修士論文

小説からの対話コーパスの自動構築

1710137　DU,Yulong

主指導教員　白井 清昭
審査委員主査　白井 清昭
審査委員　長谷川 忍
飯田 弘之
東条 敏

北陸先端科学技術大学院大学
先端科学技術研究科
(情報科学)

平成 31 年 2 月

Abstract

In recent years, there are many attempts of research and development of dialog systems. A dialog system is a system that interacts with humans through dialog in natural language. Especially, free conversational systems that can chat with human beings have been paid much attention. However, a large amount of dialog corpora are required for research and development of free conversational systems. A dialog corpus is a database that collects a large amount of dialogs between two or among three or more people. However, it is difficult to construct a large-scale dialog corpus, since it requires much cost to record and transcribe dialogs between humans and to eliminate personal information from dialogs for protection of privacy.

This thesis aims at automatically constructing a large-scale dialog corpus by extracting consecutive utterances made by multiple characters in novels. Utterances in a novel can be regarded as speech by people, and consecutive utterances can be regarded as a dialog. Dialog in a novel is not spontaneous; it is made by an author. However, it is natural as one made by humans. Thus, a dialog corpus excerpt from novels can be useful for developing a free conversational system that realizes natural chat. A wide variety of topics are appeared in novels. Furthermore, the number of novels in the world is quite huge. Therefore, novels are an appropriate information source for automatic construction of a dialog corpus. However, it is insufficient to only extract utterances from a novel. In a dialog corpus, it is required to give a speaker for each utterance. This thesis proposes a method to identify a speaker of utterance in a novel, and extract utterances and their speakers as a dialog corpus. It is the first attempt to automatically construct a dialog corpus from Japanese novels.

In the proposed method, preprocessing is first performed. Metadata other than a text is removed, then a text is split into sentences. Next, utterances are extracted from a novel. By pattern matching with regular expression, sentences in parentheses are extracted as utterances. Next, characters are extracted from a novel. First, words or compound words detected as "person name" by a named entity extraction tool are extracted as characters. CaboCha is used as a named entity extraction tool in this study. In addition, nouns that have a semantic class of people in a thesaurus are also extracted as characters. Specifically, our system extracts nouns in the semantic classes of "person name" and "person" in the Japanese thesaurus Nihongo-goi-taikei.

Next, a speaker is identified for each extracted utterance. We define two types of speakers: an explicit speaker and implicit speaker. An explicit speaker is a speaker who is clearly stated in a novel as he/she says a certain utterance. On the other hand, an implicit speaker is a speaker who is not explicitly indicated in a novel but can be understood by readers that he/she says it. First, patterns to extract explicit speakers are designed. For example, in a pattern "A says that B.", the person A is extracted as a speaker of the utterance B. When an utterance is embedded in a sentence and its speaker is not identified by the above mentioned patterns, it is regarded as not an utterance, although it is once extracted as an utterance. Next, implicit speakers are identified. Several patterns are made to extract characters around an utterance, then they are used to identify an implicit speaker. Finally, speakers are identified by using a speaker alternation pattern when the speaker identification is failed by the pattern matching. The speaker alternation pattern assumes that speakers of consecutive utterances alternate in turn. Using this pattern, speakers are identified by referring a speaker of previous or

succeeding utterance that is identified by the patterns to extract explicit and implicit speakers. The speaker alternation pattern is applied repeatedly until speakers of all the utterances are identified, or until speakers are not identified any more. Finally, consecutive utterances with their speakers identified by the above procedures are extracted to construct a dialog corpus. Names of characters in a novel are replaced with speaker IDs such as "speaker A" and are tagged in a dialog corpus, instead of giving a character name as it is.

An experiment was conducted to evaluate our proposed method. Four novels were randomly chosen from novels published in Aozora Bunko as a test data. For each extracted utterance in the test data, its speaker is manually tagged as gold data. The proposed method to identify speakers was evaluated on this test data. The applicability, which was defined as a ratio of the number of utterances that the system can determine the speaker to the total number of speakers, of the proposed method was 1 in all the novels. That is, speakers could be identified for all utterances. On the other hand, the precision for the speaker identification was 0.72. In the novel for which the system most poorly performed, the characters were represented by nicknames, and they could not be extracted as person names by the methods using named entity extraction or thesaurus. If there is a list of characters as metadata of a novel, speakers of utterances can be identified more accurately, resulting improvement of the recall. Finally, dialogs were extracted by the proposed method from 4,836 novels in Aozora Bunko. The applicability of speaker identification was 0.917. The number of dialogs that consists of consecutive utterances where the speakers of all utterances are identified was about 19,000. The average of the number of utterances per dialog was 10. It indicates that a comparatively large dialog corpus can be constructed automatically by the proposed method.

概 要

近年、対話システムの研究と開発が盛んに行われている。対話システムとは、対話によって人間とインタラクションを行うシステムである。特に人間と雑談できる自由対話システムは、近年その需要が増している。ただし、自由対話システムの研究と開発には大量の対話コーパスが必要である。対話コーパスとは、2名もしくは3名以上の人間同士の対話を大量に集めたデータベースである。しかし、人間同士の対話を収録したり書き起こしたりする作業のコストが高いこと、また個人情報保護などの問題もあるため、大規模な対話コーパスを構築することは難しい。

