的な日本語話者 8 人による省略関係タグの特性を調査し、NAIST テキストコーパスとの比較を行っている。FitzGerald ら [8] は、述語項構造解析と類似した意味役割付与 (SRL) タスクを対象としてクラウドソーシングを用いてコーパスを作成した。この研究では、日常的に言語を使用している人々に短い

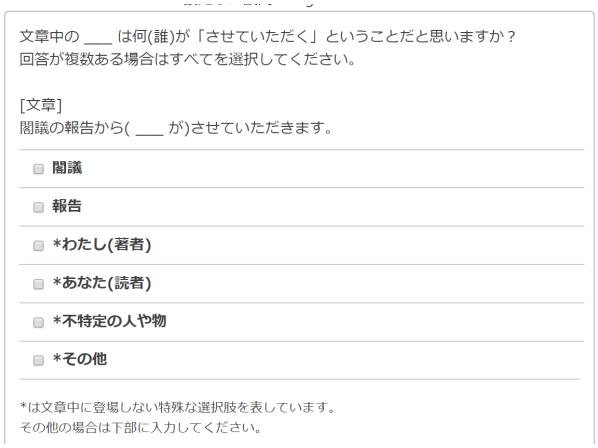

図 1: ガ格の省略についてのタグ付け作業画面

インストラクションを施すことによって比較的に高い精度でタグ付けを行うことができたと示している。日本語の述語項構造タグ付きコーパスについては、高橋ら [9] が運転ドメインを対象として、述語項構造解析および文章読解のコーパスをクラウドソーシングを用いて構築し、その検証を行っている。

3 拡張コーパスの構築方法

本研究では、大規模な述語項構造タグ付きコーパスを低コストで構築するためにクラウドソーシングを用いる。日本語話者は省略表現を自然に用いていることから、クラウドソーシングでも比較的正しいタグを付与できると考える。以下の節では、タグ付けのデザインおよびクラウドソーシングによる具体的なタグ付けについて述べる。

3.1 タグ付けのデザイン

日本語述語項構造解析で対象とされる格には主にガ格、ヲ格、ニ格の 3 種類がある。本研究では、もっとも高頻度なガ格に絞ってコーパス拡張および解析を行う。

クラウドソーシングにおける各問題は、ある述語の省略されたガ格について先行詞を問う問題とする。文章中には複数の述語が存在するため、それぞれに対応する問題を生成する。問題画面の例を図 1 に示す。先行詞の選択肢は複数選択可能なものとする。

クラウドソーシングのための文書の前処理および問題とする述語の選定は、Juman++および KNP による形態素・構文・格解析および齋藤ら [10] が提案したイベントグラフを用いる。構文・格解析器 KNP は構文解析を 93%程度、格解析を 90%程度の精度で行うことができる。そこで、KNP の構文・格解析結果を信頼し、ガ格が省略されていると判定された述語についてのみ問題を生成する。先行詞の選択肢としては、自動解析の結果から文章中の名詞句を抽出することによって生成する。なお、先行詞が述語より後方の文に出現する頻度は少ないことから、ワーカーの負担を軽減するために、対象となる述語を含む文以降の名詞句は選択肢から除外する。文章中の名詞句に加えて、照応先が文章に出現しない外界照応用の選択肢を含める。本研究では「著者」「読者」および「不特定」の 3 つを外界照応に関する項の候補として用いる。

3.2 クラウドソーシングによるタグ付け

コーパスの構築は Yahoo!クラウドソーシングを用いて行う。各クラウドワーカーは、数個の例文を含むタスク説明を受けた後、1 セット 10 問の問題に対して回答する。一つの問題に対して 10 人のワーカーから回答を募り、最大投票数を得た回答をタグ付け結果として採用する。

4 既存のタグ付きコーパスとの比較

4.1 クラウドソーシングによる比較用コーパスの構築

クラウドソーシングで得られるタグと、十分なインストラクションやタグ付けのすり合わせなどを経て作成された専門家による既存のコーパスのタグを比較することにより、この方法で得られるタグ付きコーパスの特性について調査をした。本研究では前章で述べたとおり、ガ格、特に省略解析に関して実験および分析を行う。比較用のタグ付きコーパスは KWDLC を用い、テスト用データから抽出された省略解析に関する問題のうち、ランダムに 900 個を選んだものをクラウドソーシングでタグ付けした。

クラウドソーシングによって集めたタグの信頼度を示すと指標として、ワーカーの票の集まり具合を利用する。信用する得票数の閾値を決め、最多得票の選択肢の得票数が閾値を超えた場合のみそのタグを利用する。(例えば、閾値が 5 で最多得票数が 4 の場合、その回答は用いずタグ付けは行わない。)

4.2 コーパスの比較結果

図 2 は、最多得票数の閾値ごとの回答一致率、および採用されるタグ数を示したものである。5 票以上が集まった回答では、一致率は 72.0%であった。閾値を上げることで一致率が上がることから、信頼度の指標として用いることができると考えられる。しかし、閾値上げると得られるサンプル数は少なくなる。

