修 士 学 位 論 文

題名　自然言語文のトリプル化による
ナレッジグラフの構築

指導教員　相馬　隆郎　准教授

平成　３０年　1月 31 日提出

東京都立大学大学院

システムデザイン研究科　電子情報システム工学域

学習番号　20861651

氏名　古川　冬馬

論文要旨

近年,人工知能技術への関心が高まっており,これらが社会システムに埋め込まれるようになることが予測されている.人工知能技術の中でも自然言語処理の分野において大量のテキストデータから必要な情報を抽出する技術としてテキストマイニングがある.文章を形態素解析,構文解析,意味役割付与といった解析を利用して,文章を名詞,動詞,形容詞などの単語単位に分解し,それらの出現頻度や相関関係を分析することで有益な情報を抽出することが可能である.抽出されたデータを用いて様々な知識の繋がりをグラフ構造で表したものをナレッジグラフと呼び,質疑応答システムなどで活用されている.ナレッジグラフに関する応用研究の一環として人工知能学会セマンティックウェブとオントロジー研究会では,推理小説を人工知能で解析し,与えられた情報に基づいて事件の犯人を正しく突き止め,また理由を適切に求める事を目的としたナレッジグラフ推論チャレンジという競技会が開催されている.

推論チャレンジで用いられているナレッジグラフは小説の場面ごとに ID を割り振り,場面ごとの動作主や動作の対象などの相関関係が示されている.例えば「ロイロット博士はインドにいた頃ヘレンの母と結婚した」という文章に対して,インドにいた頃という場面で情報を検索すると[動作主：ロイロット博士,動作：結婚,動作対象：ヘレンの母]といったデータを抽出できる.これらのデータは RDF という表現モデルに基づいて記述されている.RDF は主語,述語,目的語のトリプル(三つ組み)で表現され,それぞれのデータには URI と呼ばれるグローバルな ID として異なる情報源との繋がりを作る.フリーのインターネット百科事典である Wikipedia でもボランティアの編集によって RDF の形式で情報が管理されており,Wikidata や DBpedia と呼ばれるオープンデータとして公開されている.

グラフ推論チャレンジにおいて提供される RDF は人手により作成されており,作業量の問題から小説全文に対する RDF 化は行われていない.そのため小説本文から自動で情報を抽出し RDF 化することが課題となっている.そこで本研究では推理小説の自然言語文を解析し,描かれる様々な状況を統一的な形式で計算機処理のための情報を抽出するシステムの構築を行い,ナレッジグラフ推論チャレンジにおける犯人の推理に活用できるナレッジグラフの自動構築を試みた.

既存の自然言語処理の解析器はオープンで利用できるものが複数存在し,NTT Communications の COTOHA や京都大学の KNP,岡山大学の ASA などが挙げられる.それぞれの解析器は,COTOHA は固有名詞補正や感情分析など多機能であり,KNP は格・照応解析,ASA は意味役割付与と解析の種類が異なっている.形態素解析は文章を最小の単語まで分割しそれらを辞書と照らし合わせ品詞の種類や活用形を割り出し処理で,構文解析は形態素解析で得られた単語間の関係性を明らかにする処理である.また照応解析は代名詞や指示詞といった指示対象を推定したり,省略された名詞句を補完する処理のことである.推理

小説などでは会話文が長く続いたり特殊な言い回しをすることが多いため構文解析に加え照応解析が必要であることが考えられる.本研究では構文解析と照応解析がどちらも可能である NTT Communications の自然言語処理 API プラットフォームである COTOHA を利用し,小説の全ての文章において構文解析を行い,主語,述語,目的語の三つ組みの形式で情報を抽出しナレッジグラフの構築を行った.また,扱うテーマが推理小説であるため,情報源をWho(誰が),Whom(誰に),Where(どこで),From（どこから）,To(どこへ),When(いつ),What(何を),Why(なぜ),How(どのように)に絞って主語,述語,目的語を抽出を行った.

また作成したシステムを用い,推理小説をトリプル化して構築したナレッジグラフが,構文解析と照応解析の精度などを踏まえナレッジグラフ推論チャレンジの推理タスクに利用できるかの評価を行った.

目次

第 1 章

研究背景

1.1 はじめに

自然言語とは,人間が普段書いたり話したりする際に用いる日本語や英語のことを指し,プログラミング言語に比べて曖昧性が高い.そのため 1990 年代頃の自然言語処理の分野では人の手で機械判読に適した辞書などの言語知識が作成されており実用化には膨大な時間と労力が必要であった.それが今では多くの携帯端末に音声対話エージェントが搭載されていたり,家庭に向けた Amazon 社のアレクサや Goggle 社の Google Home などのスマートスピーカーなども普及しており自然言語処理技術が身近なものとなってきている.それらの背景には機械学習の発展とその処理を行えるような高性能のコンピュータが開発されたことがある.自然言語の上流工程である形態素解析の精度が上がったこともありますます関心が高まってきている分野であると言える.しかし,下流工程にあたる述語を中心として,「【動作主】が【動作対象】に【動作】した.」という情報を抽出する述語項構造解析,文章中に現れる他の事柄を指す「それ」や「あれ」などの照応詞を読み取る照応解析,他にも固有表現抽出,語義曖昧性解消など様々な解析手法があるがそれらを応用に用いるには誤りが多く,課題とされている.上流工程に問題が残ったままだとエラー伝播を起こしてしまうため,本研究では自然言語処理に用いるデータの取得方法に関して人手で行われていた部分を自動化することを目的とする.

第 1 章では,ナレッジグラフの説明と用いられるデータと自然言語文の解析手法についての紹介,またそれらの解析を行うことができる解析器について紹介する.また,本研究の目的に関わる「ナレッジグラフ推論チャレンジ」という自然言語に関するコンペティションについて詳しく述べる.

第 2 章では推理小説における解析についての課題とコンペティションに向けた提案手法を述べる.

第 3 章では提案手法の実装方法と評価を行う.

1.2 オープンデータとデータベース

1.2.1 ナレッジグラフ

図 1.2.1 のように様々な知識のつながりをグラフ構造で表したもので知的システム開発の基盤として用いられる.

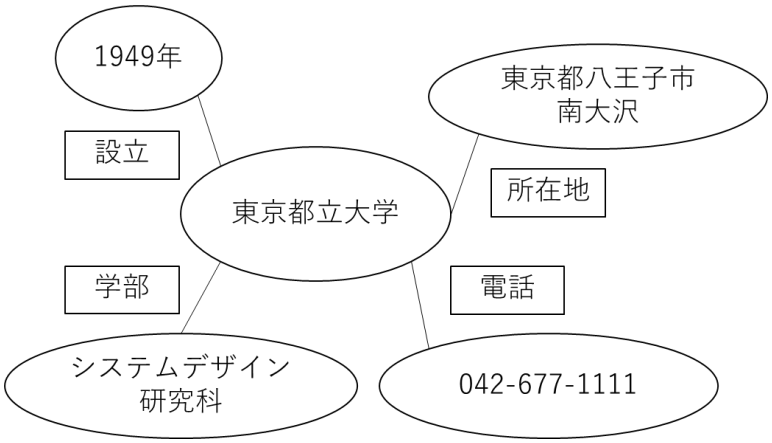


図 1.2.1 ナレッジグラフ例

Google での検索結果に図 1.2.2 のような検索に関するページとは別に分類や属性,関連事項が表示されるがこれは Google Knowledge Graph に基づいて表示されている.関連する情報先でも同様にナレッジパネルが存在し,ページにアクセスせずとも情報を取得することが可能である.

図 1.2.2 ナレッジグラフの利用例

1.2.2 メタデータ

出版物の電子書籍化が進み,大学や研究機関においても学術系機関リポジトリが整備されてきている.これはウェブ上においてもデジタルコンテンツを提供する基盤ができつつあるが,利用者が実際に求めるものを探す手段としては検索サイトが挙げられる.しかし,検索サイトを利用して得られる情報の中には信頼性に欠けるものも多々存在する.より信頼度の高い情報を得るためには,個々のデジタルアーカイブが持つデータを頼ったり各分野における専用の検索システムを利用するしかない.そこで,分野を超えて信頼性の高い検索を横断的に行うために一定の規則に則って記述されたデータが必要であり,これをメタデータと呼ぶ.また,メタデータを共有・交換目的で公開する場合にはリソースにグローバルな識別子である URI というものが与えられる.

1.2.3 オープンデータ

オープンデータとは,機械判読に適したデータ形式で二次利用が可能な利用ルールで公開されたデータである.政府 CIO[1]によって官民データ活用推進基本法において,国及び地方公共団体はオープンデータに取り組むことが義務付けられた.オープンデータへの取組により,国民参加・官民協働の推進を通じた諸課題の解決,経済活性化,行政の高度化・効率化等が期待されている.オープンデータの定義として,

① 営利目的,非営利目的を問わず二次利用可能なルールが適用されたもの
② 機械判読に適したもの
③ 無償で利用できるもの

と定義されている.

機械判読のしやすさでレベルが分けられており,表 1.1.1 に示す.

表 1.1.1 オープンデータの分類

段階	公開の状態	データ形式
1 段階	オープンライセンスで データを公開	PDF,JPG
2 段階	コンピュータで処理可能な データを公開	XLS,DOC
3 段階	オープンに利用できる フォーマットでデータを公開	XML,CSV
4 段階	Web 標準(RDF など)の フォーマットでデータを公開	RDF
5 段階	他へのリンクを入れた データ(LOD)を公開	Linked-RDF

1.2.4 RDF

RDF(Resource Description Framework)は,１９９９年にW3Cから勧告されたリソース一般の記述方法の枠組みである.2004 年には実装されるようになり RDF で記述された情報の現ミルな意味を含めた改定が勧告され,メタデータ記述方式の国際的な基準となった.RDF は,図 1.2.3 のようにリソースについての記述を主語―述語―目的語の三つ組み（トリプル）として表現するモデルであり,トリプルは,主語と述語のリソースを述語をラベルとした矢印で結んだ図で表される.

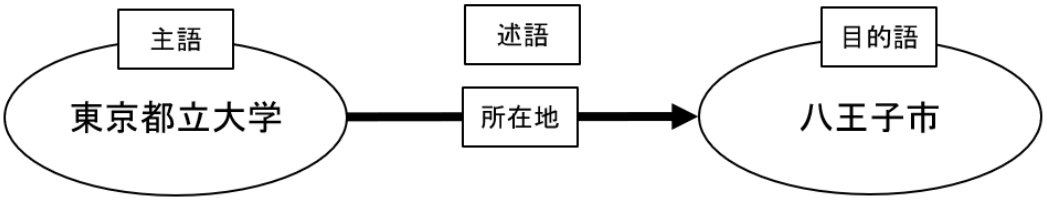


図 1.2.3 RDF データ例

RDF のトリプルの集合をグラフと呼び,グラフ中の述語をプロパティと示される.
RDF による記述は１つのデータベースだけではなくウェブ上で共有されるため 1.2.1 で説明した URI(Uniform Resource Identifier)が用いられる.URI の重要な特徴として,
①ウェブ全体で衝突する可能性がない識別子である.
②ユーザが任意の識別子を自由に作成できる分散型システムである.
の２つが挙げられる.同じ URI を用いて名前付けされたリソースは同一とみなせるため,異なる場所で別々に記述された RDF グラフであっても共通の URI を貸すことでデータを連動させることが可能である.図 1.2.4 のように例として「A はサッカー選手である」というトリプルと「B は野球選手である」というトリプルが別の場所で作られたとしても,「サッカー選手」と「野球選手」が同じ「スポーツ選手」という枠に属するためデータ間に関連性が少しでもあればデータの連結ができる.

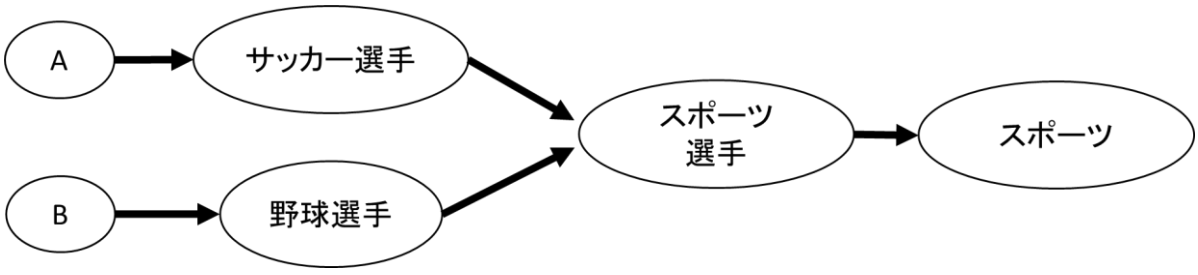


図 1.2.3 RDF データ連結

RDF の記述対象となるものはネットワーク経由でアクセスできるものに限らず名前付けが可能なものすべてである.