本研究では、小説から複数の登場人物による連続した台詞を抽出し、それを対話として大量に集めた対話コーパスを自動的に構築することを目的とする。小説における台詞は人の発話であり、複数人による連続した台詞は対話とみなすことができる。小説における対話は著者の作例であるが、対話としては自然なので、これを集めた対話コーパスは、自然な雑談を実現する自由対話システムの開発に有用である。また、小説では様々なトピックの対話が出現し、数も非常に多い。したがって、小説は対話コーパスを自動構築するための情報源として適している。しかし、小説から単に台詞を抽出するだけでは不十分である。対話コーパスではそれぞれの発話に対して話者の情報を付与することが求められる。本研究では、台詞を発した登場人物を特定し、その人物とともに台詞を抽出する手法を提案する。本研究は、日本語小説を対象に、小説から対話コーパスを自動構築する初めての試みである。

提案手法では、まず、小説のテキストに対し、本文以外のメタデータの除去や文への分割などの前処理を行う。次に、小説から台詞を抽出する。正規表現によるパターンマッチにより、括弧で囲まれた文を台詞として抽出する。次に、小説から登場人物を抽出する。まず、固有表現抽出によって「人名」として検出された単語もしくは単語列を登場人物として抽出する。具体的には、CaboCha によって「PERSON」とタグ付けされた単語また単語列を抽出する。また、シソーラスで人物に相当する意味クラスを持つ名詞を登場人物として抽出する。シソーラスとして日本語語彙大系を利用し、「人名」「人」のカテゴリに含まれる語を抽出する。次に、抽出した個々の台詞に対し、その話者を特定する。話者を特定する際に、台詞の話者を明示的な話者と暗黙的な話者の2種類に分ける。明示的な話者とは、ある台詞を発話したことが小説のテキストに明記されている話者を指す。一方、暗黙的な話者とは、台詞を発っしたことが明示されていないが間接的に分かる話者を指す。まず、明示的な話者を特定するパターンを作成する。例えば、「A は B と言った」といったパターンでは、人物 A を台詞 B の話者として抽出する。パターンマッチによって明示的話者を特定できない台詞のうち、文の中に埋め込まれている台詞については、それを台詞として検出した処理を取り消し、台詞で

はないものとみなす。次に、暗黙的な話者を特定する。ここでは台詞の周辺に出現する人物を検出するパターンを作成し、そのパターンマッチによって暗黙的な話者を特定する。最後に、話者を特定できない台詞に対して、話者交替パターンを使って話者を特定する。話者交替パターンとは、連続する台詞の話者が交互に交替すると仮定し、既に特定された明示的話者・暗黙的話者から、その前後に出現する台詞の話者を推定するものである。話者交替パターンは、全ての台詞の話者が特定されるまで、あるいは話者交替パターンによって話者を特定できる台詞がなくなるまで、繰り返し適用する。最後に、上記の手続きで得られた連続した話者情報付きの台詞を対話として収集し、対話コーパスを構築する。話者の情報として、小説の人物名をそのまま付与するのではなく、同一人物を「話者 A」のような記号に置き換える。

提案手法の評価実験について述べる。青空文庫で公開されている小説からランダムに 4 つの小説を選択し、テスト用データとする。抽出したそれぞれの台詞に対して、その発話者を人手でタグ付けし、これを正解データとして、台詞の話者の認識手法を評価する。提案手法の適用率は全ての小説で 1 となった。つまり、全ての台詞について話者を特定できた。一方、話者特定の正解率は 0.72 となった。正解率が一番低かった小説では、登場人物がニックネームで表現されていて、固有表現抽出やシソーラスを用いた手法では人名と認識できていなかった。登場人物リストのような情報があれば、登場人物抽出の再現率が上がり、話者を正確に特定できるようになると考えられる。

最後に、青空文庫に掲載されている 4,836 件の小説に対して、提案手法によって対話を抽出した。話者特定の適用率は 0.917 であった。連続して出現しかつ全ての台詞の話者を特定できた対話の数はおよそ 19,000 件であった。1 つの対話に含まれる発話数の平均は 10 であった。以上より、提案手法により比較的大規模な対話コーパスを自動構築できることを確認できた。

目 次

図 目 次

表 目 次

第1章　はじめに

1.1　背景

ここ数年、人工知能の研究の発展に伴って、対話システムの研究と開発が盛んに行われている。対話システムとは、対話によって人間とインタラクションを行うシステムである。特に近年、様々な対話システムが実用化され、私たちの生活の中でも使われ始めている。例えば、Microsoft 社が提供した「Cortana」、Apple 社が提供した「Siri」は、我々が手軽に利用できる代表的な対話システムである。

対話システムは大きく二つの種類に分けられる。一つはタスク指向型対話システム、もう一つは非タスク指向型対話システムである。タスク指向型対話システムとは、使用者が何らかの情報を求めるようなタスクがあり、それを達成するために必要な対話を行うシステムである。それに対して、非タスク指向型対話システムとは、タスクを限定せず、人間と自由に雑談できる対話システムである [1]。特に近年、スマートフォンが普及し、また IoT の研究の発展に伴って、家庭用ロボットが開発されたことから、人間がコンピュータと対話する機会が増え、人間と雑談できる自由対話システムの需要が増していると考えられる [2]。非タスク指向型対話システムは自由対話システムとも呼ばれている。

自由対話システムの研究と開発のためには、実際の人と人との間の対話を収録し、音声をテキストに書き起こした対話コーパスが欠かせない。対話コーパスは、一般に、2 名または 3 名以上の対話を収録し、必要に応じて話者の情報を付与するなどの構造化が行われる。特に機械学習のアプローチを利用する場合には、訓練データとしての大量の対話コーパスが必要である。自由対話システムは、訓練データを大量に蓄積し、それから自然な対話を学習することにより「賢く」なっていくと考えられる [3]。

