(401) ソフトウェア

機械読解による Wikipedia からの属性抽出における 表，箇条書きの変換の効果

Effects of Rewriting Table and Lists on Machine Reading Comprehension for Attribute Extraction from Wikipedia

石井 颯人† 菊井 玄一郎†

Hayato Ishii† Genichiro kikui†

†岡山県立大学大学院 情報系工学研究科 システム工学専攻

1 はじめに

文章を読み質問に答える機械読解のような自然言語処理の応用には，大規模な言語的な知識や実世界に関する知識が必要となる．このうち世界知識の作成には膨大なコストがかかり，またメンテナンスも問題となっている．この問題に対して有志が編集を行う Wikipedia が，情報資源として注目されている．しかし，Wikipedia は人が閲覧することを目的として編集されているため，そのままでは計算機で処理することはできない．

Wikipedia 構造化プロジェクト「森羅 2018」[1]は，Wikipedia に書かれている世界知識を計算機が利用可能な形に半自動的に構造化することを目的として，タスク参加者によりリソースを作成するプロジェクトである．タスクは，Wikipedia の各ページに対し，カテゴリ(例:人名, 企業名)ごとに定義された属性の値を抽出するものである．例えば，ある人物の情報を記述した人名のページであれば，その人物の「国籍」，「生年月日」などの属性値を Wikipedia のページ中から抽出する．今年度から始まった森羅 2019 ではアノテーションタスクへと変更されたことにより，属性値が Wikipedia のページ中の何行目の何文字目に存在するかという「オフセット情報」も必要となった．

Wikipedia からの属性抽出の問題は，Wikipedia の本文をドキュメント，属性名を質問, 属性値を回答と考えると, SQuAD(Stanford Question Answering Dataset)[2]のような，文章から質問の回答となる部分を選択するタイプの機械読解タスクとみなすことができる．これを解く手法として石井[3]は森羅 2018 のデータセットを SQuAD 形式に変換したものを入力として，深層学習を用いた DrQA[4] の DocumentReader をベースに日本語への対応及び複数回答への対応を行ったものを利用して回答を出力した．この手法は森羅 2018 において多くのカテゴリにおいて最高位の成績を収めたことから，森羅 2019 における評価の基準になると考えられる．

本稿ではこの手法の再現実験を行った結果のエラー分析を行い，誤った部分の特徴について述べ，入力データである Wikipedia の該当する部分の変換が機械読解における有効性について検討する．

2 既存手法とその問題点

2.1 DRQA による属性抽出法

DrQA のようなニューラルネットに基づく機械読解手法は読解対象が自然言語テキストであることを前提としている．ところが，Wikipedia において多くの属性情報が自然言語テキストではなくインフォボックスをはじめとする表形式などで記述されている．これらは自然言語を前提とする既存の機械読解ではうまく扱えない．この問題に対応するため，石井[3]の手法では，Wikipedia の記述内容のうち，インフォボックスなどの日本語の文ではないが属性抽出において有用と思われる部分を日本語の文に変換したあと，文書読解の手法を適用する．たとえば、下記 a)のような記述を b)のように変換する。

a)infobox

性別	男性

b)変換後文字列

性別は、 男性。

2.2 既存手法の問題点

我々は人名カテゴリについて, 石井[3]の手法の再現実験を行いそのエラー分析を行った．エラーが存在する Wikipedia ページごとにエラー項目を出力し，またエラー項目を

・A)正解データにあるが出力できなかった

発売日	規格	規格品番	アルバム	備考
1966年	LP	SJV-239	ある若者の歌	
1995年	CD		1.街角に唄おう 2.笑ってごらん 3.渚のしらべ 4.四角い太陽 5.手の中の落葉 6.この世の旅路	

図 3:Wikipedia 中の表

出演

テレビ

- バス通り裏（1963年、NHK）
- お姉さんといっしょ（1963年、NHK）
- 七人の刑事（1963年、TBS）
- ただいま11人（1964年、TBS）
- 特別機動捜査隊（1964年、NET）
- ハローICQ（1964年、東京12チャンネル）
- 虹の設計（1964年、NHK）
- かあちゃん結婚しろよ（1965年、TBS）

図 1:Wikipedia 中の箇条書き

目次 [非表示]

1 略歴
2 俳優活動など
3 受賞歴
4 出演
　4.1 テレビ
　4.2 映画
5 ディスコグラフィ
　5.1 シングル
　　5.1.1 プロモーション用非売品
　5.2 アルバム
　　5.2.1 非売品