1.2.5 RDF スキーマ

RDF スキーマ(RDF Schema)は記述対象の分類を行うための「クラス」と述語に用いる「プロパティ」を定義する.後述する SPARQL による検索をするときにこれらを用いる.

1.2.6 LOD

LOD(Linked Open Data)は,ウェブ上に公開されていて誰でも利用が可能であるオープンデータのことを指す.ウェブの創始者であるティム・バーナーズ=リーによると Linked Data には次の 4 原則がある.

①事物の名前として URI を用いる.
②これらの名前を参照できるように HTTP URI を用いる.
③URI を参照したときに,RDF や SPARQL のような標準技術を用いて有用な情報を提供できるようにする.
④さらに多くの事物を発見できるように,他の URI へのリンクを含むこと.

これらを満たすことであらゆるデータを相互に繋ぐことが実現できる.

1.2.7 DBpedia

DBpedia は Wikipedia を自動的に RDF トリプルのデータに変換した大規模な LOD であり,カテゴリ情報,画像,地理座標,外部ウェブページへのリンク,といった構造化情報も記事に組み込まれている.これらの構造化情報が抽出され,問い合わせ可能な統一データセットの中に保存される.国立情報学研究所の加藤文彦[2]らによると 2017 年時点で日本語版のトリプル数は 1.1 億程度である.

DBpedia のデータモデルはリンクドデータに沿った形の RDF である.ウェブは文書表現としての HTML とその文章を一意に識別するための HTTP URI,文書を取得するための HTTP という要素で構成されている.リンクドデータはそれを拡張して色や単語といったような抽象的な概念もウェブ上で扱える.

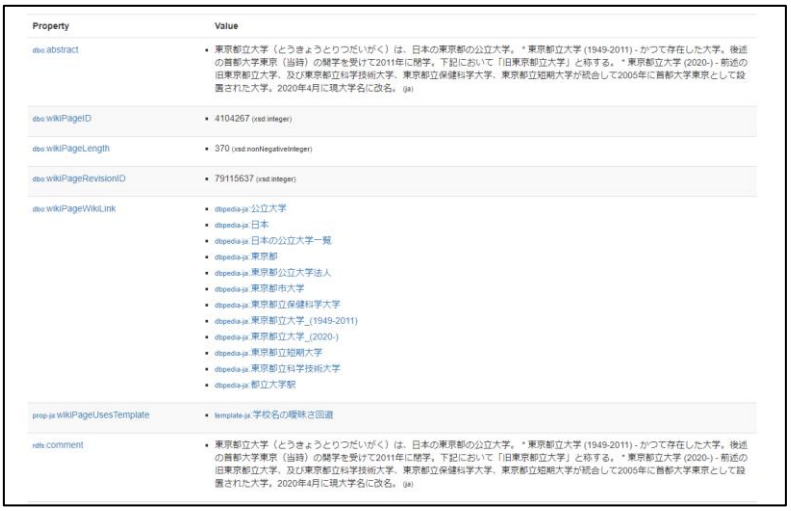


図 1.2.4 DBpedia

1.2.8 SPARQL

DBpedia の応用的な使い方で,データセットへのアクセス手段として公開 SPARQL エンドポイントを提供している.SPARQL は RDF 用のクエリ言語であり,クエリ言語とはユーザが DBpedia のようなデータベースに対して複雑な問い合わせが可能であり,公開 SPARQL エンドポイントは Web API であるのでデータ利用やアプリケーション開発を促進する.

検索例として図 1.2.3 のようなトリプルのデータがあるとして,「東京都立大学」を主語として述語が「所在地」のデータを検索すると「八王子市」という出力を得ることができる.目的語を「八王子市」,述語を「所在地」で検索をすると八王子市に存在する建造物がすべて出力される.

実際に SPARQL で検索する際には図 1.2.3 のような単語ではなく図 1.2.5 に示すような,すべてのことや物に割り振られている URI を参照する .図 4 は DBpedia を用いたクエリ検索である.

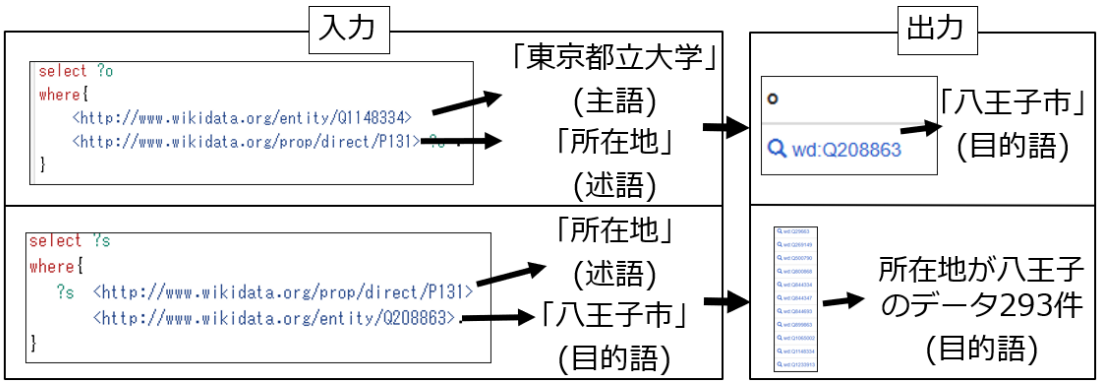


図 1.2.5 URI によって表記された RDF

1.2.9 グラフデータベース

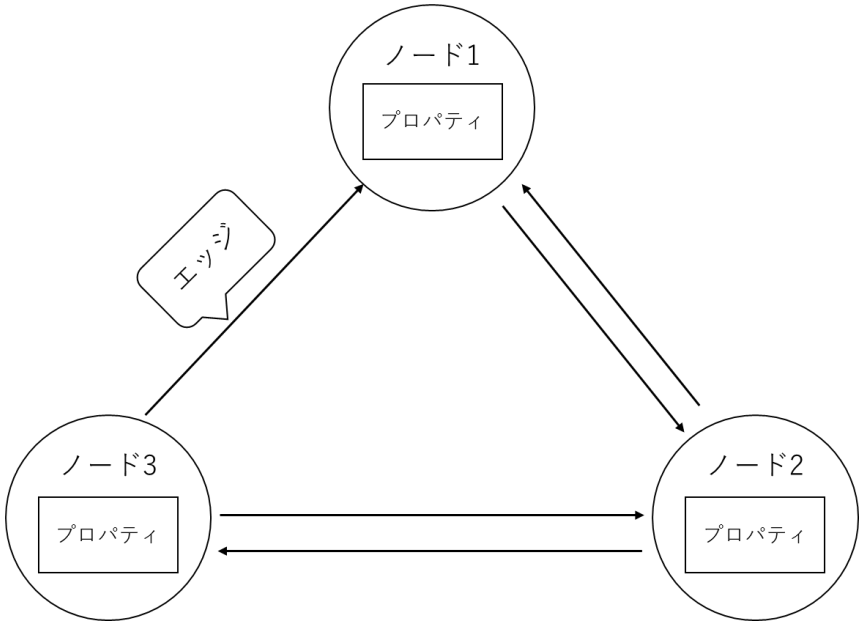


図 1.2.6 グラフデータベースの基本要素

グラフデータベースの基本要素は以下の 3 つで構成される.

ノード(node)：バーテックスとも呼ぶ.頂点で点やまるで表現されるエンティティ.ラベルをつけて種別を分類される.

エッジ(edge)：リレーションシップとも呼ぶ.辺でノード間の関係を表す.方向とタイプを有する.

プロパティ(property)： 属性とも呼ぶ.ノードとエッジにおける属性情報である.

これはトリプルのデータ構造と同様であり,三つ組みのデータをグラフデータベースに読み込ませることによって図 1.2.6 のような図式化や SPARQL でのデータ検索が可能である.

図 1.2.7 Fuseki

Fuseki[3]は RDF ファイルを永続化しクエリで検索可能にする RDF データベース(グラフデータベース)である.データの構造がネットワーク上で構成されており,データの格納・検索をすることが容易になる.

1.3 自然言語文の解析手法

自然言語処理は文章を①形態素解析②構文解析③意味解析④文脈解析の順で解析を行う.しかし,代名詞などについてはその対象を解析する照応解析が存在するように詳細な目的に合わせて多様な解析が存在する.

1.3.1 形態素解析

形態素解析とは図 1.3.1 に示すような自然言語文を言語上で意味を持つ最小単位に分け品詞の判別を行うこと.英語の文章ならば単語間に空白があるため分割が容易だが,日本語文だと単語の境目の考慮が必須でその分他言語より精度が求められる.形態素解析ツールの定義を以下に示す.

① 文章を最小単位に分割する分かち書きを行う
② 名詞や動詞など品詞を分類する品仕分けを行う
③ 単語の基本形を出す原型付与を行う

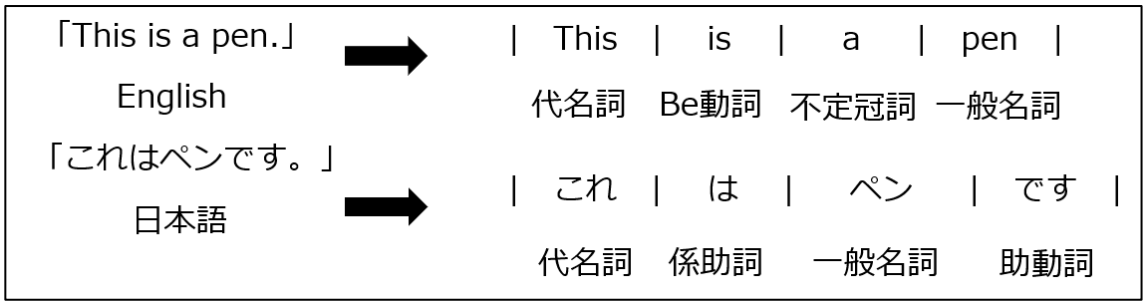


図 1.3.1 形態素解析例

1.3.2 構文解析

構文解析とは形態素解析で分割した単語同士の関連性を解析し文節間の係り受け構造を図式化することである.代表的な解析方法として,図 1.3.2 に示すように句構造規則に基づいて単語間の関係を決定する.

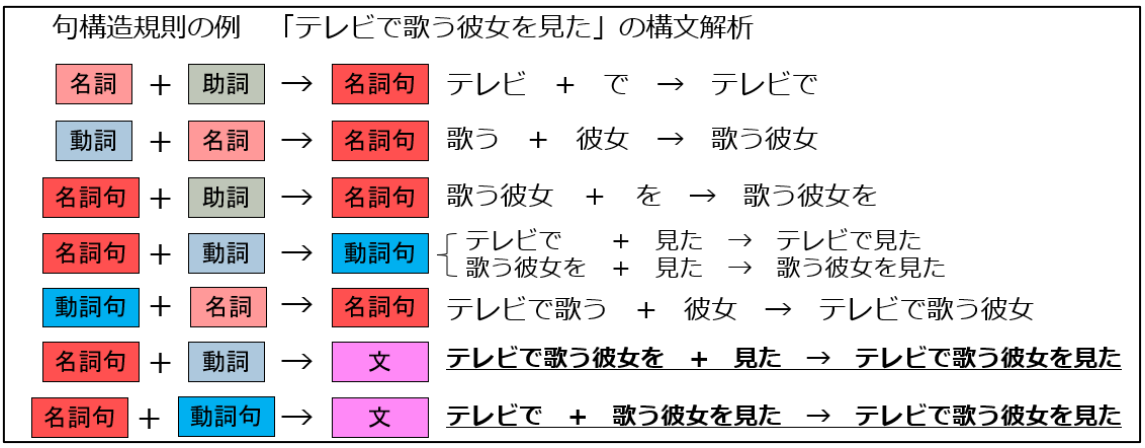


図 1.3.2 句構造規則に基づく解析

そして複数の文の形が候補として挙がるため図 1.3.3 の様に構文木と呼ばれる表現方法を用いて示す.