しかし、実際には、人間同士の対話データを収録したり、書き起こしたりする作業には多くの時間を要し、その工賃もしくは作成コストが高いという問題がある。一方、さまざまな人間同士の対話を収録する場合には、プライバシー、個人情報保護などの問題もある [4]。以上のような理由から、一般に、大規模な対話コーパスを構築することは難しい。

1.2 目的

本研究では、非タスク指向型対話システムの開発のために、小説から複数の登場人物による連続した台詞を抽出し、それを対話として大量に集めた対話コーパスを自動的に構築することを目的とする。

表 1.1: 小説における対話の例

「じゃ、あしたは出入の商人の方はどうしましょう」
方太太は突然押掛けて来て床の前に突立った。
「商人？……八日の午後来いと言え」
「わたしにはそんなことが言えません。向うで信用しません、承知しません」
「信用しないことがあるもんか。向うへ行って聞けばわかる。
役所じゅうの人は誰一人貰っていない。皆八日だ」
彼は人差指を伸ばして蚊帳の中の空間に一つの半円を画いた。

(魯迅「端午節」(井上紅梅訳) より)

小説における台詞は人の発話であり、複数の登場人物による連続した台詞は対話とみなすことができる。表 1.1 は、実際の小説に出現する連続した発話、すなわち対話である。小説における対話は作家の作例であり、完全に自然な対話とは言えないが、作家は自然に発生する対話を想定しているため、自然な対話に近い性質を有すると考えられる。

小説は対話コーパスを自動的に構築する際の情報源として適している。小説では様々なトピックの対話が出現しているため、これらを網羅的に収集することにより、多様なトピックやそれに関する発話を含む対話コーパスを構築できる。一方、現存する小説の数は非常に多い、また小説のデジタル化が進んでいるため、大量の小説を集めることは比較的容易である。小説を大量に収集し、それから対話を抽出すれば、大規模な対話コーパスを低コストで構築できるというメリットがある。以上の理由から、小説から対話コーパスを構築することは、自然な雑談を実現する自由対話システムの開発に有用であるといえる。

ただし、対話コーパスの構築を目的とする場合には、小説から単に連続した台詞を抽出するだけでは不十分である。話者交替のタイミングを決める研究や三者以上の対話の研究に利用するためには、人の発話を集めるだけでなく、それぞれの台詞を発っした話者を特定し、話者の情報を付与した発話 (台詞) の列を抽出する方が望ましい。そのため、本研究では、台詞を発した登場人物を特定し、その人物とともに台詞を抽出する手法を探究する。

本研究は青空文庫 [5] で公開されている小説を用いた。青空文庫は、著作権が切れたおよそ 4,836 件の小説を公開しているウェブサイトである。小説から対話を抽出する手法は青空文庫のいくつかの小説を対象に開発を進めた。また、提案手法の評価も青空文庫の小説を用いた。ただし、青空文庫の小説に限らずどんな小説にも適用可能な汎用的な手法を提案する。

1.3　本論文の構成

本論文の構成は以下の通りである。第 2 章では、関連研究について述べ、また本研究との違いについて論じる。第 3 章では、本論文の提案手法について説明する。特に、台詞の発話者を特定する手法を重点的に述べる。第 4 章では、提案手法の評価実験について説明し、実験結果を考察する。また、青空文庫から構築された対話コーパスについて報告する。最後に、第 5 章では、本論文の成果を総括し、結論を述べる。また、今後の課題について述べる。

第2章　関連研究

本章では、本研究の関連研究について説明する。本研究の目的は、自由対話システムの基盤となる対話コーパスを構築する手法を探求することである。したがって、2.1 節では、自由対話システムに関する先行研究を紹介する。また、本研究のもう一つの目的は、小説における台詞を発した登場人物を特定し、その人物とともに台詞を抽出する手法を探究することである。2.2 節では、小説から登場人物を抽出する先行研究について説明する。2.3 節では、台詞の話者を特定する先行研究について説明する。最後に、2.4 節では、本研究と関連研究との違いについて説明する。

2.1　自由対話システム

畑らは、複数の言語資源を用いて、ユーザの入力から対話文を動的に生成するシステムを提案した [1]。彼らが提案したシステムでは、ユーザが入力した文から話題語を抽出する。次に、話題語を基点に Web 日本語 N グラムを検索し、その単語を含む文字列を再帰的に繋げることで応答文を生成する。この研究の提案システムでは、ユーザの発話から話題語を抽出するため、システムが出力する発話はユーザの発話との適性が高い。また、大量のウェブデータから構築された Web 日本語 N グラムを用いて応答文を生成していることから、対話システムが提供する話題の多様性も高い。しかし、応答文は動的に生成されているが、その質は自由対話システムで要求される水準に及ばないことも多い。すなわち、システムが生成する応答文の質が低い。そのため、より自然な発話を自動的に生成する技術が必要である。

小林と萩原は、ユーザの発話内容を記憶し、嗜好や人間関係を考慮する非タスク指向型対話システムを提案した [2]。この研究では、ユーザの発話内容からその嗜好を推定し、またシステムが発話選択する際に、ユーザの嗜好に加えて、人称を推定することにより、ユーザと他者との人間関係を考慮する。ユーザの個人情報に応じて、それぞれのユーザに対して適した発話を生成することが可能になる。

