推論の根拠を提示できる矛盾検出テストセットの研究

研究代表者　土屋雅稔　国立大学法人豊橋技術科学大学　大学院工学研究科　教授
共同研究者　内山将夫　国立研究開発法人情報通信研究機構
先進的音声翻訳研究開発推進センター　上席研究員

1 はじめに

インターネットの発展に伴い，現代社会は，過去とは比べ物にならないほど情報通信が増大している．古典的な社会モデルでは，情報通信の増大によって人と人の相互理解は深まると期待されていたが，現実のソーシャルネットワークサービスでは，ノイジーマイノリティによる信頼性が低い発言が目立つようになり，分断が進行している．これは，情報通信の増大によって，一人の人間が取得できる情報は飛躍的に増大しているのに対して，取得した情報が信頼に足るものであるか否かを判断する能力には大きな変化がなく，増大した情報を咀嚼しきれていないことが原因の一つと考えられる．このような状況に対応するには，計算機によって情報の信頼性の判定を支援する技術が有用と考えられる．

ある情報が信頼できるか否かを判定する課題（ファクトチェック）は，現在，広い関心を集めている課題である．ファクトチェックは，判定対象となる情報 A が，既知の複数の信頼できる情報 B,C,D...と矛盾せず，整合的であるか否かに基づいて行われる．ある情報がテキスト（複数文）によって表現される場合，この課題は，文 1 が文 2 を含意しているか，それとも矛盾しているかを判定する問題に還元される．このような文間の論理的関係の判定問題は，自然言語処理分野においては含意関係認識という名前で知られており，自然言語理解の基礎的な問題設定として重要な問題と考えられている[乾 2008]．含意関係認識は，与えられた 2 つのテキスト間の論理的関係を分類する問題である．例えば，以下の 2 文が入力された時，矛盾という含意関係ラベルを自動的に出力する問題である．

t1 川端康成は,「雪国」などの作品でノーベル文学賞を受賞した.
t2 梶井基次郎は,「雪国」の作者である.

含意関係ラベルとしては，含意・矛盾・中立の 3 種類を想定することが一般的である．

大規模データセットの整備[Bowman 2015, Marelli 2014]と深層学習の発展にともなって，大規模言語モデルが各種の自然言語推論を解くことができるという報告が相次いでいることから，深層学習モデルを用いて情報の信頼性を判定するという着想は，非常に自然なものである．しかし，深層学習モデルの適用による精度向上は，データセットの作成作業者の偏りによる見かけの精度向上ではないかという指摘が相次いでいる[Tsuchiya 2018, Geva 2019]．加えて，深層学習モデルは，内部で行っている推論の根拠を可視化することが困難であるだけでなく，結果の予測も困難であるという問題点が指摘されている．また，深層学習モデルは，学習に用いたコーパスの偏りによって予想外の挙動を示すことがあるため，深層学習による推論結果を無批判に利用すると，偏りを再利用してしまう危険性がある．このような危険を避けるためには，推論結果の根拠が示されることが重要である．

矛盾は，他の含意関係ラベルに比べて，データセットを作成する上で非常に難しい問題を含むラベルである．既存の大規模データセットに含まれる偏りを避けるには，作業者に仮説文の作成を依頼するのではなく，既に存在する仮説文に対する含意関係ラベルの付与を依頼する必要がある．しかし，矛盾する文対を自動収集することは容易ではない．Wikipedia を対象として言語横断含意関係データベースの構築した場合，矛盾している文対は 10%未満しか得られていない[Hayakawa 2016]．また，同じく Wikipedia の日時の異なる版を対象として，矛盾する文対を収集した場合も，矛盾している文対は 6%未満だった[Tsuchiya 2022]．また，SICK コーパスにおいても，含意が 28.7%出現しているのに対して，矛盾は 14.6%とほぼ半分である[Marelli 2014]．このように， 2 つの文が矛盾しており，かつ，その矛盾がきちんと根拠や論理をもって説明できる文対を自動収集することは，きわめて困難である．