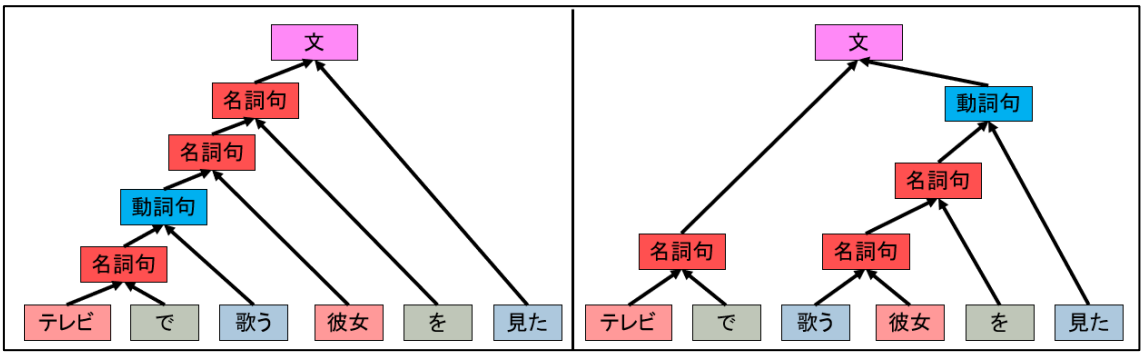


図 1.3.3 構文木

1.3.3 意味解析

構文解析によって出力された構文木から意味内容が正しいものを選ぶ解析.格解析,述語項構造解析,多義性解消など文の意味を明らかにする処理すべてが意味解析とされる.

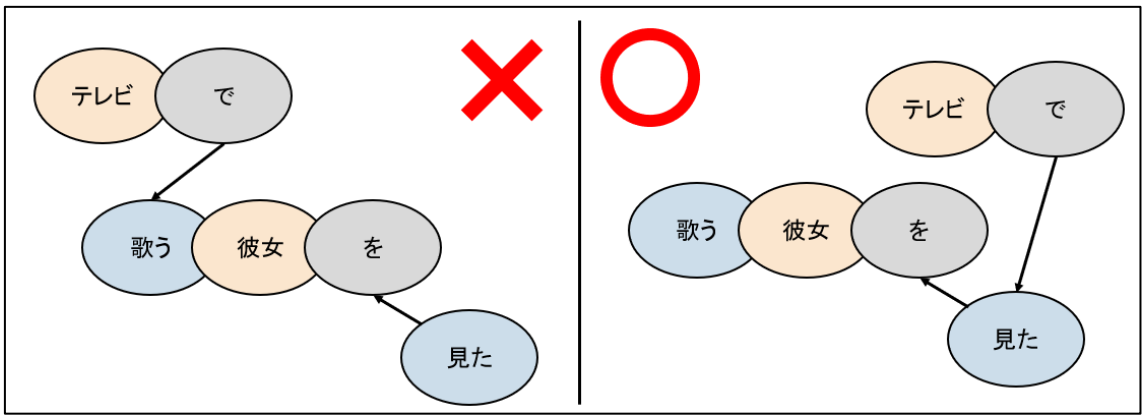


図 1.3.4 意味解析

1.3.3 文脈解析

図 1.3.5 のように複数の文にまたがる構文木作成と意味解析を行う.照応詞などの参照問題も含まれ,物語の理解などには文脈解析が不可欠である.

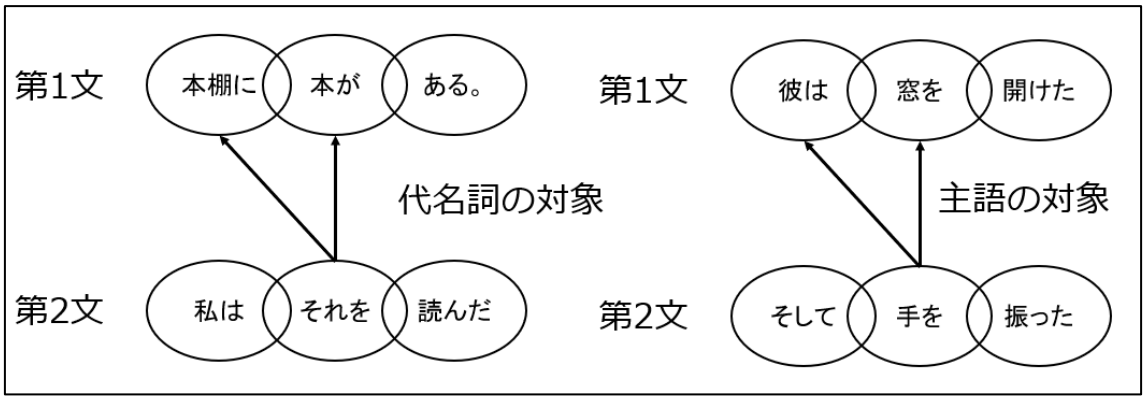


図 1.3.5　文脈解析

1.3.5 格解析

格解析は各用言に対して付与されるもので,構文解析による係り受け解析では分からない,用言と名詞との関係を解析する.一般的なガ・ヲ・ニ・デ格を解析することで原文では省略されている語を補完した「彼女が歌う」と判別され,機械判読に適したものになる.

文節

ID:0, 見出し:テレビで, 係り受けタイプ:D, 親文節 ID:3, 素性:<文頭><デ><助詞><体言><係:デ格><区切:0-0><格要素><連用要素><正規化代表表記:テレビ/てれび><主辞代表表記:テレビ/てれび>

ID:1, 見出し:歌う, 係り受けタイプ:D, 親文節 ID:2, 素性:<連体修飾><用言:動><係:連格><レベル:B><区切:0-5><ID:（動詞連体）><連体節><正規化代表表記:歌う/うたう><主辞代表表記:歌う/うたう>

ID:2, 見出し:彼女を, 係り受けタイプ:D, 親文節 ID:3, 素性:<SM-主体><SM-人><ヲ><助詞><体言><係:ヲ格><区切:0-0><格要素><連用要素><正規化代表表記:彼女/かのじょ><主辞代表表記:彼女/かのじょ>

ID:3, 見出し:見た, 係り受けタイプ:D, 親文節 ID:-1, 素性:<文末><補文ト><時制-過去><用言:動><レベル:C><区切:5-5><ID:（文末）><提題受:30><主節><正規化代表表記:見る/みる><主辞代表表記:見る/みる>

基本句

ID:0, 見出し:テレビで, 係り受けタイプ:D, 親基本句 ID:3, 素性:<文頭><デ><助詞><体言><係:デ格><区切:0-0><格要素><連用要素><正規化代表表記:テレビ/てれび><解析格:デ>

ID:1, 見出し:歌う, 係り受けタイプ:D, 親基本句 ID:2, 素性:<連体修飾><用言:動><係:連格><レベル:B><区切:0-5><ID:（動詞連体）><連体節><正規化代表表記:歌う/うたう><用言代表表記:歌う/うたう><格関係 2:ガ:彼女><格解析結果:歌う/うたう:動 18:ガ/N/彼女/2/0/1;ニ/U/-/-/-/-;デ/U/-/-/-/-;カラ/U/-/-/-/-;ヨリ/U/-/-/-/-;マデ/U/-/-/-/-;時間/U/-/-/-/-;ノ/U/-/-/-/-;修飾/U/-/-/-/-;ガ 2/U/-/-/-/-;外の関係/U/-/-/-/->

図 1.3.5 格解析例

1.3.6 照応解析

ある言語表現が,後に現れる言語表現と同じ内容や対象を指すとき,これらの表現は照応関係にあるといい,前者を先行詞,後者を照応表現という.

照応表現には以下の 5 つがある,

①繰り返し,②代名詞,③名詞形態指示詞,④連体詞形態指示詞,⑤ゼロ代名詞

それぞれの例を図 1.3.6 に示す

①おじいさんとおばあさんがいました.

おじいさんは山へ芝刈りに…

②太郎君は研究室の友達だ.彼は…

③パソコンを買いました,それは先月…

④文章解析システムが発明されました.

このシステムは…

⑤太郎は転んだ.

そして(∅は)泣いた.

図 1.3.6 照応表現例

1.4 解析器

1.3 項で説明した解析を行う解析器を紹介する.代表的な解析器はウェブ上でデモが可能であり,また開発者のためのリファレンスが用意されている.

1.4.1 COTOHA API

NTT Communications[4]の自然言語解析 AI エンジンでありさまざまな自然言語処理 API を提供しており 1.3 項で説明した解析を全て行うことが可能.各 API について開発者向けのリファレンスが記載されており,Developers として登録を行えば誰でも利用が可能である.

図 1.4.1 COTOHA API

COTOHA API で利用できる機能は図 1.4.2 である.

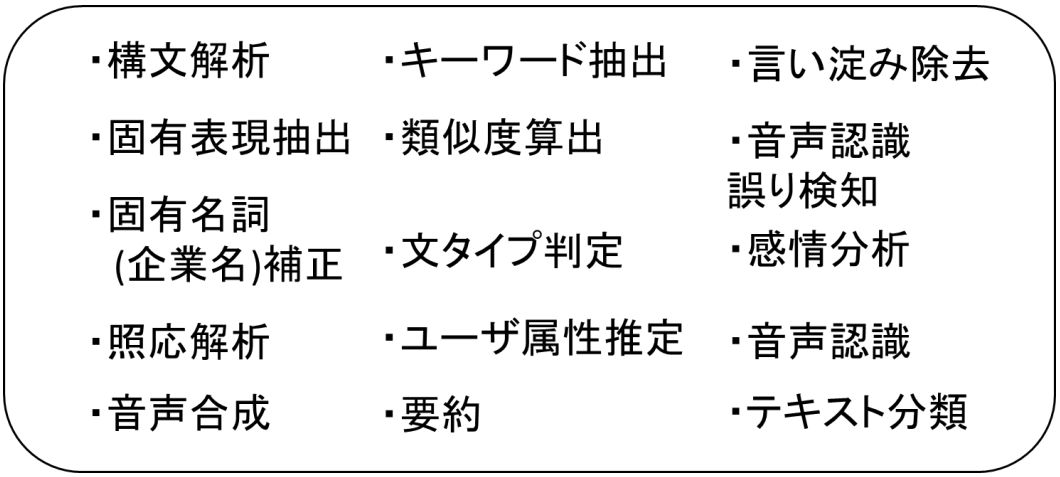


図 1.4.2 COTOHA API 全機能

図 1.3.6 の照応表現を全て試してみた結果を図 1.4.3~図 1.4.8 に示す.

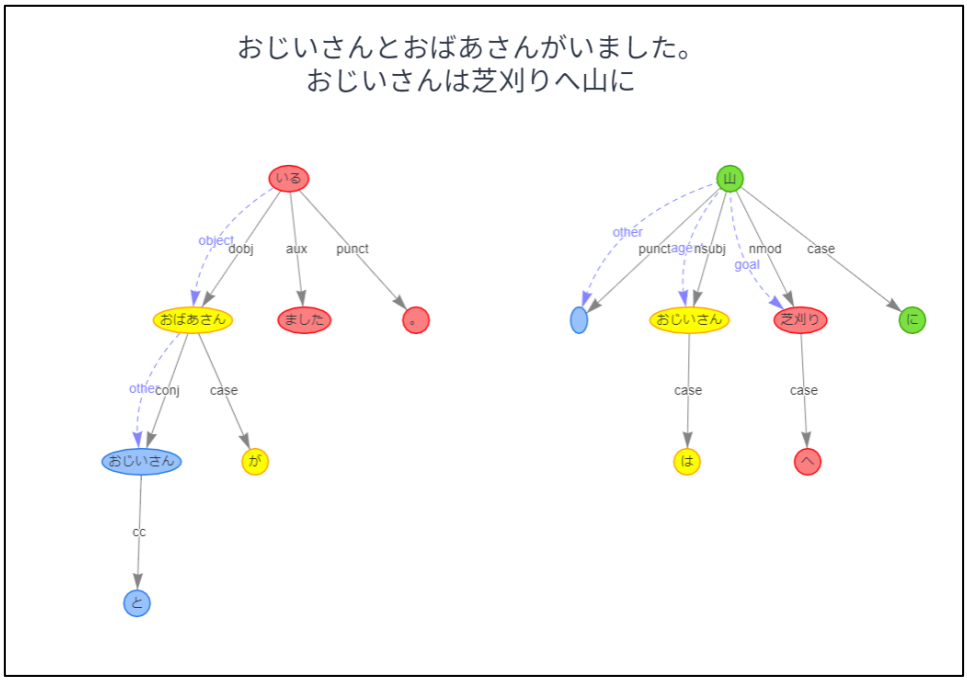


図 1.4.3 繰り返し表現

繰り返し表現には対応していないことが図 1.4.3 からわかる.
図 1.4.4 のように代名詞解析は確認できた.