2.2　小説からの人物抽出

西原と白井は物語テキストから登場人物ならびにそれらの人物関係を抽出する手法を提案した [6]。

```
太郎と花子は2020年の東京オリンピックを見に行きたい。
* 0 1D 0/1 1.629817
太郎    名詞,固有名詞,人名,名,*,*,太郎,タロウ,タロー    B-PERSON
と      助詞,並立助詞,*,*,*,*,と,ト,ト  O
* 1 5D 0/1 -2.318426
花子    名詞,固有名詞,人名,名,*,*,花子,ハナコ,ハナコ    B-PERSON
は      助詞,係助詞,*,*,*,*,は,ハ,ワ    O
* 2 3D 1/2 1.679104
2020    名詞,数,*,*,*,*,*       B-DATE
年      名詞,接尾,助数詞,*,*,*,年,ネン,ネン      I-DATE
の      助詞,連体化,*,*,*,*,の,ノ,ノ    O
* 3 4D 1/2 1.165264
東京    名詞,固有名詞,地域,一般,*,*,東京,トウキョウ,トーキョー  B-LOCATION
オリンピック    名詞,一般,*,*,*,*,オリンピック,オリンピック,オリンピック        O
を      助詞,格助詞,一般,*,*,*,を,ヲ,ヲ  O
* 4 5D 0/1 -2.318426
見      動詞,自立,*,*,一段,連用形,見る,ミ,ミ    O
に      助詞,格助詞,一般,*,*,*,に,ニ,ニ  O
* 5 -1D 0/1 0.000000
行き    動詞,自立,*,*,五段・カ行促音便,連用形,行く,イキ,イキ    O
たい    助動詞,*,*,*,特殊・タイ,基本形,たい,タイ,タイ    O
。      記号,句点,*,*,*,*,。,。,。      O
EOS
```

図 2.1: CaboCha による解析例

まず、彼らは固有表現抽出による手法として、CaboCha[7] による固有表現抽出で「PERSON」とタグ付けされた語を人物として抽出する。CaboCha とは、Support Vector Machines に基づく高性能な日本語係り受け解析器である。文節の係り受け解析だけでなく、入力した文に対して固有表現抽出を行うことができる。抽出される固有表現の種類は IREX(Information Retrieval and Extraction Exercis) の定義に基づく。例えば、図 2.1 のように、「太郎と花子は 2020 年の東京オリンピックを見に行きたい。」という一文を入力すると、「太郎」と「花子」という人名に対して、「PERSON」とタグ付けする。一方、「東京」という地名には「LOCATION」とタグ付けする。

表 2.1: 日本語語彙大系の「人名」に登録されている単語の例

眞田、眞島、眞嶋、眞奈美、眞鍋、
眞二、眞之、眞之助、眞能、眞板

次に、西原と白井の研究では、シソーラスを用いて登場人物を抽出している。具体的には、日本語語彙大系 [8] における「人名」「人」のカテゴリに含まれる語を人物として抽出する。表 2.1 は「人名」のカテゴリに登録されている単語の例である。日本語語彙大系とは大規模な日本語シソーラスである。 NTT の日英機械翻訳システム ALT-J/E で用いられているコンピュータ用辞書を再編集したもので、30 万語の日本語単語と 14,000 件の文型パタンが収録されている。また、収録されて

いる 30 万語は 3,000 種の意味分類カテゴリに分類されており、最大規模の日本語シソーラスである。

さらに、この研究では、登場人物を抽出した後、親子関係や夫婦関係などさまざまな人物間の関係を推定している。しかし、この研究では、小説から登場人物を抽出してはいるが、台詞とそれを発した人物の対応は決めていない。

小林は物語をシーンごとに分割する手法を提案した [10]。既存の辞書などを利用して場所、時間、人物候補を抽出し、これらの 3 種類の候補の異なり数を基準としてシーンを分割する。小説から対話を獲得する際には、分割されたシーン毎に台詞を抽出する手法が考えられる。この研究では、小説をシーンに分割することで、人物が特定の場面に存在するか否かの「入退場情報」を決めることができるが、台詞とその発話者の対応は決めていない。本研究において、トピック (シーン) が異なる対話を個別に獲得するときに、この手法を適用できる可能性がある。

2.3 台詞の話者の特定

He らは、英語の小説を対象とし、台詞の発話者を推定する手法を提案した [9]。

Category	**Example**
Implicit speaker	*"Don't keep coughing so, Kitty, for heaven's sake!"*
Explicit speaker	*"I do not cough for my own amusement," replied* **Kitty**.
Anaphoric speaker	*"Kitty has no discretion in her coughs," said* **her father**.

図 2.2: He ら [9] による話者の分類

まず、He らは台詞の話者を図 2.2 に示す 3 種類に分類した。暗黙的な話者 (Implicit speaker)、明示的な話者 (Explicit speaker)、照応的な話者 (Anaphoric speaker) の 3 種類である。明示的な話者とは、ある台詞を発話した登場人物が小説のテキストの中に明確に記述されている話者である。それに対して、暗黙的な話者とは、台詞を発っしたことが明記されていない話者である。照応的な話者とは、台詞を発話した登場人物が代名詞などの照応表現により記述されている話者である。

次に、小説の中から明示的に書かれている台詞の話者をパターンマッチで抽出し、話者の候補リストを作った。話者の候補リストにおける人物はその小説における全ての台詞の話者の候補とする。次に、それぞれの台詞に対し、話者の候補の中から適切な話者を決定するランキングモデルを学習している。このとき、正解の話者がタグ付けされた訓練データを必要とする。

さらに、He らは、話者推定の精度を向上させるために、次に示す話者交替パターンを提案した。

- 連続している台詞の発話者はたいてい人物である。
- 一つの対話において、n 番目の台詞の話者と $n-2$ 番目の台詞の話者は同一人物である。

すなわち、話者交替パターンとは、連続した台詞の話者は 2 名の登場人物が交互に台詞を言うという制約である。しかし、このパターンは登場人物が 2 名の場合を想定しており、3 名以上の人物による対話文には適用できない。