以上の 2 点から，本研究では，推論の根拠を提示できる矛盾検出テストセットの構築に取り組む．テストセットを開発する手順は 4 段階からなる．第 1 に，Wikipedia テキストを対象として，出典が明示されてい

公益財団法人電気通信普及財団
研究調査助成報告書　第39号　2024年度

る文を収集する．これらの文は、その文の記述内容に関する根拠が、出典テキスト中に記載されていると期待される．Wikipedia で用いられる典拠文書には、書籍やウェブページがあるが、本研究では、研究時点でアクセス・内容を容易に確認できるウェブページに限定する．第 2 に，共同研究者が研究開発した機械翻訳技術に基づく言い換えモデルを用いて，基準文の言い換えを生成する．当該モデルは，入力文の意味を保存した言い換えを生成することが期待されているが，入力文と矛盾する文を一定の割合で生成してしまう性質を持っており，本研究に適している．第 3 に，作業対象データを整理した上で，基準文と言い換え文の論理的関係をクラウドソーシングにより分類する．第 4 に，矛盾している判定された文対を対象として，当該文対と出典テキストを作業者に提示し，矛盾していると判定するために必要な外部知識が出典テキストに明記されているか判定を依頼する．本稿では，出典が明示されている基準文の収集方法と，出典が明示されている基準文と出典テキストの論理的関係をクラウドソーシングにより判定する場合の問題について報告する．

2 基準文候補の収集

2-1 収集手法

本研究では，根拠情報が必要な基準文を収集するコーパスとして，複数の編集者によって作成されているインターネット上のオンライン百科事典である Wikipedia に注目する．Wikipedia では，複数の編集者による共同作業の品質を担保するため，いくつかの作業方針が事前に宣言されているが，その方針の 1 つに検証可能性がある．検証可能性の作業方針として，Wikipedia では以下の 3 点が示されている．

- 記事には、信頼できる情報源が公表・出版している内容だけを書くべきです。
- 記事に新しい内容を加筆するときは、信頼できる情報源―出典（参考文献）―を明らかにすべきです。出典が明示されていない編集は、誰でも取り除くことができます（出典のない記述は除去されても文句は言えません）。
- 出典を示す義務を負うのは、書き加えようとする側であり、除去を求める側ではありません。

この作業方針から，Wikipedia の記事には，ある出典に基づいて作成された文が含まれているはずである．本研究では，これらの出典に基づく文を対象とする自然言語推論課題を作成することによって，当該文の出典を根拠テキストとして利用できる自然言語推論課題を作成する．

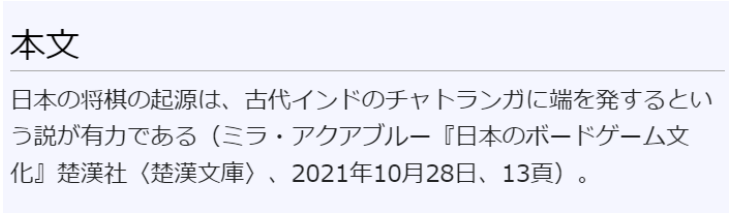
この提案手法を実現するには，Wikipedia の記事テキストを対象として，以下の処理を行う必要がある．

- 記事テキストを文分割する
- 出典情報が付与されている文（基準文候補）を取り出す
- 出典テキスト（根拠テキスト候補）を収集する

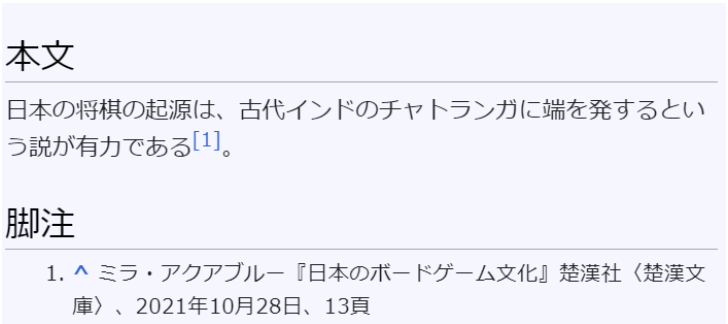
まず、Wikipedia のデータを対象として，出典が付随しているテキストに注目する．出典で示されるテキスト内には，その文を記載するに至った根拠となる文が存在する可能性があるためである．出典で用いられる文献の中には書籍や論文，ウェブページがあるが，本研究ではアクセスやデータ収集が容易なウェブページに限定し，ウェブページからの出典とその出典が存在する文のセットを収集する．