	文章	gold	crowd	正誤
1	桐学舎の講師陣が責任を持って指導を行います。生徒ひとりひとりの 特性を**考慮した**指導で卒業までしっかりとサポートしていきます。	陣	陣	○
2	すっかり暖かくなってきた昨今、体を動かす絶好の**チャンスだ**。	昨今	著者	×
3	ご旅行の際ご利用いただく航空会社は予告なしに**変更される**場合があります。	航空会社	航空会社	○
4	今日も何も口にしていません。病院から戻ると疲れたのか、**眠ってしまいました**。	不特定	著者	×
5	アトマイザーはその性質上、次第に汚れや焦げなどが付着していきます。 これを**放置しておく**と、だんだんと煙量が少なくなったり、味が落ちたりします。	不特定	読者	×

表 1: クラウドソーシングにより得られたガ格省略のタグ付け例

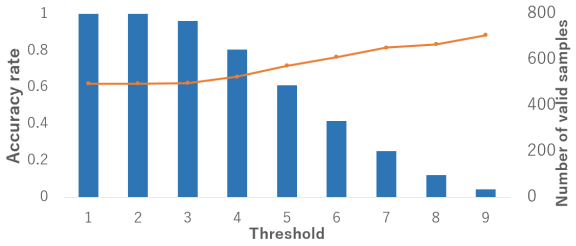

図 2: 最多得票数の閾値と正解との一致率。横軸を n として、折れ線グラフは n 票以上獲得した回答のみの一致率、棒グラフは得られたサンプル数を表す。

表 1 に、閾値 5 以上としたときにクラウドソーシングによって得られた回答の例を示す。「gold」と「crowd」はそれぞれ専門家とクラウドワーカーによるタグを表している。

例 1 では、文をまたぐ省略に関しても正しく答えられている。(「陣」は、「講師陣」を形態素解析した際の末尾の名詞としてタグ付けされている。) 例 2 では、「昨今がチャンスだ」という述語項構造が正しいが、誤ってガ格が著者とタグ付けされている。例 3 では、何が「変更される」と思いますか？と聞くことでワーカーが受身を認識して考え、正しい答えを選択できている。一方、外界照応については多少の曖昧性があり、外界照応のトークン間での食い違いが見られる。特に、正解が不特定の場合で著者や読者が選ばれるパターンが複数存在する (例 4, 例 5)。例 4 については、文の曖昧性を減らすために後方に続く文章も表示することが有効だと思われ、今後の改善点とする。

5 拡張コーパスを用いた述語項構造解析

5.1 タグ付きコーパスの拡張

クラウドソーシングでタグ付けを行ったコーパスをモデルの学習に用いることの効果を検証するため、生コーパスを用いてクラウドソーシングによるタグ付けを行った。クラウドソーシングを利用してタグ付けを行った追加のコーパスを**拡張コーパス**と呼ぶ。生コーパスの収集は KWDLC と同様の手法で行い、ウェブから収集した文章から先頭 3 文を抽出したものを使用する。3,016 文書から生成した 7,012 問についてクラウドソーシングを行った。クラウドソーシングにかかった時間は約 3 日間、金額は約 70,000 円であった。

5.2 述語項構造解析実験の設定

述語項構造解析のベースラインモデルとして、BERT [11] と呼ばれる様々な自然言語処理のタスクに応用されている手法を用いる。BERT は 2 つのステップで学習を行う手法であり、大量の生コーパスを用いて行う事前学習と、fine-tuning と呼ばれる適用するタスクに応じた学習に分かれる。本実験においては、fine-tuning に相当する層では全結合層を用いたスコア計算を行い、最もスコアの大きい選択肢を目的の項とする。

具体的な方法を以下に示す。選択肢としては、入力文章をサブワードに分割したトークン列、外界照応を表す [著者]、[読者]、[不特定]、および当てはまる項がないことを示す [NULL] を用意する。ある選択肢が目的の項であるスコアは以下のように計算する。

$$score_{case} = v^{\mathrm{T}} \tanh(W_{case} h_{pred} + U_{case} h_{arg}) \quad (1)$$

ただし、h_{pred} および h_{arg} は、BERT に各トークンを入力して得られた最終層の出力のうち、述語および着目している選択肢を表現するベクトルである。また W_{case} および U_{case} はそれぞれの格について存在する学習パラメータ、v はスカラー値を出力する全結合層である。

本研究ではさらに、Shibata ら [5] に倣った共参照との同時学習モデルでも実験を行う。式 (1) と同様にあるトークンからすべてのトークンおよび [NULL] トークンに対する共参照スコアを計算し、最もスコアの高い項を共参照先として推定する。

すべての実験に共通して、BERT には日本語 Wikipedia 1,800 万文で pre-training した BERT-LARGE を使用する。fine-tuning は KWDLC と拡張