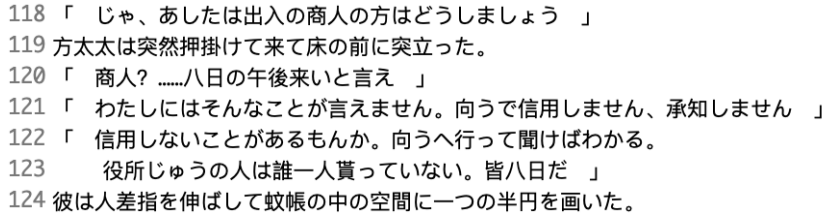

図 2.3: 話者交替パターンの例 (魯迅「端午節」(井上紅梅訳) より)

図 2.3 の例を用いて話者交替パターンを説明する。この対話では「方太太」と「彼」という 2 人の登場人物がいる。118 行目、121 行目の台詞の話者は「方太太」であり、120 行目、122 と 123 行目の台詞の話者は「彼」である。また、二人が交互に台詞を発していることがわかる。He らはこのような話者が交替するパターンも話者を特定するために利用している。

Elson と McKeown は、ルールベースの手法と機械学習により、物語テキストにおける台詞の発話者を特定する手法を提案した [11]。まず、英語の小説を対象として、人手で発話者を特定した 3,000 以上の台詞を含むコーパスを構築した。次に、台詞を表 2.2 に示す 7 種類の構文カテゴリーに分類した。

表 2.2: 構文カテゴリー ([11] より)

Syntactic category	Definition	Prediction
Added quote	⟨OTHER QUOTE by PERSON1⟩⟨TARGET QUOTE⟩	PERSON1
Quote alone	⟨TARGET QUOTE⟩	
Appraent conversation	⟨OTHER QUOTE by PERSON1⟩ ⟨OTHER QUOTE by PERSON2⟩ ⟨TARGET QUOTE⟩	PERSON1
Character trigram(1)	⟨TARGET QUOTE⟩⟨PERSON1⟩⟨EXPRESS VERB⟩	PERSON1
Character trigram(2)	⟨TARGET QUOTE⟩⟨EXPRESS VERB⟩⟨PERSON1⟩	PERSON1
Anaphora trigram	⟨TARGET QUOTE⟩⟨PRONOUN⟩⟨EXPRESS VERB⟩	
Backoff		

⟨TARGET QUOTE⟩ は話者を決めるべき台詞、⟨OTHER QUOTE⟩ はそれ以外の台詞、⟨PERSON⟩ は人物、⟨EXPRESS VERB⟩ は発言を表す動詞である。Prediction は ⟨TARGET QUOTE⟩ の発話者を PERSON1 に決めることを表す。

追加型台詞 (Added quote) とは、ある台詞の直後に段落の切れ目なしに続く別の台詞であり、この話者は直前の話者と同じとみなす。独立型台詞 (Quote alone) とは、それだけで段落を構成する台詞である。明示的な会話 (Apparent conversation) とは、連続した台詞から構成される会話であり、2 名の人物が交互に台詞を言うという制約から、3 番目の台詞の発話者を 1 番目の台詞の発話者に決める。人物トライグラム (Character trigram) とは、台詞、登場人物、発言を表す動詞という 3 つの隣接するトークンの並びであり、台詞の話者を PERSON1 と特定する。この構造カテゴリーは 2 種類あるが、⟨PERSON1⟩ と ⟨EXPRESS VERB⟩ の順序が違う。照応トライグラム (Anaphora trigram) とは、人物トライグラムと似ているが、登場人物ではなく代名詞 PRONOUN を含む 3 つのトークンの並びである。バックオフ (Backoff) とは、上記の構文カテゴリに分類されない台詞である。台詞をパターンマッチによって構文カテゴリーに分類し、これにより話者を決めることができる場合には話者を決定する。

表 2.3: 発話者を特定する分類器の素性 ([11] より)

1. 人物候補と台詞の距離 (単語数)
2. 人物候補と台詞の間に句点が存在しているか、またその句点の種類。段落の切れ目を含む。
3. 人物候補が台詞の近くにある全ての人物候補の中で何番目に台詞に近いか
4. 最近の台詞のうち、人物候補が発した台詞の割合
5. 段落に出現する人名、台詞、単語の数
6. 人物候補が出現する数
7. 人物候補と台詞の近くに、発言を表す動詞、句点、別の人物が出現するか
8. 台詞自身の素性：台詞の長さ、文中にある位置、人物を含むか

構文カテゴリーで話者を特定できない場合、機械学習の手法で話者を決定する。具体的には、台詞と発話者の候補の組に対し、その候補が真に台詞の発話者になっているかを判定する分類器を学習する。この際、表 2.3 に示す素性を用いた。

2.4 本研究の特色

本研究は、以上に述べた先行研究を参考しつつ、小説における台詞の中でも特に連続して出現する台詞の話者を特定する手法を提案し、話者の情報を付与した自由対話コーパスを自動構築する方法を提案する。本研究では、日本語小説から対

話を抽出することで対話コーパスを自動構築する初めての試みである。また、大量の小説から大規模な対話コーパスを自動構築する技術は、対話システムの研究にとって非常に有意義である。

本研究の主な目的は、小説における台詞の話者を正確に同定する手法を確立することである。ただし、He らの手法 [9]、Elson と McKeown の手法 [11] とは異なり、正解の話者がタグ付けされた正解データを必要としない手法を提案する。