出典の書き込み方として，Wikipedia の編集者向けノートでは，以下の方法が指定されている．第 1 は本文に括弧書きで書き込む方法，脚注機能を用いて書き込む方法，第 2 は専用のテンプレートを用いて書き込む方法，第 3 は 1 箇所にまとめて出典を掲載する方法である．

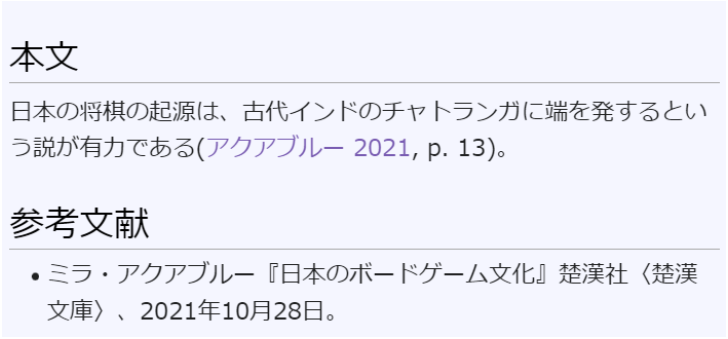
第 1 の方法は，右図のように本文の直後に括弧書きで書き込む方法である．この方法では，特別な記述用のタグは用いられず，括弧の存在によって，出典の存在が示されている．



第 2 の方法は，脚注機能を用いて書き込む方法である．右図のように，脚注機能を用いて，本文から出典部分が分離して表示される．本文中で脚注機能を用いる場合は，<ref>...</ref>というマークアップタグに囲まれた部分に出典の内容が書き込まれ．出典の URL もこのマークアップの中に記述される．



第 3 の方法は，専用のテンプレートを用いて書き込む方法である．右図のように，脚注機能を使用して書き込む際にテンプレートを用いることで，読者などの負担を軽減する方法である．出典情報は，基本的に脚注情報を用いて書き込む形式と同様に，マークアップタグによって表現されているが，脚注のマークアップタグに囲まれた中でさらにテンプレート形式で書かれている点が異なる，出典が URL を含む場合は"Cite web|"に続く形で URL が記載されている．



したがって、これらの出典から URL を取り出し，その出典に対応している文を収集すればよい．最初に，Wikipedia に含まれる記事の本文を対象として，本文を文単位で分割する前に出典を含む箇所にマーキングをする．記事の本文を文単位で分割する前に出典の箇所にマーキングをするのは，文単位で分割をするためにはマークアップ等をすべて外した状態でスクリプトを実行する必要があり、分割の実行後には出典がどこに位置していたのかが分からなくなるためである．ここでは，先に示した出典の書き込み方のうち，脚注機能を用いて書き込まれた出典であり，かつ URL で示されたウェブ上の出典のみを対象にマーキングし、マーキングの記号と出典の記述を紐づけできるように保管する．次に，上記の処理がされた文を文単位で分割することで，マーキングされた文と，マーキングされていない文を得ることができる．マーキングされた文に対して、マーキングの記号と出典の記述を入れ替えることで，出典と出典に付随する文の組を得る．続いて，出典に付随する文を根拠が必要な文として用い，出典であるウェブサイトのテキストを抽出する．最後に，根拠情報が必要な文と出典テキストに関係があるかどうかを比較可能にするために，出典テキストを段落ごとに分割する．手順を整理すると、以下のとおりである．

1. Wikipedia に含まれる記事の本文を対象として，出典を含む箇所にマーキングをする
2. 上記の処理をされた本文を文単位に分割する．
3. 根拠テキストが含まれるテキストと根拠情報が必要な文として，それぞれ出典の URL と出典に付随する文の組を得る
4. 出典の URL に含まれるテキストである出典テキストを抽出する
5. 出典テキストを段落ごとに分割する．