図 2:変換すべきでない箇条書きの例

・B)正解データにないものを出力した

の 2 種類に分け，エラーが存在するページごとに集計した．その結果，「作品」や「所属組織」などの属性で A のエラーが多数見られた．エラー項目を確認すると Wikipedia 上で箇条書きや表といった自然言語で記述されずに存在する正解を抜き出せていないことが分かった．たとえば図 1 はある人物を表す Wikipedia ページの一部であり，この人物の「作品」属性にあたる出演作を示す部分が箇条書きで記述されている．人間は見出しの構成からこの箇条書きが出演作品であると解釈できるが，DrQA では「X の出演作品は Y」といった自然文による記述を想定しているため，作品かどうかという判定に失敗していると考えられる．

よってこれらを自然言語で説明する文章に置き換えることにより，表や箇条書きに正解が記述されやすい属性の再現率の改善が見込めると考えられる．

その際，図 2 の目次のような属性情報が書かれていない場合があるため，すべての箇条書きや表を変換するのは避ける必要がある．

3　提案手法

提案手法は前述のエラー分析に基づき入力データである Wikipedia の箇条書きや表で記述されている部分を下記の手順に従って変換したものを機械読解形式に変換し，そのデータセットを用いて機械読解により学習と予測を行う．

まず，Wikipedia 中の箇条書きや表のうち、自然言語に変換すべきものを選択し、次に選択されたものを自然言語文に変換する。以下、これらについて説明する．

3.1 変換すべき表や箇条書きの選択

2 章で述べたように Wikipedia 中には目次など，箇条書きであるにもかかわらず，属性値を表さない部分が存在する．これらを自然文に変換すると誤って属性値として抽出される可能性があるため、変換すべき箇条書きのみを選ぶ必要がある。

これを行うために我々は箇条書きの見出しに注目する。図 1 のように箇条書きで記述されている部分が何を示すものであるかは，「出演」と「テレビ」という 2 つの見出しで説明できる．すなわち，見出しは箇条書きで記述されている部分がそのページの何であるかを説明するものであると考えられる．

そこで，正解データが存在する訓練用の Wikipedia のページについて，見出しごとに Wikipedia の記事を区切り，その見出しの部分中の表，箇条書きされている部分中に正解が存在するかを確認し，存在すればその見出しを抽出しておく．見出しには階層構造が存在し，図 1 であれば見出し「出演」の下層に「テレビ」

表 1:実験結果

データセット	適合率	再現率	F 値
Infobox	50.1	13.8	21.7
Infobox+Table+Lists-Bottom	47.3	13.7	21.4
Infobox+Table+Lists-Top	50.1	14.4	22.4

表 2:変化があると予想した属性の比較

	Infobox			Infobox +Table+Lists-Bottom			Infobox+Table+Lists-Top		
属性	適合率	再現率	F 値	適合率	再現率	F 値	適合率	再現率	F 値
作品	60.0	5.3	9.7	55.8	5.0	9.2	61.2	5.4	9.9
所属組織	57.3	31.1	40.3	48.3	34.0	39.9	50.7	36.8	42.6

という見出しがある．階層構造のある複数の見出しに属する文章の場合，見出しの階層のどの部分が文章を説明するのにふさわしいか分からないため，階層構造を保ったまま見出しを抽出し上層，下層それぞれを選んだ場合を試すことにする．

最終的に抽出した見出しを箇条書きと表ごとにリストにした以下のような辞書を出力し，この辞書に存在するものと一致する見出しに属する箇条書きや表に対して以降の処理を行った．

{Lists:[[出演,テレビ],[….],….]
Table:[[…],[…],…]}

3.2 箇条書きと表の変換

箇条書きの変換

正解を持ちうる箇条書きの部分が存在する見出しを見つけた場合，箇条書きを

[見出し]は、[箇条書きの項目]。

という自然文に変換する．見出しに階層がある場合，最も階層の低い，または高い見出しを採用する．図 1 の場合，「出演」が最も高い階層の見出しであり，「テレビ」が最も低い階層の見出しである．変換は項目同士の間に存在する html タグや空白と置換する形で行い，変換前の属性値部分と変換後の属性値部分とオフセット位置が変わらないようにする．そのため見出しの文字数が短い場合は空白を用いて文字数を調整し，見出しの文字数が長い場合は変換しないものとした．置換後の文字列は例えば図 1 の入力に対して，最も階層の高い見出しを採用する場合，

出演は、バス通り裏(1963 年、NHK)。
出演は、お姉さんといっしょ(1963 年、NHK)。…

のように変換する．

表の変換

正解を持ちうる表が存在する見出しを見つけた場合，表を

[該当するセルと同列のヘッダ]は、[セルの単語]。

という自然文に変換する．同列に複数のヘッダが存在している場合，最も下のヘッダを採用する．また，全列で共通のヘッダが存在する場合は変換に利用しないものとした．例えば図 3 の入力に対して，