第3章　提案手法

本章では、小説から複数の登場人物の台詞を抽出し、それぞれの台詞の話者を特定することで対話コーパスを自動的に構築する手法を提案する。

3.1　概要

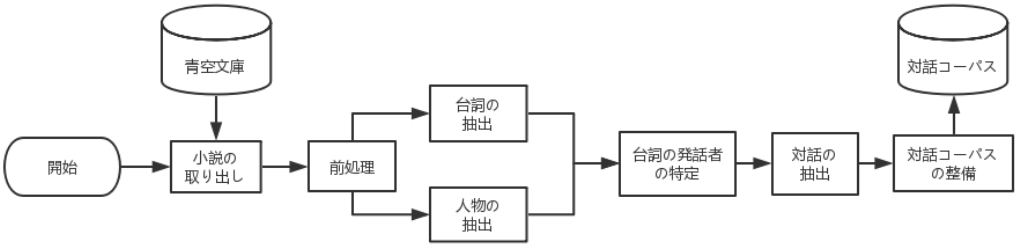

図 3.1: 提案手法の概要

提案手法における処理の流れを図 3.1 に示す。本研究では、対話を抽出する対象とする小説として、青空文庫 [5] で公開されているデジタル化された小説データを用いる。まず、青空文庫から入手した小説のテキストに対し、本研究での処理に適した形式にするために、本文以外のメタデータの削除、文の整形などの前処理を行う。次に、前処理を行った後の小説から台詞を抽出する。さらに、二つの手法により小説から登場人物を抽出する。次に、抽出した個々の台詞について、その話者を小説の登場人物の中から選ぶことで、台詞の話者を特定する。次に、連続している台詞を話者の情報とともに対話として抽出する。最後に、抽出した対話を整備し、最終的な対話コーパスを構築する。

以下、提案手法の詳細について述べる。3.2 節では前処理について述べる。小説からの台詞の抽出については 3.3 節で、登場人物の抽出については 3.4 節で述べる。3.5 節では台詞の話者を特定する手法について述べる。3.6 節では小説から対話を抽出する手法を説明する。最後に、3.7 節では対話コーパスの整備に関する構想について述べる。

3.2　前処理

本節では青空文庫の小説に対する前処理について述べる。

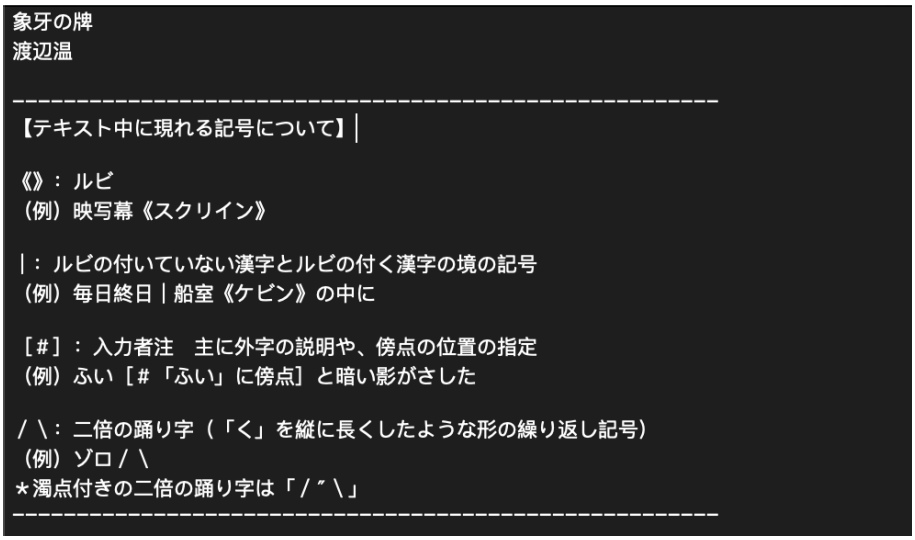

図 3.2: テキストの前のメタデータの例

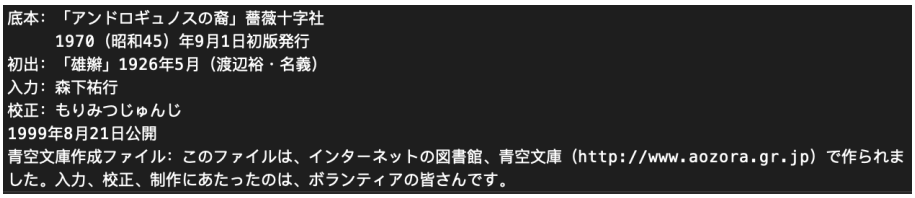

図 3.3: テキストの後ろのメタデータの例

青空文庫に掲載された小説は、テキストの前後にメタデータが付いている。図 3.2 は小説の前のメタデータであり、小説に使われる記号についての説明がある。図 3.3 は小説の後のメタデータであり、底本、初出、テキスト入力者などの情報がある。これらのメタデータは除去する。