次に，後述のデータセットに，基準文の根拠となる可能性のあるテキスト（根拠テキスト）を出典のウェブページから収集したものを追加する．最後に，根拠テキストと基準文を比較し，関連性があるかどうかを判定する．

2-2 出典を含む基準文候補の収集実験

まず Wikipedia に存在する記事の本文を，句点やピリオドを区切りとして文単位に分割した．次に，出典が付随する文として，分割された文の中から出典を含み、かつ資料がウェブページである文のみを抽出した．本研究では，この文を，基準文と呼ぶ．最後に，基準文に付随する出典のうち現在アクセス可能で，かつ根拠がテキストで示されているものをダウンロードし，統計的手法を用いて段落単位に分割した．得られた結果を，表に示す．

記事数	1386531
文数	46558793
根拠が必要な文数	2678605

3 基準文候補と根拠テキスト候補の対応付け

この節では，基準文候補と出典に含まれる根拠テキスト候補を対応付けする方法の信頼性と作業コストについて検討する．出典を含む根拠テキスト候補c' を web コーパスから収集する方法を示す． 根拠テキスト候補c' の収集には 2 種類の手法を用いた．第 1 は，基準文候補・根拠テキスト候補に関連するキーワードを含む文を検索してヒットした文の中から， 既存根拠テキストcとの類似度が高いものを採用するという手法で収集する方法である．第 2 は，解答を含む文に対して，読解することで質問に答えられるか，すなわち解答可能性を大規模言語モデルによって分類し，解答可能と分類された文を採用するという手法である．ただし根拠テキスト候補c' は 1 つの基準文に対して複数存在するため，複数の根拠テキスト候補からなる根拠テキスト候補集合をC' ，この集合の要素である各根拠テキスト候補をc' と表記する．また根拠テキスト候補集合C' に対して，既知の根拠テキストからなる集合を以降既知根拠テキスト集合Cと表記する．既知根拠テキストCも 1 つの質問に対して複数存在することから，その集合を既知根拠テキスト集合C' ，各既知根拠テキストをcと表記する．

3-1 キーワード検索に基づく基準文候補と根拠テキスト候補の対応付け

この手法では，根拠テキスト候補c' を、web コーパスからキーワード検索を行った後、ヒットした文に対して既存根拠テキスト集合 C との類似度を計算することによって収集する．キーワードは， 基準文候補qや基準文候補qとの関連度が高い文章を見つけられる単語が好ましい． 既知根拠テキストcや基準文候補qと関連度が高い文章を見つけるために，キーワードの選定が重要となる．関連度が高い文章を検索で見つけるためのキーワードとして，対象となる基準文候補q，既知根拠テキストcの特徴をよく表す単語が望ましいと考えられる．本研究では， 基準文候補qとキチン根拠テキストcの特徴をよく表す単語として，基準文候補qと既知根拠テキストc中での出現頻度が高い名詞列から， 文章の絞り込みに適さないと考えられる代名詞や非自立名詞を除いたものを用いる．また，解答aが出現しない根拠文からは，読解による解答が不可能なため，解答aは必ずキーワードに含まれるようにした．具体的には以下の手順でキーワードを作成した．

① 既存データセットから，基準文候補，既知根拠テキストc，解答の三つ組を取り出す
② 基準文候補と全ての解答可能な既知根拠テキストに対して，MeCab を用いて形態素解析を行う．形態素解析を行なった結果から， 代名詞または非自立名詞以外の名詞列を抽出し， キーワード候補とする．
③ キーワード候補から解答とその部分文字列を取り除く．
④ キーワード候補のうち、質問文と根拠文の中で n 回以上出現しているものをキーワードとする．
⑤ コーパス内で多く出現するキーワードは、文章の絞り込みに寄与しないと考えられるためキーワードから除外する。