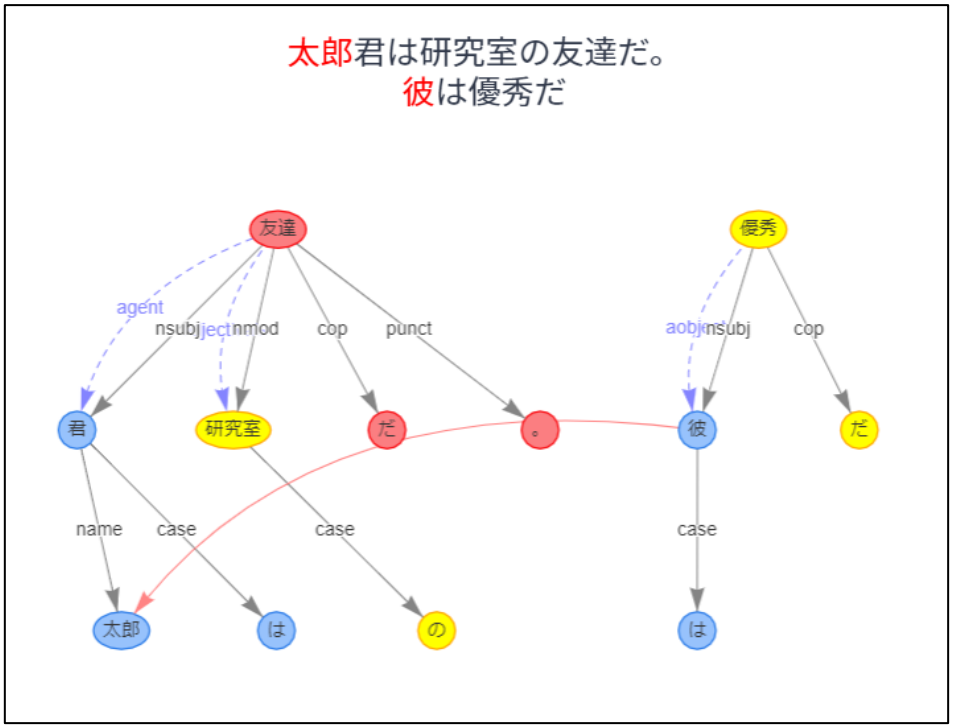


図 1.4.4 代名詞解析結果

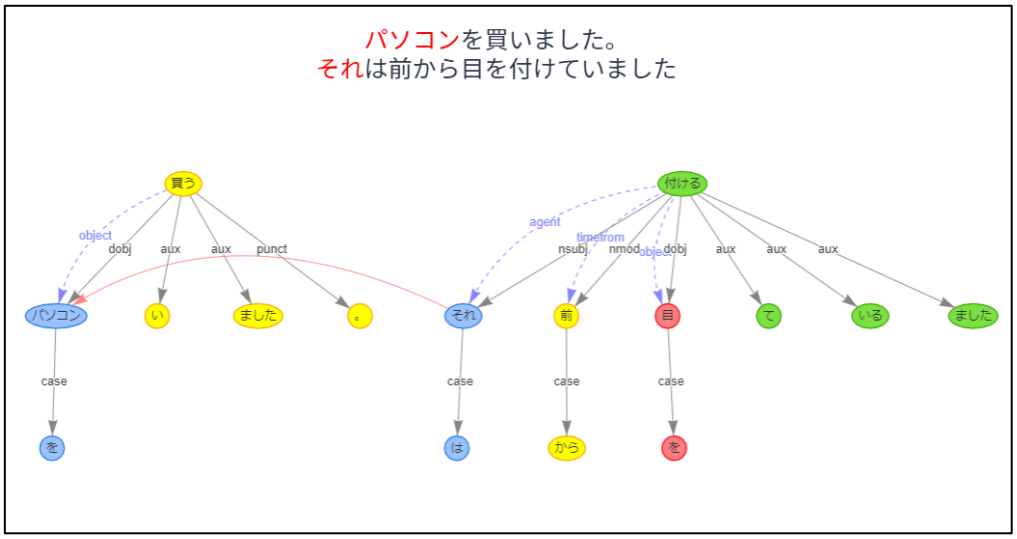


図 1.4.5 名詞形態指示詞解析結果

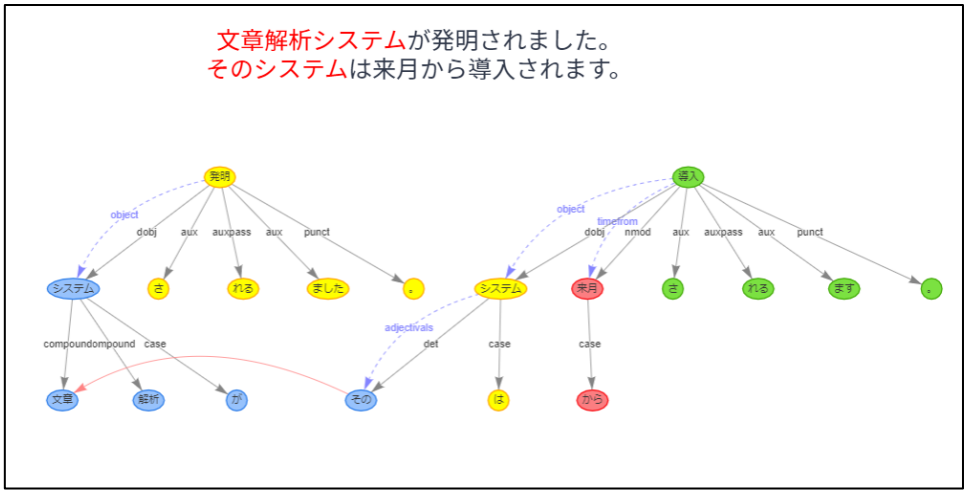


図 1.4.6 連体詞形態指示詞解析結果

名詞形態指示詞と名詞形態指示詞の解析は図 1.4.5 図 1.4.6 から確認できたがゼロ照応解析には対応していないことが図 1.4.7 から分かる.

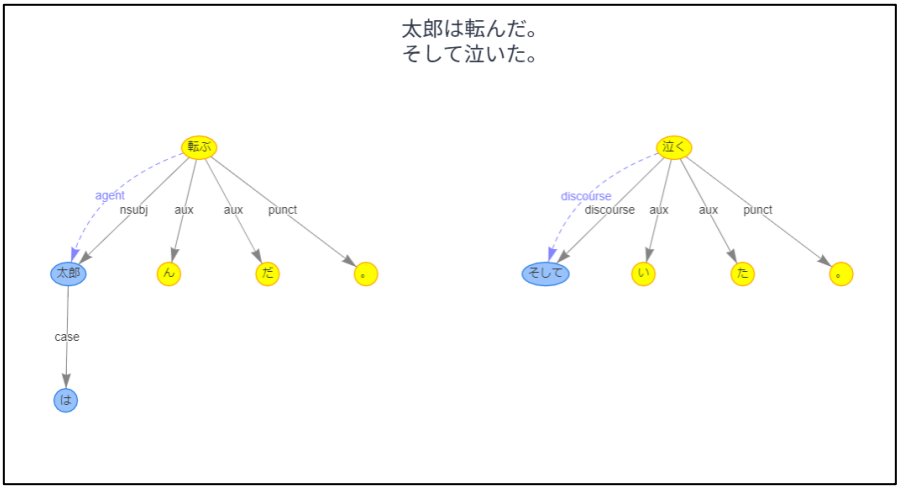


図 1.4.7 ゼロ照応解析結果

1.4.2 KNP/JUMAN

京都大学の黒橋・褚・村脇研究室が開発した[5]日本語文の構文・格・照応解析を行うシステムである.同研究室により開発された形態素解析システムの JUMAN の結果を入力としており,どちらも Python のバインディングが公開されており誰でも利用が可能である.

格解析によって格関係や照応関係を web から自動構築した大規模格フレームに基づく確率的モデルによって決定することが可能である.JUMAN は最小コスト法に基づいて形態素解析を行う.最小コスト法は文章の全分割パターンを表現したラティス構造を作り,その中から最小コストになる経路を選択する.

KNP の格解析・照応解析の出力を図 1.4.8 に示す.

```
# S-ID:1 KNP:4.11-CF1.1 DATE:2021/12/20 SCORE:-1045.07639
* 1D <体言><係:未格>
+ 1D <体言><係:未格><NE:PERSON:太郎><EID:0>
太郎 たろう 太郎 名詞 6 人名 5 * 0 * 0 "人名:日本:名:45:0.00106 疑似代表表記 代表表記:太郎/たろう" <係:未格><NE:PERSON:S>
は は は 助詞 9 副助詞 2 * 0 * 0 NIL
* 3P <用言:動><係:文末>
+ 3P <用言:動><係:文末><EID:1><述語項構造:転ぶ/ころぶ:動 7:ガ/N/太郎/0>
転んだ ころんだ 転ぶ 動詞 2 * 0 子音動詞バ行 8 タ形 10 "代表表記:転ぶ/ころぶ" <係:文末>
. . . 特殊 1 句点 1 * 0 * 0 NIL
* 3D <係:連用>
+ 3D <係:連用><EID:2>
そして そして そして 接続詞 10 * 0 * 0 * 0 "代表表記:そして/そして" <係:連用>
* -1D <用言:動><係:文末>
+ -1D <用言:動><係:文末><EID:3><述語項構造:泣く/なく:動 1:修飾/C/そして/2;ガ/O/太郎/0>
泣いた ないた 泣く 動詞 2 * 0 子音動詞カ行 2 タ形 10 "代表表記:泣く/なく 反義:動詞:笑う/わらう"
. . . 特殊 1 句点 1 * 0 * 0 NIL
EOS
```

図 1.4.8 KNP ゼロ照応解析

<EID>というデータが照応解析のための ID で全ての基本句に付与される.下線がひかれた部分が各格要素を示しており書式は, 格/フラグ/ENTITY 表記/EID という並びになっている.「泣いた」という語がガ格であることから主語を最後の要素で示しており,0 と表記されているが EID0 の基本句は「太郎」であるため省略されている主語を検出できていることが分かる.

1.4.3 ASA

岡山大学の竹内研究室が開発した[6]意味役割付与を行うシステムである.入力分に対して述語項構造解析を行い,その後述語の語義を同定してかかわり関係にある項の意味役割付与を行う.句の分解と形態素解析には奈良先端科学技術大学院大学の CaboCha を利用している.

JUMAN で利用する 3 万語程度の基本辞書についてさまざまな語彙情報・意味情報を人手で正確に整備し,その範囲を超えるものに関しては Wikipedia や web コーパスからの自動語彙獲得を行う.

ID	0
Surface	彼は
Link	2
CType	elem
Main	彼
Part	は
Category	["人"]
Semrole	対象
Similar	3.8073549270629883
Tense	PRESENT
Arg	["Arg1"]
Frames	["2-verb"]
Morphemes	0 彼 カレ 彼 名詞,代名詞,一般 O 1 は ハ は 助詞,係助詞 O

図 1.4.9 ASA 出力

図 1.4.8 の KNP の出力では単語を意味カテゴリとして大きく分類しており図 1.4.9 の Category の行にあたる.ASA はそれに加えて Arg と Frames という要素で述語項構造解析による意味役割付与を行っている.

1.5 ナレッジグラフ推論チャレンジ

AI技術を安全・安心に社会の中で活用していくためには,システムが正しく動作しているかの検証や品質保証のため,システムが判断に至った理由を背記名できる(解釈可能性を有する)AI 技術が必須である.その様な背景のもとに人工知能学会セマンティックウェブとオントロジー研究会では,人工知能技術による推論(推定)に関して,認識の旧友と必要な技術の開発・促進を図ることを目的として開催されるコンテストである[7].