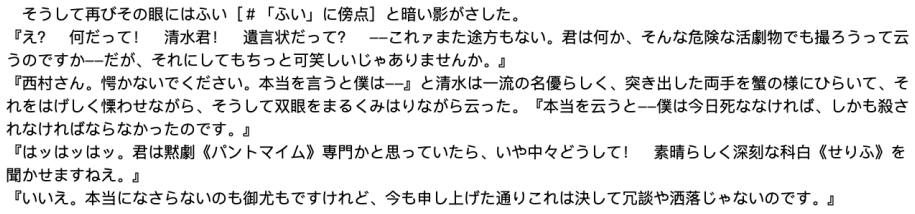
そうして再びその眼にはふい［#「ふい」に傍点］と暗い影がさした。
『え？　何だって！　清水君！　遺言状だって？　――これァまた途方もない。君は何か、そんな危険な活劇物でも撮ろうって云うのですか――だが、それにしてもちっと可笑しいじゃありませんか。』
『西村さん。愕かないでください。本当を言うと僕は――』と清水は一流の名優らしく、突き出した両手を蟹の様にひらいて、それをはげしく慄わせながら、そうして双眼をまるくみはりながら云った。『本当を云うと――僕は今日死ななければ、しかも殺されなければならなかったのです。』
『はッはッはッ。君は黙劇《パントマイム》専門かと思っていたら、いや中々どうして！　素晴らしく深刻な科白《せりふ》を聞かせますねえ。』
『いいえ。本当になさらないのも御尤もですけれど、今も申し上げた通りこれは決して冗談や洒落じゃないのです。』

図 3.4: ルビや注釈の例

また、小説の本文にはルビや注釈などの情報が付与されている。例を図 3.4 に示す。「≪パントマイム≫」や「≪せりふ≫」はルビ、「［＃「ふい」に傍点］」は注

釈である。後続の処理のため、ルビと注釈を削除する。具体的には、表 3.1 に示すルールにしたがい、本文中のルビ・注釈を削除する。

表 3.1: ルビと注釈を削除するルール

1	《》でマークアップされたルビを削除する。
2	《 《》 《》 。》のようなルビ記号と台詞記号が混じっているとき、ルビのみ削除する。
3	― という記号を削除する。
4	［＃］で囲まれた内容と記号を削除する。
5	無駄な空白記号を削除する。

最後に、小説を文に分割し、一行につき一文の形式に変換する。後続の処理では、文節の係り受け解析ツール CaboCha による文の解析を行うが、CaboCha の入力は 1 つの文なので、小説を文に分割する必要がある。具体的には、表 3.2 に示すルールに従って文に分割する。基本的には句点「。」で文を区切る。また、ルール 3 は、ひとつの文が複数の行にまたがっているとき、それを一行に直すルールである。

表 3.2: 文を整形するルール

1	台詞以外の文について、句点「。」の後に改行記号を入れる。
2	2 つ以上の台詞が並ぶとき、台詞の間に改行記号を入れる。
3	「、」の後ろに改行記号があるとき、それを削除する。
4	各行の前に行番号を付ける。

前処理が完了した小説の例を図 3.5 に示す。

```
18 そうして再びその眼にはふいと暗い影がさした。
19 『 え? 何だって! 清水君! 遺言状だって? ―これァまた途方もない。君は何か、そんな危険な活劇物でも撮ろうって云うのですか―だが、それにしてもちっと可笑しいじゃありませんか。 』
20 『 西村さん。愕かないでください。本当を言うと僕は― 』と清水は一流の名優らしく、突き出した両手を蟹の様にひらいて、それをはげしく慄わせながら、そうして双眼をまるくみはりながら云った。
21 『 本当を云うと―僕は今日死ななければ、しかも殺されなければならなかったのです。 』
22 『 はッはッはッ。君は黙劇専門かと思っていたら、いや中々どうして! 素晴らしく深刻な科白を聞かせますねえ。 』
23 『 いいえ。本当になさらないのも御尤もですけれど、今も申し上げた通りこれは決して冗談や洒落じゃないのです。 』
```

図 3.5: 前処理後の小説の例

3.3 台詞の抽出

本節では、小説から台詞を抽出する手法について述べる。青空文庫で掲載された小説は、主に「と」、『と』という 2 種類の括弧で台詞を表す。また――と」、―

表 3.3: 台詞の例

パターン	台詞	タイトル
「.+」	「片手に書物を抱えて片手に銭を要求するのははなはだ高尚でない」と、彼はこの時、初めて彼の夫人に対して不平を洩した。	端午節
『.+』	『僕は活動役者の清水茂です。』と、客は正にそんな風な職業らしい愛想のいい微笑や言葉つきで挨拶した。	象牙の牌
——.+」	——姉さん。どうしたの？」と私は訊ねた。	可哀相な姉
（.+）	（うん。それは行かないでいいだろう。）と須利耶さまは何の気もなくぼんやりと斯うお答えでした。	雁の童子
《.+》	《いいとも、いいとも、確かにおれが引き取ってやろう。しかし一体お前らは、どうしたのだ。》	雁の童子

—と』、（と）、《と》のような開括弧と閉括弧の組で台詞を表す場合もある。青空文庫における台詞の例を表 3.3 に示す。

台詞はパターンマッチによって抽出する。具体的には、表 3.4 に示した 6 つのパターンを用いる。これらのパターンは、前述の 6 種類の開括弧と閉括弧の組で囲まれた文字列を台詞として抽出する。

表 3.4: 台詞を抽出するためのパターン

「.+」	『.+』	（.+）	《.+》	——.+」	——.+』

.+は任意の文字列にマッチすることを表す。

抽出した台詞は以下のようにタグ付けする。

⟨utterance⟩ 台詞 ⟨/utterance⟩

本研究では、登場人物は台詞の話者を特定するために抽出する。一般に、台詞の話者は台詞以外の場所に書かれている。そこで、登場人物を抽出する際には、台詞の中のテキストからは人物を抽出せず、台詞以外の地の文のみから人物を抽出する。