	ガ格		ヲ格		ニ格	
	格解析	省略	格解析	省略	格解析	省略
[Shibata, ACL2018]	**0.945**	0.646	**0.857**	0.343	0.411	0.465
BERT	0.893	0.686	0.772	**0.433**	0.540	0.478
+クラウド (5 票以上)	0.883	0.710	0.731	0.377	0.529	0.509
+クラウド (6 票以上)	0.873	0.700	0.750	0.408	**0.549**	0.488
BERT+coref	0.895	0.720	0.781	0.417	0.451	0.524
+クラウド (5 票以上)	0.894	0.715	0.746	0.428	0.503	0.527
+KWDLC fine-tuning	0.893	**0.724**	0.742	0.430	0.519	**0.532**

表 2: 拡張コーパスを学習に利用した述語項構造解析の結果。“クラウド “は拡張コーパスによってガ格省略関係のサンプルを増やした状態、“coref” は共参照解析との同時学習、“KWDLC fine-tuning” は後半 3 エポックで KWDLC のみを用いて学習したことを表す。

コーパスを混合したものを用い、4 エポック行う。精度は KWDLC のテスト用データを用いて F 値で評価する。

5.3 拡張コーパスを用いた述語項構造解析

拡張コーパスを追加して実験を行った結果を表 2 に示す。coref は共参照の同時学習を、KWDLC fine-tuning は拡張コーパスを含めた 1 エポックの学習に続き、KWDLC の訓練データのみを用いた 3 エポックの学習を行ったことを示す。

coref を用いない場合、ガ格省略に関しては既存コーパスのみの学習でも先行研究を上回ったうえ、拡張コーパスを用いることでさらに精度を向上させることができた。ただし票数の閾値を 6 票に変化させても精度の向上が確認できず、拡張コーパスの質と学習に用いることのできるサンプル数がトレードオフになっていると考えられる。coref を同時学習した場合、ベースラインはさらに高い精度を示したものの拡張コーパスによる寄与は見られにくい。これは拡張コーパスには共参照に関するタグがないこと、質が完璧でないことから解析精度の上限が抑えられてしまうことが考えられる。このような欠点は KWDLC のみを用いた fine-tuning を行うことで、ある程度解消することが確認できた。

6 おわりに

本研究では、述語項構造解析、特に省略解析についてのタグ付けをクラウドソーシングを用いることで効率的に行う手法を提案し、その効用を検証した。今後の課題として、クオリティコントロール等、拡張コーパスの質の向上への取り組み、およびヲ格・ニ格に関しての同様の拡張、検証を行う予定である。また、本コーパスは一般に公開する予定である。

7 謝辞

本研究は科学技術振興機構 CREST「知識に基づく構造的言語処理の確立と知識インフラの構築」（JPMJCR1301）の支援のもとで行われた.

参考文献

[1] 飯田龍, 小町守, 井之上直也, 乾健太郎, 松本裕治. 述語項構造と照応関係のアノテーション: Naist テキストコーパス構築の経験から. 自然言語処理, 2010.
[2] 萩行正嗣, 河原大輔, 黒橋禎夫. 多様な文書の書き始めに対する意味関係タグ付きコーパスの構築とその分析. 自然言語処理, 2014.
[3] Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. Neural modeling of multi-predicate interactions for japanese predicate argument structure analysis. In *ACL*, 2017.
[4] Yuichiroh Matsubayashi and Kentaro Inui. Distance-free modeling of multi-predicate interactions in end-to-end Japanese predicate-argument structure analysis. In *COLING*, 2018.
[5] Entity-centric joint modeling of japanese coreference resolution and predicate argument structure analysis. In *ACL*, 2018.
[6] Shuhei Kurita, Daisuke Kawahara, and Sadao Kurohashi. Neural adversarial training for semi-supervised Japanese predicate-argument structure analysis. In *ACL*, 2018.
[7] 飯田龍, 橋本力, 鳥澤健太郎, 黒橋禎夫, 乾健太郎, 宮尾祐介, 柴田知秀, 笹野遼平. 日本語書き言葉を対象とした人間の自然な省略検出の分析. 言語処理学会第 21 回年次大会発表論文集, pp. 565–568, 2015.
[8] Nicholas FitzGerald, Julian Michael, Luheng He, and Luke Zettlemoyer. Large-scale qa-srl parsing. In *ACL*, 2018.
[9] 高橋憲生, 柴田知秀, 河原大輔, 黒橋禎夫. ドメインを限定した機械読解モデルに基づく述語項構造解析. 言語処理学会 第 25 回年次大会, 名古屋, 2019.3.
[10] 齋藤純, 坂口智洋, 柴田知秀, 河原大輔, 黒橋禎夫. 述語項構造に基づく言語情報の基本単位のデザインと可視化. 言語処理学会第 24 回年次大会, 2018.
[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, 2019.