発売日は、1966 年。規格は、LP。
規格品番は、SJV-239。…

のように変換する．

4 実験

4.1 データセット

森羅 2019 においてシステム開発用に配布された人名カテゴリ 1028 ドキュメントのデータに対し，以下の 3 種類の処理を適用して 3 つのデータを作成した．

・Infobox:石井[3]同様インフォボックスの内容のみを自然文に変換したデータ

・Infobox+Table+Lists-Top:インフォボックスに加え，表と箇条書きを自然文に変換し，箇条書きの変換には**最上層**の見出しを採用したデータ

・Infobox+Table+Lists-Bottom:インフォボックスに加え，表と箇条書きを自然文に変換し，箇条書きの変換には**最下層**の見出しを採用したデータ

以上の 3 つのデータをそれぞれ SQuAD の入力形式型に変換し，これら各々を学習データ 873 ドキュメント，開発データ 52 ドキュメント，テストデータ 103 ドキュメントの 3 つに分割した．

4.2 評価指標

石井[3]同様，モデルの評価には，出力された回答と回答の適合率と再現率から求められる F 値[1]のマイクロ平均を用いた．

4.3 実験対象モデル

石井[3]の DrQA-Label を使用した．

4.4 パラメータ設定の詳細

石井[3]同様，単語ベクトルは，FastText を利用し事前に学習した 300 次元の分散表現を用いた．学習には日本語版 Wikipedia 本文を Mecab により単語分割したデータデータを用いた．

各モデルはエポック数 30，ミニバッチサイズは Infobox は 4，Infobox+Table+Lists の 2 種類は 8 で各モデルの学習を行った．テストデータに適用するモデルとして，全エポックのうち，最も開発データに対する F 値が高かったエポックのモデルを採用した．各閾値は石井[3]と同様に設定した．

5　実験結果と考察

各データのテストデータによる結果を表 1 に示す．Infobox と比べると，Infobox+Table+Lists-Bottom は Infobox に比べて適合率，再現率ともに下回る結果となり，Infobox+Table+Lists-Top は再現率がわずかに上昇した．

また変化があると予想した属性ごとの比較結果を表 2 に示す．「作品」，「所属組織」のどちらも Infobox+Table+Lists-Top の再現率が Infobox を上回る結果となった．このことから，表，箇条書きを自然文に変換することが機械読解において有効であると考えられる．また，Infobox+Table+Lists-Top が Infobox+Table+Lists-Bottom の再現率，適合率どちらも上回った．これは箇条書きを変換する際の主語について，最上層の見出しが最下層の見出しよりも一般的な語句であり，多くの記事で共通して記述されるためであると考えられる．

6　おわりに

本研究では，機械読解による Wikipedia からの属性抽出において，表や箇条書きの部分にエラーが多いことを述べ，これらを事前に自然文へ変換することによる改善を試みた．実験の結果，書き換えの際に多くの記事で共通する語句を主語として採用することにより抽出性能が改善できることが分かった．

今回はオフセットをそろえて変換を行うため採用する主語に制限を設けたが，箇条書きにおいて主語が結果に影響を及ぼすことも分かったため変換した後にオフセットをもとに戻すアルゴリズムを作成するなどしてオフセットに関する制限をなくし，より効果的な主語を選択することで結果の改善が期待される．

[1] 適合率と再現率の調和平均であり，式は $\frac{2Recall \cdot Precision}{Recall + Precision}$ で表される．ここで，Precision は適合率，Recall は再現率である．

謝辞

森羅データセットを整備・ご提供いただいた理研 AIP 森羅プロジェクトの関根聡委員長，小林暁雄委員をはじめとする森羅プロジェクト実行委員に感謝します．また，DrQA に基づく属性抽出プログラムを公開頂いた日本ユニシス株式会社の石井愛氏に感謝いたします．

参考文献

[1] 関根聡，小林暁雄，安藤まや．Wikipedia 構造化プロジェクト「森羅 2018」．言語処理学会第 25 回年次大会，2019

[2] P. Rajpurkar, J. Zhang, K. Lopyrev, and P . Liang . Squad:100,000 ＋ Questions for Machine Comprehension of Text. In EMNLP, pp.2383-2392

[3] 石井愛, 機械読解による Wikipedia からの情報抽出．言語処理学会第 25 回年次大会，2019

[4] D. Chen, A. Fisch, J. Weston, and A. Bordes . Reading Wikipedia to Answer Open-Domain Questions. In ACL, pp1870-1879, 2017