キーワードの生成後，解答aと一つ以上のキーワードを含む文を検索し，その前後 1 文と合わせて収集する．cc-100 コーパスは web 上のテキストを集めたコーパスであるため， 検索により得られた文をそのまま根拠テキスト候補c' として用いるのは，既知根拠テキストcと文量の面で差が生じるため，好ましくないためである．文の区切りは，改行記号に基づいて行う．また，文章にキーワードのいずれかが含まれることは、必ずしも基準文候補qとの関連度が高いことを意味しない．よって，既知根拠テキストcと得られた根拠テキスト候補をベクトル化した上で余弦類似度を計算し，余弦類似度が高いテキストを採用する．

3.2. 大規模言語モデルによる自動分類を用いた対応付け手法

この手法では， 根拠テキスト候補c' を， 解答を含む文に対して解答可能性， すなわち読解によって解答aを導き出せるかを自動分類することで収集する．まず解答 a を含む文とその前後 1 文を cc-100 コーパ

スから検索する．3.1 節と同様の理由から， 検索により得られた文をそのまま根拠テキスト候補 c' として用いることは，既知根拠テキストcと文量の面で差が生じる事から，好ましくないためである．文の区切りは，改行記号に基づいて行う．

解答 a が含まれる文全てが基準文候補qに関連した内容であるとは限らない．よって，次に大規模言語モデルを用いて， 検索で得られた文における解答可能性， すなわち読解によって解答aを導き出せるかを自動分類する．大規模言語モデルには基準文候補q，解答aと検索で見つかったテキストの組を入力し，解答可能であるかどうかの二値分類を行う．分類の結果，解答可能であると分類された文だけを，根拠テキスト候補 c' として採用する．