図 1.5.1 ナレッジグラフ推論チャレンジ

1.5.1 コンテスト内容

シャーロック・ホームズ短編集におけるホームズが解決した事件を題材として行う.
対象とする小説のナレッジグラフを公開しており,それを用いて事件の真相を推論（推理）し,真相と判断した理由の説明と共に示す.タスクの正解としては,ホームズと同じ結論に辿り着けること.（ホームズが解決した事件の真相を説明する.）

1.5.2 応募部門

①本部門：対象小説 1 つ以上のタスクを解くシステムを開発
②ツール部門：いずれかのタスクを部分的に解くツールを開発
③アイデア部門：①，②の実現方法のアイデア（実装無しで OK）

1.6 自然言語処理分野の課題

自然言語処理の分野の大きな課題を 2 つ説明すると 1 つ目に,冒頭で述べた音声対話エージェントなどが活用されているが,それらが実用的であるのは家電の操作,Web での検索,スマートフォンなどのリモート操作といったあらかじめ機能の範囲が想定されているからである.1.5 項で説明したナレッジグラフ推論チャレンジではシャーロック・ホームズの推理小説の内容を用いるが,そのための 1.2.2 項で説明したような三つ組みのデータは人手で作られており,また 1.3 項で自然言語処理の解析手法を紹介したが,これについても最終的な出力結果の正誤を判定するのが人手であるため自然言語文の文章量を考えると,応用的なシステムに用いることは困難である.2 つ目に自然言語文の解析をする上で誤りが伝播してしまう事が挙げられる.一般的に下流タスクとなる構文解析などは形態素解析などの上流解析が正確に実行されたことを前提として解析を行う.意味解析は構文解析からの出力を入力とするため始めの形態素解析の結果が誤りを含んでいる場合誤りの伝播が起こってしまう.現行の形態素解析器の精度は 97%程度であるが,3%の誤りが伝播することで大きな影響が出てしまう.

本研究ではその人手で RDF のデータが構築されているを解消し,1.5 項のナレッジグラフ推論チャレンジのツール部門に提出できるような自然言語文からトリプルのデータを抽出できるようなシステムの開発を目標とする.

1.7 関連研究

1.7.1 Wikipedia 記事からの中間 RDF グラフと DBpedia トリプルの抽出

電気通信大学大学院の末木,兼岩[8]らによる Wikipedia 記事本文のコンテンツを構造化データとして活用するための,自然言語文の係り受け関係と述語項構造に基づく中間 RDF グラフ及びその生成手法が開発されている.Wikipedia の記事本文を述語項構造解析と格解析による係り受け関係を解析し中間データを抽出する.また,DBpedia 中のリソースと語を結びつけることで文書中のコンテンツを意味的に検索できるようにする.語間関係から役割関係,PAS 関係を定義し SPARQL クエリを用いてその 2 つの関係の候補トリプルを抽出する.

1.7.2 日本語 Wikipedia インフォボックスからのプロパティ自動抽出

慶応義塾大学の玉川,桜井,手島,森田,和泉,山口[9]らによる日本語版 Wikipedia をリソースとして Infobox からプロパティを抽出して各プロパティの定義域と値域の評価を行った.Wikipedia はオントロジー学習において,コストレスで大規模なオントロジーを構築するために有用なリソースであることを示した.

1.8 ナレッジグラフ推論チャレンジ過去作品

1.8.1 登場人物一覧の取得

2018 年度の本部門に投稿されたもので,富士通研究所の岡嶋,鵜飼,村上,神戸常盤大学の高松,神戸市立西神戸医療センターの岸田ら[10]による犯人予測の手法では,ホームズ短編小説に出現する登場人物について,対応する単語ベクトルを特徴に「犯人」「被害者」「その他」の分類モデルを学習し,予測を行う.その中の本文からの登場人物一覧の作成手法について紹介する.

形態素解析を利用して図 1.8.1 のように単語区切りの文を作成する.

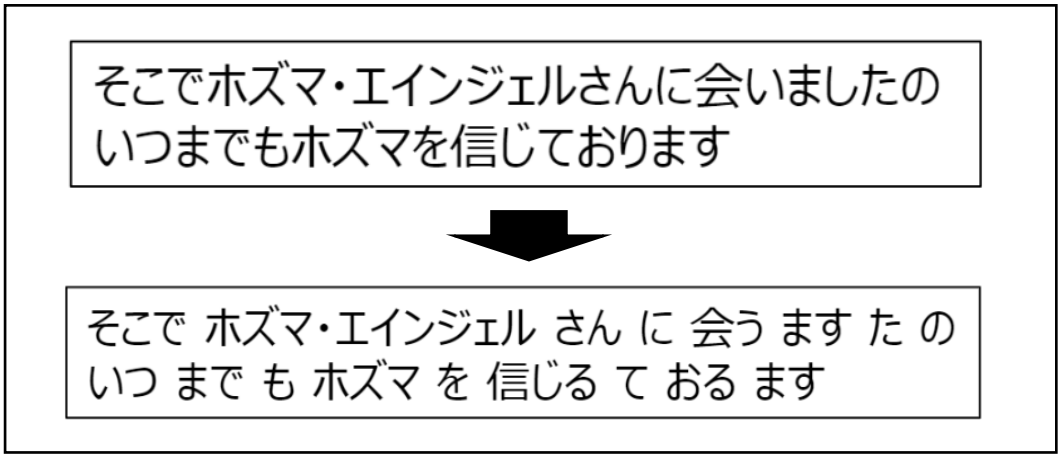


図 1.8.1 形態素解析

次に固有表現抽出を適用し人物を抽出する.固有表現抽出によって図 1.8.2 のように人物名が<PERSON></PERSON>で区切られる.

そこで <PERSON>ホズマ・エインジェル</PERSON> さん に 会う ます た の
いつ まで も <PERSON>ホズマ</PERSON> を 信じる て おる ます

図 1.8.2 固有表現抽出

カテゴリが PERSON に分類された語に対して表記ゆれを解消する.図 1.8.2 の表記ゆれを解消すると図 1.8.3 のように「ホズマ」を「ホズマ・エインジェル」に変換できる.

そこで <PERSON>ホズマ・エインジェル</PERSON> さん に 会う ます た の
いつ まで も <PERSON>ホズマ・エインジェル</PERSON> を 信じる て おる ます

図 1.8.3 表記ゆれの解消

上記の解析で 14619 文を生成し,人物の一覧を作成したところ,113 名の人物を抽出した.

1.8.2 オントロジーを用いた犯人の推論

2018 年度の本部門に投稿されたもので,富士通研究所の松下,金子,吉川,小林,小柳,鵜飼,西野,織田ら[11]による事件での動機を持つ人物について,殺人同期のオントロジを作成,殺人動機が生じる状況の記述し,そこで推論された人物が殺害したい人物を持つかを推論する.ナレッジグラフを用いたルールを設定し推論を行う手法の紹介をする.

殺人の動機を持つ人を

①何らかの事象が起きると　お金を得るもしくはお金を失う　可能性がある人.

②明確な誰かから恐怖や肉体的もしくは精神的苦痛を与えられている人.

③親密な人がなくなっていて,なくなった原因となる人物の予測がついている.

であると考え,①を「金欲」,②を「自己防衛」,③を「怨恨」と定義し,親密な関係を持つ人物を判断するために家族・友人関係のナレッジグラフを追加作成し,他にも本文中で必要であると考えられるがナレッジグラフ推論チャレンジで公開されているナレッジグラフには記述されていない部分を追加作成した.

①②③の条件と追加したナレッジグラフを用いて検索したところ図 1.8.4 のような出力を得た.

ロイロット は “金欲” のために ジュリア と ヘレン を殺す可能性がある
村人 は “自己防衛” のために ロイロット を殺す可能性がある
ヘレン は “自己防衛” のために ロイロット を殺す可能性がある

図 1.8.4 動機を持つ人物の推論結果

1.8.3 グラフニューラルネットワークによる犯人推定

2020 年度のツール部門に投稿されたもので,東京都市大学の宍田ら[12]が作成したナレッジグラフ推論チャレンジが作成したナレッジグラフに操作方法に関する情報が欠けており,それを未知エンティティとしてニューラルネットワークで解を探して補完するシステムである.

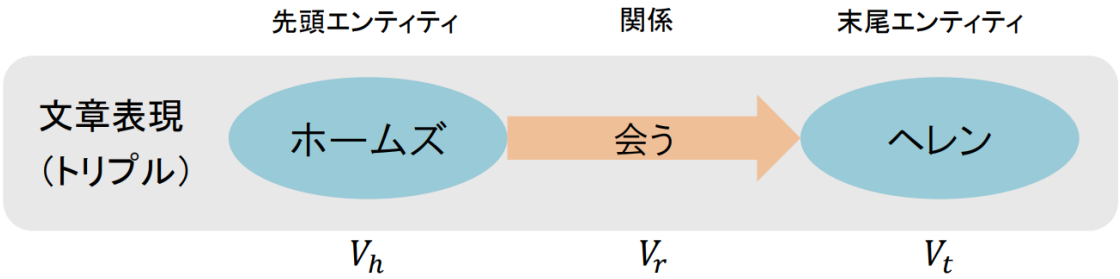


図 1.8.5 エンティティの定義

通常の知識グラフ補完では,説明性のための“ルール”や“常識”の追加ができないため,図 1.8.5 のように,トリプルの先頭エンティティと関係,末尾エンティティの表現ベクトルがそれぞれ $\mathbf{V}_h$, $\mathbf{V}_r$, $\mathbf{V}_t$ の時,

$\mathbf{V}_h + \mathbf{V}_r = \mathbf{V}_t$

が成り立つようにベクトルの学習を行う.その様にしてナレッジグラフを訓練し,図 1.8.6 のようにナレッジグラフ上にない犯人に関するエンティティを考える.

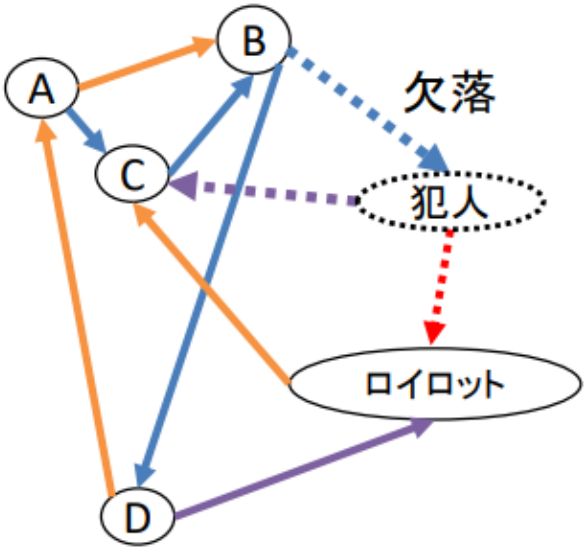


図 1.8.6 犯人に関する未知エンティティ

ここで 2018 年に野村総合研究所の濱口ら[13]がナレッジグラフ推論チャレンジの本部門に投稿したシステムのグラフニューラルネットワークを利用する.

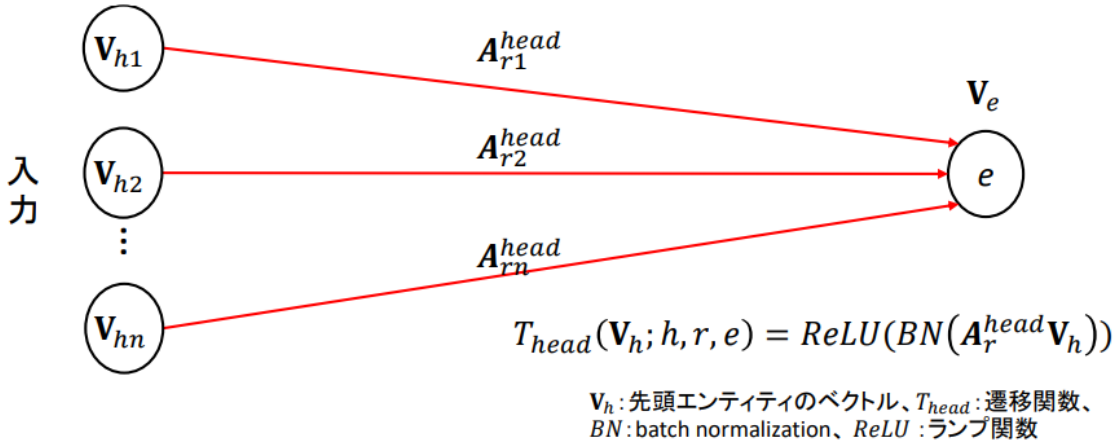


図 1.8.7 グラフニューラルネットワーク

グラフニューラルネットワークでは,あるエンティティ(e)の近傍エンティティのベクトルを入力,そしてそれをつなぐ関係を重みとする.

伝播モデルを説明する.遷移関数としてエンティティeとその近傍h,tとの関係rを反映するような近傍のベクトル$\mathbf{V}_h$,$\mathbf{V}_t$を変換する.

$$T_{head}\ \mathbf{V}_h\ ; h, r, e = ReLU(BN\ (\boldsymbol{A}_r^{head}\mathbf{V}_h\ ))$$
$$T_{tail}\ \mathbf{V_t}; e, r,t = ReLU(BN\ (\boldsymbol{A}_r^{tail}\mathbf{V}_t))$$

プーリング関数として先頭・末尾の近傍集合から共通の側面を抜き出す.$N_h(e)$と $N_h(e)$をそれぞれ先頭近傍,末尾近傍,P を要素ごとの平均とすると,

$$S_{head}\ (e) = \{\ T_{head}\ \mathbf{V}_h\ ; h, r, e\ |(h, r, e) \in N_h(e)\}$$
$$S_{tail}\ (e) = \{T_{tail}\ \mathbf{V}_t\ ; e, r,t\ |(e, r,t) \in N_t(e)\}$$
$$\mathbf{V}_e = P(S_{head}(e) \cup S_{tail}(e))$$

出力モデルを説明する.伝播モデルで計算されたベクトルを用いてトリプルの正誤と最小化すべき関数を定義する.

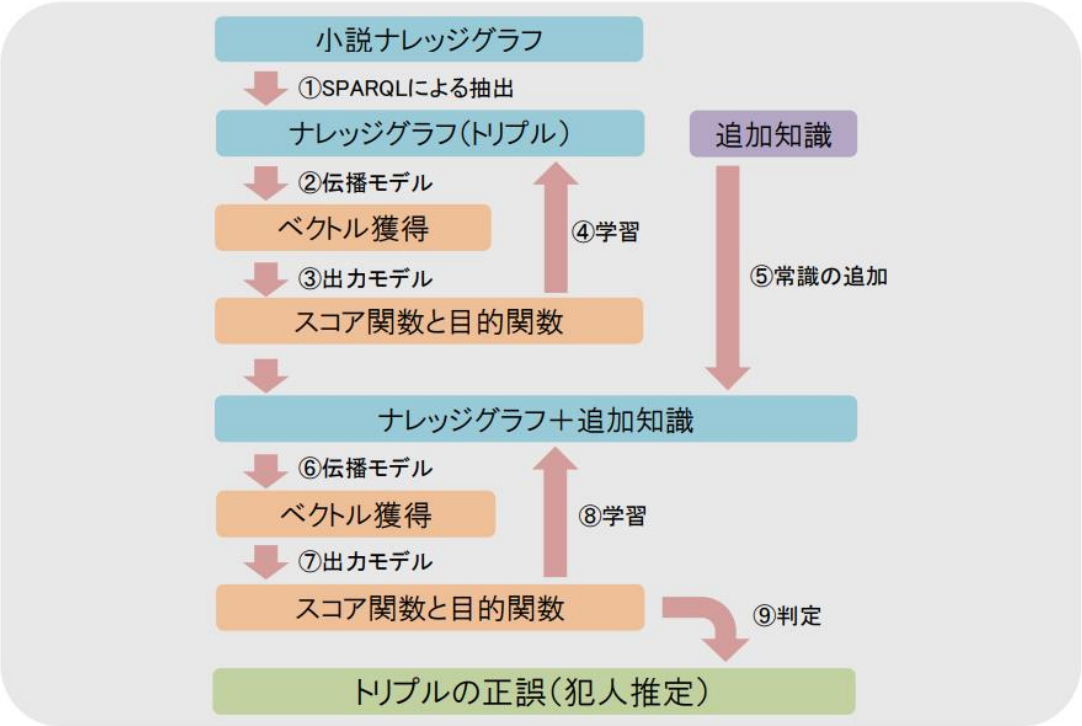
トリプルの評価を行うスコア関数を,

$$f\ (h, r,t) = ||\ \mathbf{V}_h + \mathbf{V}_r\ - \mathbf{V}_t||$$

スコア関数に基づいて最小化すべき関数を,

$\zeta = \sum_i \mathrm{f}(\mathrm{hi}, \mathrm{ri}, \mathrm{ti}) - [\tau - f(h_i', r_i, t_i')]$

として,ナレッジグラフをトリプル化したデータを用いて図 1.8.8 のようなプロセスで解析を行う.



1.8.8 犯人推定プロセス

訓練されたデータに図 1.8.9 の殺害犯に関するデータを加えたことによって図 1.8.10 のトリプルを得ることができた.

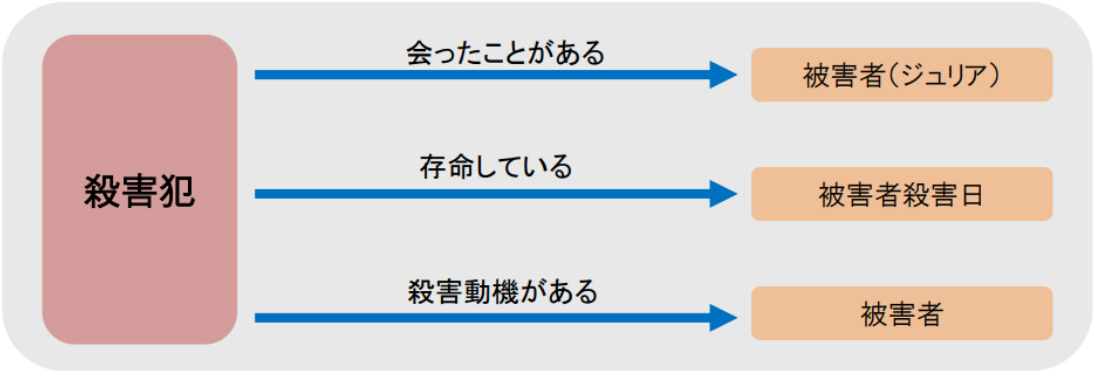


図 1.8.9 追加したエンティティ

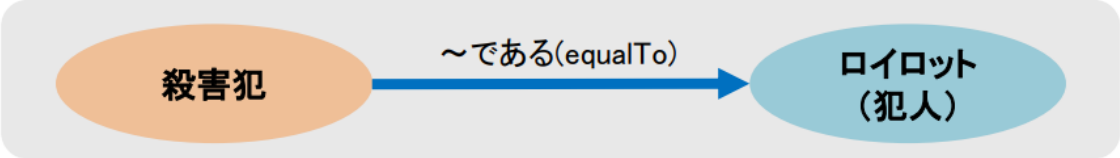
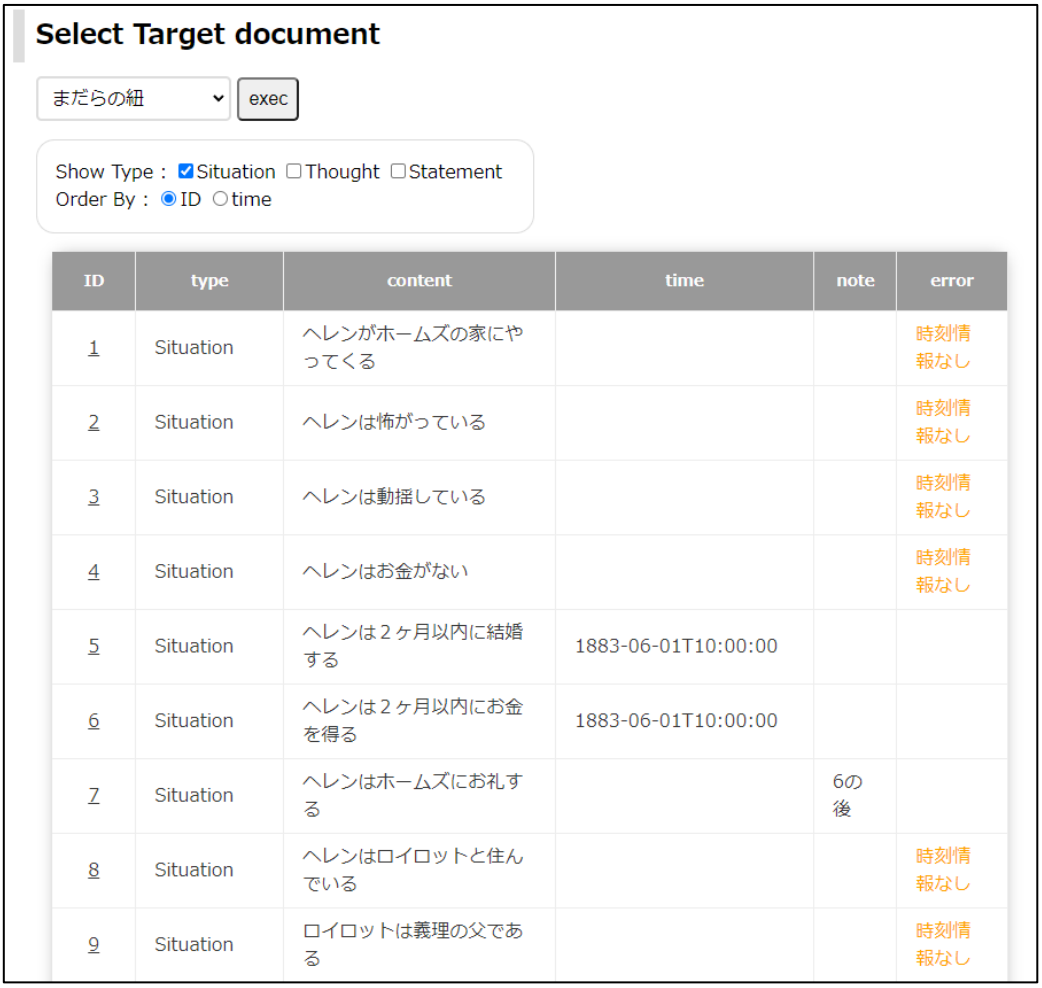


図 1.8.10 補完されたトリプル

1.8.4 ソートツール

2020 年度のツール部門に投稿されたもので,加藤敦丈(プロフィール非公開)[14]が作成した.ナレッジグラフ推論チャレンジのナレッジグラフを ID 順,時間順にソートできるツールである.



Select Target document

まだらの紐 exec

Show Type : Situation Thought Statement
Order By : ID time

ID	type	content	time	note	error
1	Situation	ヘレンがホームズの家にやってくる			時刻情報なし
2	Situation	ヘレンは怖がっている			時刻情報なし
3	Situation	ヘレンは動揺している			時刻情報なし
4	Situation	ヘレンはお金がない			時刻情報なし
5	Situation	ヘレンは２ヶ月以内に結婚する	1883-06-01T10:00:00		
6	Situation	ヘレンは２ヶ月以内にお金を得る	1883-06-01T10:00:00		
7	Situation	ヘレンはホームズにお礼する		6の後	
8	Situation	ヘレンはロイロットと住んでいる			時刻情報なし
9	Situation	ロイロットは義理の父である			時刻情報なし

第 2 章

提案手法

自然言語処理用のデータベースの構築を手動で行うことは実現不可能であるため,本研究では自然言語文を解析器を通して得た出力から自動でトリプルのデータを抽出し,データベースの構築を行うシステムを提案する.また,ナレッジグラフ推論チャレンジのツール部門に提出できるようなシステムを目的とする.

2.1 解析器の情報ラベル

1.2 項で説明した代表的な解析器の中でどれが最もトリプルのデータの抽出に適しているか選定する.各解析器の出力には,単語,用言ごとにラベルが割り当てられており,

2.1.1 COTOHA

12 種類ある API の中の,入力として日本語で記述された分を受け取り,文の構造と意味解析を行い出力をする「構文解析 API」を利用する.入力された文は,文節・形態素に分解され,文節間の係り受け関係や形態素間の係り受け関係,品詞情報などの意味情報を付与する.以下に出力される情報を示す.

COTOHA は表 2.1.1 に示す文節情報オブジェクトと表 2.1.2 に示す形態素情報オブジェクトで 1.3.1 項と 1.3.2 項の解析にあたる情報である.

表 2.1.1 COTOHA 文節情報ラベル

キー名	データ型	説明
id	integer	形態素番号
head	integer	係り先の文節番号
dep	string	係りタイプ(6種)
chunk_head	integer	内容後の形態素開始位置
chunk_func	integer	機能語の形態素開始位置
links	array(object)	掛かり元の情報
predicate	array(string)	機能語の付加情報の配列

表 2.1.2 COTOHA 形態素情報ラベル

キー名	データ型	説明
id	integer	文節番号
form	string	表記
kana	string	カナ読み
lemma	string	lemma
pos	string	品詞
features	array(string)	副品詞の配列
dependency_labels	array(object)	依存先情報の配列
attributes	object	付属情報

表 2.1.3 は意味情報ラベルを示している.