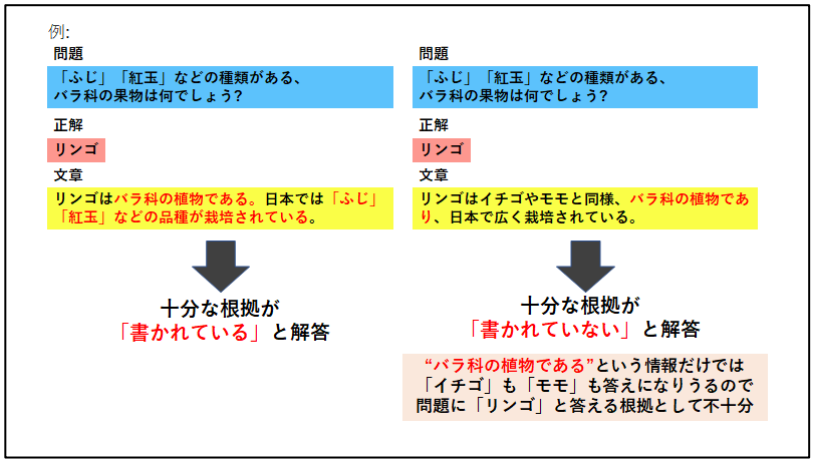
3.3. クラウドソーシングによる根拠の妥当性の評価

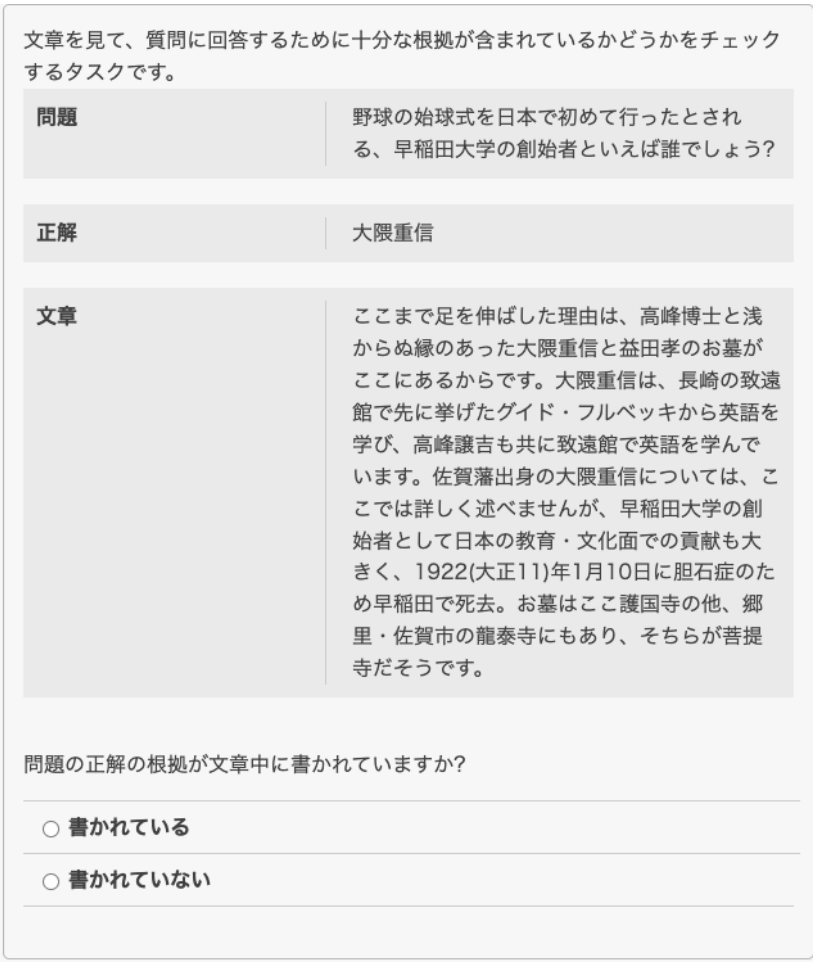
3.1 節および 3.2 節の対応付け手法によって得られた根拠テキスト候補集合C' は機械的に獲得されたものであるため，対応付けが妥当であるか否かは信頼できない可能性がある．よって，先行研究と同様にクラウドソーシングを用いて対応付けの妥当性をアノテーションする．

クラウドワーカーには，基準文候補・正解・根拠テキスト候補c' の組 $(q,\ a,\ d)$ を提示し，根拠テキスト候補c' に基準文候補qの正解aの十分な根拠が書かれているかどうかを質問し，「書かれている」「書かれていない」の 2 択で回答するよう教示した．

クラウドワーカーに対するタスクの教示画面を右上図に示す．クラウドワーカーが回答を行うタスク実施画面を右下図に示す．

ただし，不誠実なクラウドワーカーによる判定結果の混入を防止するために，注意設問を設ける．注意設問には RCQA に含まれる既知根拠テキストcとそれに対応する基準文候補qの中から，妥当性判定スコアs が 5 または 0 の事例をランダムに抽出して使用した．注意設問に使用する既知根拠テキストcには， 全く読まずに回答する不誠実なクラウドワーカーを排除するために，妥当性判定スコアs が 5 の場合は当該テキストの末尾に【これはチェック質問です。本当に根拠が書いてあるかどうかに関係なく「書かれていない」を選択してください】という文言を追加し，反対に妥当性判定スコア s が 0 の場合は【これはチェック質問です。本当に根拠が書いてあるかどうかに関係なく「書かれている」を選択してください】という文言を追加し





た.

データ作成の品質保証のため，1 回の作業に注意設問を 2 問導入し，注意設問に対する誤答がある作業結果は受け入れないようにした．1 作業あたりの設問数は 12 問（注意設問を含む）とした.

クラウドソーシングにより得られた根拠妥当性判定スコアs と既存の根拠妥当性判定スコアとの関係を調査するため，RCQA のデータに対して，上述の手順により再度アノテーションを行い，根拠は妥当性判定スコアs の変化を確認した．RCQA から根拠妥当性判定スコアs が 0 または 5 である基準文候補とそれに対応する根拠テキストと解答をそれぞれ 50 組ずつランダムに抽出し，クラウドソーシングによりアノテーションを付与した．結果を表に示す.

既知スコア	0	14	22	11	2	1	0
	1	3	8	9	16	9	5
	2	2	10	12	16	7	3
	3	1	3	11	9	11	15
	4	0	5	3	12	17	13
	5	0	0	3	5	13	29
		0	1	2	3	4	5
		クラウドソーシングによるスコア					

表の各要素は対応するスコアが得られた根拠テキスト候補の数である．ここで RCQA のスコアとクラウドソーシングで再度アノテーションした根拠妥当性判定スコアが共に 3 以上の根拠テキストを解答可能根拠テキスト，2 以下の根拠テキストを解答不能根拠テキストとすると仮定する．そうした合，RCQA の根拠妥当性判定スコアが 0 または 1 である根拠テキスト 100 文の内 67 文が解答不能根拠テキストであり，RCQA の根拠妥当性判定スコアが 4 または 5 である根拠テキスト 100 文のうち 89 文が解答可能根拠テキストである．それに対して根拠妥当性判定スコアが 2 または 3 である根拠テキスト 100 文のうち解答不能根拠テキストは 28 文，解答可能根拠テキストは 35 文であり，解答可能であるとも解答不能であるともいえない．つまり，たとえ注意設問を設けた場合であっても，クラウドソーシングの結果には大きな不確実性が存在し，そのまま利用することは危険である．そのため，1 つの作業結果を採用する場合も，少人数による結果ではなく，多人数の結果を採用し，かつ，大幅な過半数による一致を得た結果を採用するようにしなければならない.

よって本研究では，11 人のクラウドワーカーによる作業結果を採用する．11 人のクラウドワーカーによる作業結果について，根拠妥当性判定スコアs が 8 以上の根拠テキスト候補を解答可能根拠テキスト，根拠妥当性判定スコアs が 3 以下の根拠テキスト候補を解答不能根拠テキストとみなし，根拠妥当性判定スコアs が 4 以上 7 以下である根拠テキスト候補は使用しない.

4. おわりに

本研究では，推論の根拠を提示できる矛盾検出テストセットの構築に取り組んだ．テストセットを開発する手順は 4 段階からなる．第 1 に，Wikipedia テキストを対象として，出典が明示されている文を収集する．第 2 に，共同研究者が研究開発した機械翻訳技術に基づく言い換えモデルを用いて，基準文の言い換えを生成する．第 3 に，作業対象データを整理した上で，基準文と言い換え文の論理的関係をクラウドソーシングにより分類する．第 4 に，矛盾していると判定するために必要な外部知識が出典テキストに明記されているか判定する．本稿では，出典が明示されている文の収集方法と，出典が明示されている文と出典テキストの論理的関係をクラウドソーシングにより判定する場合の問題について報告した.

【参考文献】

[乾 2008] 乾健太郎．言語情報間の含意･矛盾関係の認識．言語, Vol.37, No.8, pp.30–37 (2008).

[Bowman 2015] Bowman,S.R., Angeli,G., Potts,C. and Manning,C.D.: A large annotated corpus for learning natural language inference, Proceedings of EMNLP2015, pp.632–642 (2015).

[Marelli 2014] Marelli,M., Menini,S., Baroni,M., Bentivogli,L., Bernardi,R. and Zamparelli,R.: A SICK Cure for the Evaluation of Compositional Distributional Semantic Models, Proceedings of LREC2014, pp.216–223 (2014).
[Geva 2019] Geva,M., Goldberg,Y. and Berant,J.: Are We Modeling the Task or the Annotator? An Investigation of Annotator Bias in Natural Language Understanding Datasets, Proceedings of EMNLP-IJCNLP2019, pp.1161–1166 (2019).
[Tsuchiya 2022] Masatoshi Tsuchiya and Yasutaka Yokoi. Developing a dataset of overridden information in Wikipedia. In Proceedings of LREC2022, pp.5601–5608 (2022).
[Tsuchiya 2018] Masatoshi Tsuchiya. Performance impact caused by hidden bias of training data for recognizing textual entailment. In Proceedings of LREC2018, pp.1506–1511 (2018).
[Hayakawa 2016] Daiki Hayakawa, Masatoshi Tsuchiya, and Hitoshi Isahara. Developing corpus of Japanese-English singular sentence textual entailment. In Proceedings of ICAICTA2016 (2016).

〈発 表 資 料〉

題　　名	掲載誌・学会名等	発表年月
Evaluating the Robustness of Question Answering Model against Context Variations	The 2023 International Conference on Advanced Informatics: Concepts, Theory and Application	2023 年 10 月