表 2.1.3 COTOHA 意味情報ラベル

意味ラベル名称	説明	例
agent	有意動作を引き起こす主体	私がを食べる。
agentnonvoluntary	主体に意思性のないagent	森が自然の大切さを教えてくれた。
coagent	agentと一緒に動作をする主体	太郎は花子と結婚した。
aobject	属性を持つ対象	花がきれい。
object	動作・変化の影響を受ける対象	私がを食べる。
implement	有意思動作における道具・手段	バットでたたく。
source	事象の主体または対象の最初の位置	を友達からもらった。
material	材料または構成要素	ビーズでを作る。
goal	事象の主体または対象の最後の位置	東京に行く。

意味ラベル名称	説明	例
manner	動作・変化のやり方	和やかにする。
time	事象の起こる時間	7時に起きる。
timefrom	事象の始まる時間	朝から晩まで働く。
timeto	事象の終わる時間	朝から晩まで働く。
basis	比較の基準	アメリカと比べて
unit	単位	100グラム単位で売っている。
fromto	範囲	4巻まで発売されています。
purpose	目的	遊びに出かける
condition	事象・事実の条件関係	雨なので家に帰った。
adjectivals	形容	可愛い姪。
adverbials	形容(動作)	-
other	その他	-

2.1.2 KNP

KNP の意味情報の付与に関しては,入力に用いられている形態素解析システム JUMAN に依存する.しかし,KNP は格解析・照応解析システムであるため単語の格や意味カテゴリという分類はしているが,他の単語との関係を決定する意味役割付与に関して出力する機能が標準として備わっていない.

2.1.3 ASA

ASA は係り受け解析に KNP を用いており,述語項構造解析と意味役割付与を語義概念が記述された EDR 辞書を拡張利用して行う.表 4 に意味情報ラベルについて示す.

表 2.1.4 ASA 意味情報ラベル

Arg0-5(内項),ArgM(付加詞)(仮)	属性	意味
Arg0-5	使役	ある動作や状態，状態変化を引き起こす無生物やイベント．「波が[使役]ゴミを海岸に打ち上げる」
Arg0-5	使役者	ある自主的な動作を相手にさせる人．相手は被使役者となる．「母親が [使役者] 子守唄を [対象] 娘に [被使役者] 聞かせる 」
Arg0-5	動作主	ある動作や状態，状態変化を引き起こす人．「老人が [動作主] 道を [通過点] 横切る 」
Arg0-5	動作主（操作対象）	ある動作や状態，状態変化を引き起こすものが乗り物など人の操作されているものの場合．「バスが [動作主（操作対象）] バス停を [起点] 発進する」
ArgM-CAU	原因	ある動作や状態，状態変化を起こす理由や根拠となるもの．「検察が [動作主] 社長を [対象（人）] 収賄容疑で [原因] 起訴する」
ArgM-CAU	原因（内容物）	ある空間を埋めるという状態変化において，埋めていくもの．「沼を [対象] 土砂で [原因（内容物）] 埋め立てる 」
Arg0-5	対象	ある動作や状態，状態変化の変化を取り上げるそのもの．「守備を [対象] シフトする 」
Arg0-5	対象（事態）	ある動作や状態，状態変化の変化を取り上げるそのものが事態の場合．「合格を[対象(事態)] 祈る」
Arg0-5	対象（人）	ある動作や状態，状態変化の変化を取り上げるそのもの．「 酔っ払いを [対象（人）] 外へ [着点] 追い出す 」
Arg0-5	対象（動作）	ある動作や状態，状態変化の変化を取り上げるそのものが動作の場合．「狩猟を/動作　解禁する」「寄付を/動作　企業に/対象(人) 仰ぐ」
Arg0-5	対象（感情）	ある動作や状態，状態変化の変化を取り上げる対象が感情の場合．「興奮が[対象(感情)]が冷める」
Arg0-5	対象（生成物）	ある動作や状態変化で生成したものを対象として述べているときに付与する．「劇場が[対象(生成物)] 完成する．」
Arg0-5	対象（身体部分）	ある動作や状態で取り上げる対象がある主体の一部であるとき．「彼女が 視線を[対象(身体部分)]私に 投げる」
Arg0-5	相互	ある動作や状態変化がとりあげる対象とともに起こるもの．「私の意見が　彼女の意見と[相互]ぶつかる」
Arg0-5	相互（人）	ある動作や状態変化がとりあげる対象とともに起こる相手．「ケーキを みんなで[相互(人)]分ける」
Arg0-5	着点	ある動作や状態変化の変化終了点での場所．「生徒達が 運動場に[着点] 出る」
Arg0-5	着点（人）	ある動作や状態変化の変化終了点での対象が人の場合．「ニセモノを 客に[着点(人)]掴ませる」

2.1.4 ナレッジグラフ推論チャレンジのデータ

ナレッジグラフ推論チャレンジが公開している RDF のデータは小説の原文を対象文とするのではなく人手で対象文を作成し,ID が割り振られており例を図 2.1.1 に示す.

「ヘレンがホームズの家にやってくる」というデータが 1 つ目のデータとして用意されているが,実際には原文では文章の初めから 3 9 行目,文字数 1225 文字の文章でその場面が構成されている.

1	対象文	ヘレンがホームズの家にやってくる
2	対象文	ヘレンは怖がっている
3	対象文	ヘレンは動揺している

図 2.1.1 ナレッジグラフ推論チャレンジの配布データ

また,データ構成は表 2.1.5 のようになっている.

表 2.1.5 ナレッジグラフ推論チャレンジデータ構造

1	対象文	ヘレンがホームズの家にやってくる
	Type	
	情報源	
	hasPredicate	来る
	主語/Who（誰が）	ヘレン
	Whom（誰に）	
	Where（どこで）	
	From（どこから）	
	To（どこへ）	ホームズの家
	When（いつ）	
	What（何を）	
	Why（なぜ）	
	How（どのように）	
	その他の修飾	
	関連する文	

2.2 開発手法

2.2.1　開発環境と概要

ナレッジグラフ推論チャレンジではシャーロック・ホームズ短編集におけるホームズが解決した事件の話を題材としており,図 2.1.1 と表 2.5 で示している文章は 56 ある短編集のうちの「シャーロックホームズの冒険」に収録されている「まだらのひも」という作品のものであり,この作品は 2018 年度のナレッジグラフ推論チャレンジでコンテストの題材とされた.

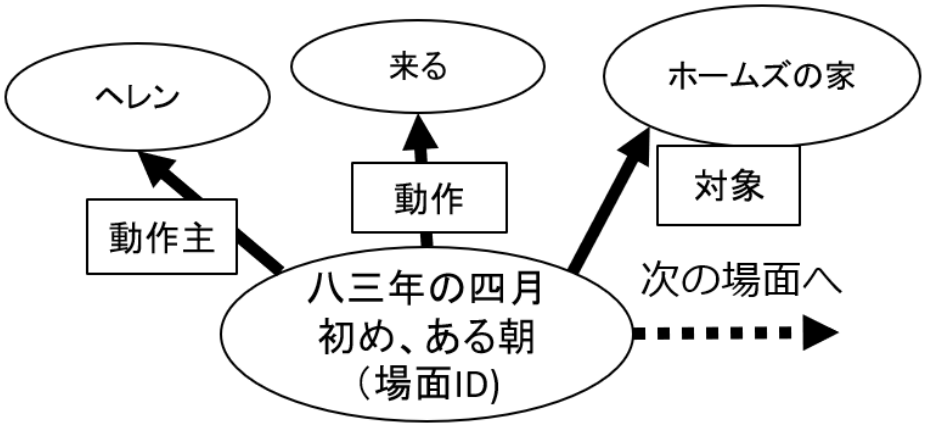


図 2.2.1「まだらのひも」トリプル

図 2.2.1 は表 2.5 に基づいたトリプルでありリストで構造を示すと,表 2.2.1 のようになる.

表 2.2.1 リスト化

場面 ID	動作主	ヘレン
場面 ID	動作	来る
場面 ID	対象	ホームズの家

各解析器の推理小説における解析精度を調べ,また意味情報ラベルが表 2.1.5 のナレッジグラフ推論チャレンジの構造に加工することを考えて解析器を選定し三つ組みの抽出を行う.

第 3 章

自然言語文のトリプル化の結果

3.1 解析器の選定と評価

3.1.1 解析精度

小説「まだらのひも」原文の 30 文を各解析器のデモを利用して解析した際の出力の係り受けの正誤を人手で分類し,各解析器の精度を求めたものを表 3.1.1 に示す.

表 3.1.1 構文解析精度

解析器	正	誤	精度[%]
KNP	27	3	90
ASA	27	3	90
COTOHA	28	2	93.3

1.6 項で形態素解析の精度は 97%程度であると述べたが,それと比較すると精度は低い.原因は,推理小説特有の言い回しや表現の語に対応できていないためである.

2.1 で示した意味情報ラベル付与の精度を表 3.1.2 に示す.

表 3.1.2 意味解析精度

解析器	正	誤	精度[%]
KNP	-	-	-
ASA	25	5	83.3
COTOHA	24	6	80

意味解析の精度は形態素解析の誤りの伝播によってどの解析器も多くの誤りが見られた.また,KNP に関しては格解析による日本語の動詞,形容詞,形容動詞の 3 品詞を意味する用言に関する出力は得られるが意味情報についてはカテゴリが大きく分けられるだけで役割は判別されない.

3.1.2　解析失敗例

解析の失敗要因はどの解析器も共通であり,出力された情報は図 3.1.1 のようなツリー表記や図 3.1.2 のようなリスト表記で示される.

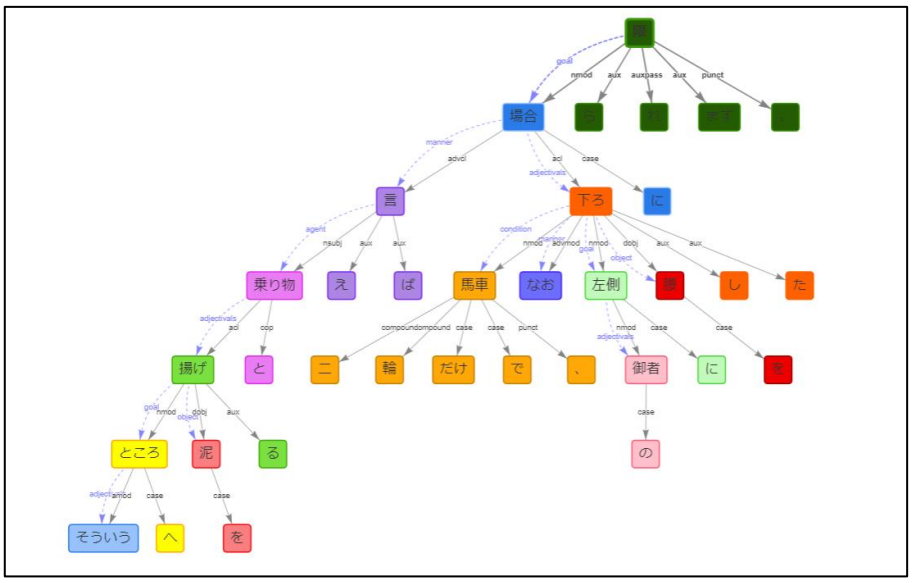


図 3.1.1 構文解析の出力(ツリー表記)

form	kana	lemma	pos	features
そういう	ソウイウ	そういう	連体詞	[]
ところ	トコロ	所	名詞	[]
へ	へ	へ	格助詞	["連用"]
泥	ドロ	泥	名詞	[]
を	ヲ	を	格助詞	["連用"]
揚げ	アゲ	揚げる	動詞語幹	["A"]
る	ル	る	動詞接尾辞	["連体"]

図 3.1.2 構文解析の出力情報(リスト表記)

図 3.1.1 の中での失敗は,例えば図 3.1.3 に拡大図を示すように「御者」という単語は馬車などを操作する運転手であるが[adjectivals]という形容詞を指す意味ラベルが割り振られてしまっている.

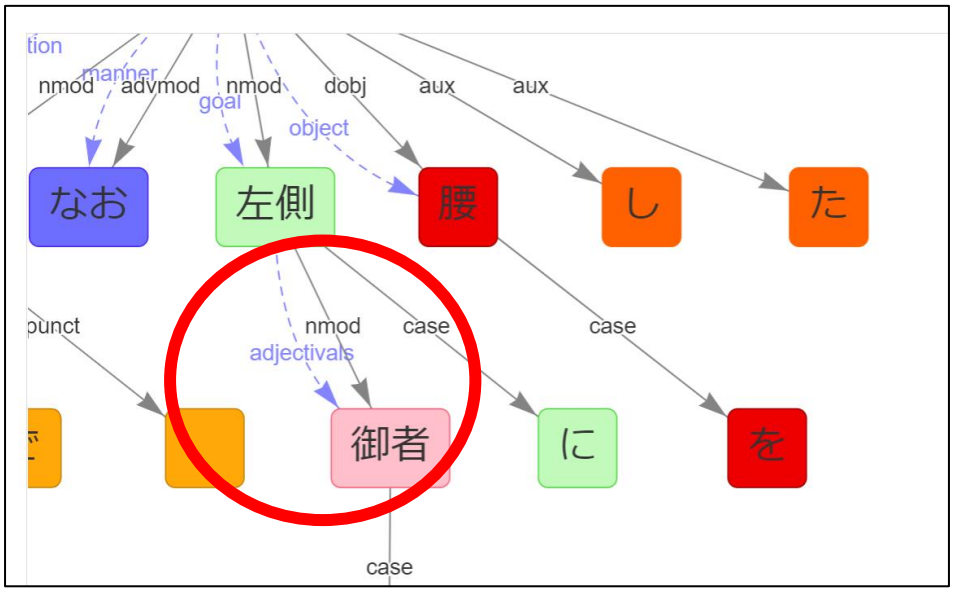


図 3.1.3 意味解析失敗例

KNP はかかり先や語義の解析ミスが他の解析器より多く見られた,失敗例としては「一蹴する」という文の解析を「一」と「蹴」で分けてしまい,「途方もない」のような文を慣用句として識別できず他 2 つの解析器より辞書の規模が小さいことが挙げられる.

ASA は KNP の係り受け解析を用いてさらに独自の述語項構造シソーラスを用いて述語に対して概念を付与し,述語にかかる係り元に意味的な関係を結ぶことが可能であるためかかり先や意味役割の解析の精度は高い.しかし,照応解析の機能がないため照応詞を多用する推理小説の文に用いるには先のことを考えると実用的ではない.

COTOHA は照応解析の機能を有し,精度が KNP より優れている,また利用する際の意味情報ラベルなどの出力情報が他の解析器より利用し易いため,本研究では自然言語文の解析に COTOHA を用いる.

3.1.3 データの同定

COTOHA の構文解析結果をナレッジグラフ推論チャレンジで利用可能にするため意

味役割データを同定する必要がある.表 2.1.3 の意味情報を表 2.1.5 のものに当てはめると表 3.1.3 のようになる.

表 3.1.3 意味情報の同定

ナレッジグラフ推論チャレンジ	COTOHA
HasPredicate	動作
主語/Who(誰が)	agent agentnonvoluntary coagent
Whom(誰に)	aobject object beneficiary
Where(どこで)	place
From(どこから)	source
To(どこへ)	goal
When(いつ)	scene time timefrom timeto
What(何を)	impelement
Why(なぜ)	condition
How(どのように)	副詞
その他の修飾	形容詞

COTOHA の意味情報ラベルのうち図で示されているものを抽出し,
[文の行数(場面 ID),意味情報ラベル,単語]という三つ組みのデータを作成する.

3.2 三つ組みデータの抽出

「まだらのひも」全文(689 行)に対して三つ組みデータの抽出を行い図 3.2.1 のように表形式のファイルに出力を行った.

sentence 0	動作	見ると、
sentence 0	object	覚え書き
sentence 0	動作	ある。
sentence 0	object	事件
sentence 1	動作	通じて
sentence 1	object	事件
sentence 1	動作	考えてきた。
sentence 1	object	手際
⋮		
sentence 687	動作	痛めつけたから、
sentence 687	agent	僕
sentence 687	動作	むき出しして、
sentence 687	object	本性
sentence 687	動作	とまった
sentence 687	place	目
sentence 687	動作	かみついた。
sentence 687	object	人間
sentence 688	動作	あると
sentence 688	place	死
sentence 688	object	責任
sentence 688	動作	言えるが、
sentence 688	agent	僕
sentence 688	動作	感じそうにない。
sentence 688	object	呵責

図 3.2.1　抽出した三つ組みデータ

3.2.1 精度

「まだらのひも」全文のトリプル化を行い出力されたデータの 150 個の正誤を人手で判別すると表 3.2.1 の結果になった.

表 3.2.1 トリプル化精度

正	誤	精度[%]
114	36	76

表 3.1.2 の意味ラベルの正誤より精度が低くなったが,サンプル数の他に表 3.2.2 に示すように意味情報ラベルと語義の不一致だけでなく表 3.1.3 で行ったデータの同定についてずれが生じてしたことが判り,精度の低下が必然的になってしまった.

表 3.2.2 誤解析原因

誤解析の原因	失敗数	比率[%]
意味情報ラベルと語義の不一致	35	97.2
語義の同定のずれ	1	2.7

表 3.2.3 に不一致が生じたラベルについてまとめた表を示す.動作・object ラベルに偏っているがどのラベルでも全体的に誤りが生じている.

表 3.2.3 誤解析原因

ラベル名	不一致数	比率[%]
動作	8	22.9
aobject object beneficiary	8	22.9
agent agentnonvoluntary coagent	4	11.4
place	3	8.6
goal	5	14.3
time timefrom timeto	3	8.56
condition	4	11.4

誤りの例をラベルごとに示す.

表 3.2.4 動作ラベルの誤り

sentence 97	動作	裕福でありました.
sentence 97	time	「一時,

動作ラベルに「裕福で」と余分な単語を含んでしまっているため誤り.

表 3.2.5 object ラベルの誤り

sentence 5	動作	事件でなくてならない.
sentence 5	object	途方ない

「途方もない」という慣用句を識別できずに object のラベルが振られてしまっており誤り.

表 3.2.6 agent ラベルの誤り

sentence 15	動作	回ったばかり.
sentence 15	time	普段
sentence 15	agent	男なのに,

「なのに」という余分な単語を含んでしまっているため誤り.

表 3.2.7 place ラベルの誤り

sentence 22	動作	たたき起こされ,
sentence 22	agent	ハドソンさん
sentence 22	place	音

「音」は場所を示す単語ではないため誤り.

表 3.2.8 goal ラベルの誤り

sentence 40	動作	なって
sentence 40	goal	お話

表 3.2.9 time ラベルの誤り

sentence 101	動作	甘んじた.
sentence 101	time	先代

「先代」は事象の起こる時間として不適切であるため誤り.

表 3.2.10 condition ラベルの誤り

sentence 26	動作	待ってもらっている.
sentence 26	condition	居間

「居間」は事象・事実の条件関係として不適切であるため誤り.

表 3.2.12 同定にずれによる誤り

When(いつ)	scene time timefrom timeto

語義の同定のずれについては,表 3.2.12 のようにナレッジグラフ推論チャレンジのデータ構造の When のデータとして scene,time,timefrom,timeto と複数ラベルを当てはめた際に,timefrom は事象の始まる時間 timeto は事象の終わる時間を指すため When(いつ)という一つの枠では表現しきれない語まで同定してしまったことによる誤りで,この誤りを除去するなら timefrom と timeto を除外すれば解消できると考えられる.

3.3 作成したデータの加工と評価

図 3.2.1 のデータを RDF データで用いられる turtle(.ttl)というデータに書き換え RDF ストアの Fuseki にデータをロードさせて SPARQL での検索を可能にした. また,1.8.3 項で紹介した過去のナレッジグラフ推論チャレンジに投稿されたツールで,ナレッジグラフ推論チャレンジが公開しているナレッジグラフを,ニューラルネットワークを用いてトリプルのデータにベクトルを追加している.そこに本研究で作成したトリプルのデータを加えることによって,小説について全てのデータを訓練に用いることができることが考えられる.しかし,本研究で分かったことは,全ての自然言語文に適用できるような汎用性のあるトリプル化は困難であるということである.今回のトリプル化における精度を解消するには,推理小説を解析しているため,シャーロックホームズに関する単語や言い回しをまとめた辞書であったり,物語の内容に沿った密室殺人の殺害リストやインドに生息する生き物のリストなど解析する対象に関わる情報が必要であると考えられる.しかし,形態素解析や意味解析の精度以外でも,ニューラルネットワークのようなディープラーニングをビッグデータなどを用

いて自動で最適なラベルに振り分けたりすることでラベルと語義の不一致などは解消できると考えられる.

第 4 章

おわりに

4.1 まとめ

本研究では自然言語処理の分野で三つ組みのデータを人手で用意することを問題点として,その工程を自動化するため NTT Communications の提供する COTOHA API の解析を利用してその出力から三つ組みのデータを抽出した.

ナレッジグラフ推論チャレンジの題材の推理小説「まだらのひも」の全文の三つ組みの抽出を確認し,RDF ストアにデータをロードさせて SPARQL での検索を可能にした.

評価の結果,推理小説のような独特な文体であったり用言に対して解析の上流工程の段階から誤りが出てしまう事から汎用性は低いということが分った.

4.2 今後の課題

本論文では利用しなかった自然言語分野の技術でオントロジー辞書いうものがあり,その分野に適した概念辞書のことを指し,本研究で扱った「まだらのひも」に対してはシャーロック・ホームズオントロジーといったようにシャーロック・ホームズに関する人物,物,言い回しなどを含んだ辞書があれば解析の精度は上がると考えられる.

また,精度が向上したのちに照応解析を用いて全ての照応詞を明らかにし,図 3.2.1 で省略されている三つ組みのデータの agent にあたる単語を抽出することで機械判読により適したデータになると考えられる.

3.3 項でも述べたようにニューラルネットワークをラベルの振り分けや,もしくはそれの前段階である形態素解析や意味解析に適用することによっても解消できると考えられる.

謝辞

本研究を行うにあたって，指導教員として数々のご指導を賜りました相馬 隆郎准教授に心 よりの感謝を申し上げます.

参考文献

[1] 高度情報通信ネットワーク社会推進戦略本部," オープンデータ基本指針", 官民データ活用推進戦略会議決定,令和元年 6 月 7 日改正,令和 3 年 6 月 15 日改正

[2] Apache Jena,"https://jena.apache.org/documentation/fuseki2/",Apache Jena Fuseki,2021-11-15

[3] KATO Fumihiro,"DBpedia Linked Data Project", 2017, vol. 60, no. 5, p. 307-315.

[4] ナレッジグラフ推論チャレンジ," https://challenge.knowledge-graph.jp/2021/", 人工知能学会セマンティックウェブとオントロジー研究会,2021-11-10

[5] COTOHA API," https://api.ce-cotoha.com/contents/index.html", NTT Communications,2021-10-10

[6] KNP," https://nlp.ist.i.kyoto-u.ac.jp/?KNP", [日本語] / [English]
京都大学 大学院情報学研究科 知能情報学専攻 知能メディア講座 言語メディア分野, 2021-10-10

[7] ASA," http://www.cl.cs.okayama-u.ac.jp/", 岡山大学学術研究院　自然科学学域
工学部数理データサイエンス系 情報工学コース 自然言語処理学研究室,2021-10-10

[8] 末木 顕人, 兼岩 憲,Wikipedia 記事からの中間 RDF グラフと DBpedia トリプルの出, SIG-SWO-045-03,2018

[9] 玉川 奨, 関本 有佳, 森田 武史, 山口 高平, 慶應義塾大学, 青山学院大学. “日本語 Wikipedia からプロパティを備えたオントロジーの構築”. The 25th Annual Conference of the Japanese Society for Artificial Intelligence, 2011

[10] 岡嶋 成司，鵜飼 孝典，村上 勝彦，高松 邦彦，岸田あおい," https://github.com/okajima-s/kgc2018-fllml/blob/master/presentation/20181125_kgc_FLL-ML_presentation.pdf", Nov. 30 2018

[11]　松下 京群,金子 貴美,吉川 和,小林 賢司,小柳 佑介,鵜飼 孝典,西野 文人,織田 充, " https://www.slideshare.net/TakanoriUgai/ss-124021828 “, "「まだらの紐」事件の別アプローチ, Nov. 26, 2018

[12]　勝島 修平, 穴田 一" https://challenge.knowledge-graph.jp/submissions/2020/Katsushima/Katsushima.pdf “, 東京都市大学, 2018

[13]　田村 光太郎,外園 康智," https://challenge.knowledge-graph.jp/submissions/2018/tamura/submission_tamura.pdf “, 株式会社野村総合研究所,2018

[14]　KnowledgeGraph Viewer for KGRC," https://museums-info.net/kgrc_tool/" ,加藤 敦丈, 2020