3.4 人物の抽出

本節では、登場人物を抽出する手法を説明する。本研究では、西原と白井の手法 [6] を参考とし、図 3.6 に示す手続きで登場人物を抽出する。すなわち、文ごとに、CaboCha による固有表現抽出とシソーラスによる抽出の 2 通りの方法で人物を抽出する。抽出した人物は以下のようにタグ付けする。

⟨person⟩ 人物 ⟨/person⟩

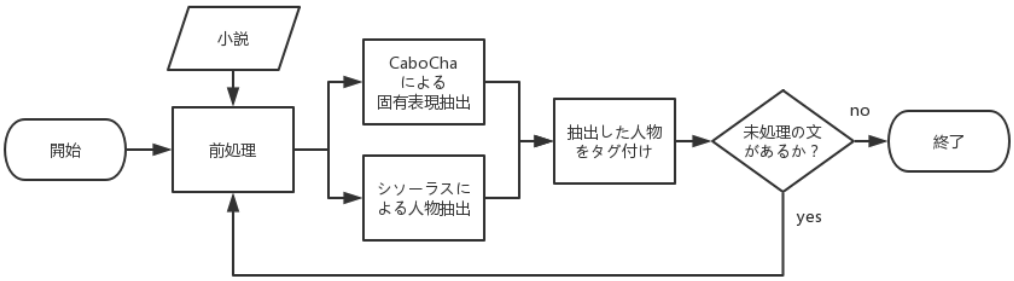

図 3.6: 登場人物の抽出の処理の流れ

3.4.1 固有表現抽出による人物の抽出

固有表現抽出によって「人名」として検出された単語もしくは単語列を登場人物として抽出する。固有表現抽出ツールとして CaboCha を用いる。CaboCha による解析結果を図 3.7 に示す。各行の一番最後にある B-PERSON、I-PERSON、O が固有表現抽出の結果を表す。この出力結果は IOB 形式による。PERSON は「人名」を表す固有表現のクラスであり、B-PERSON は人名の最初の単語を、I-PERSON は人名の 2 番目以降の単語を表す。B-PERSON と I-PERSON が並んだときは、これらの単語を連結したものが人名となる。図 3.7 の例では、「西村」「敬吉」が人名として抽出され、person タグでタグ付けされている。

```
西村敬吉はひどくドギマギとして、彼の前に立った様子のいい陽気な客の顔を眺め返した。
* 0 11D 1/2 -1.555911
西村    名詞,固有名詞,人名,姓,*,*,西村,ニシムラ,ニシムラ        B-PERSON
敬吉    名詞,固有名詞,人名,名,*,*,敬吉,ヨシキチ,ヨシキチ        I-PERSON
は      助詞,係助詞,*,*,*,*,は,ハ,ワ    O
* 1 2D 0/0 0.448681
ひどく  形容詞,自立,*,*,形容詞・アウオ段,連用テ接続,ひどい,ヒドク,ヒドク      O
* 2 11D 0/1 -1.555911
ドギマギ        名詞,一般,*,*,*,*,*     O
として  助詞,格助詞,連語,*,*,*,として,トシテ,トシテ     O
、      記号,読点,*,*,*,*,、,、,、       O
```

⟨person⟩ 西村敬吉 ⟨/person⟩ はひどくドギマギとして、
彼の前に立った様子のいい陽気な客の顔を眺め返した。

図 3.7: 固有表現抽出の例

3.4.2　シソーラスによる人物の抽出

CaboCha による固有表現抽出では、小説における登場人物を完全に抽出できない。図 3.8 の例では、「清水」は人名として認識されていない。そのため、シソーラスを用いて人物を抽出する。

```
清水はきっぱりと云った。
* 0 2D 0/1 -1.407021
清水    名詞,固有名詞,組織,*,*,*,清水,シミズ,シミズ
は      助詞,係助詞,*,*,*,*,は,ハ,ワ
* 1 2D 0/1 -1.407021
きっぱり        副詞,助詞類接続,*,*,*,*,きっぱり,キッパリ,キッパリ
と      助詞,格助詞,引用,*,*,*,と,ト,ト
* 2 -1D 0/1 0.000000
云っ    動詞,自立,*,*,五段・ワ行促音便,連用タ接続,云う,ユッ,ユッ
た      助動詞,*,*,*,特殊・タ,基本形,た,タ,タ
。      記号,句点,*,*,*,*,。,。,。
EOS
```

『ありません。』
⟨person⟩ 清水 ⟨/person⟩ はきっぱりと云った。

図 3.8: シソーラスによる人物抽出の例

CaboCha による形態素解析の結果、品詞が固有名詞と判断された単語を抽出する。図 3.9 は、CaboCha で採用されている IPA 品詞体系における固有名詞の品詞である。単語の品詞が図 3.9 の品詞のいずれかに該当するとき、シソーラスによる人名の判定を行う。

固有名詞	一般		「北穂高岳」、「電通銀座ビル」、「Ｇ１」
	人名	一般	「グッチ裕三」、「紫式部」
		姓	「山田」、「ビスコンティ」
		名	「Ｂ作」、「アントニオ」、「右京太夫」
	組織		「いすゞ自動車」、「ニチレイ」、「統一アイルランド党」
	地域	一般	「北海道」、「やながわ工業団地」、「ラムサール」
		国	「露西亜」、「バングラデシュ」

図 3.9: IPA 品詞体系における固有名詞の品詞

シソーラスとして日本語語彙大系を用いる。以上の手続きにより特定した固有名詞が、日本語語彙大系における「人名」「人」のカテゴリに含まれるとき、それを登場人物として抽出する。図 3.8 の例では、「清水」の品詞は「名詞-固有名詞-組織」であり、日本語語彙大系における「人名」のカテゴリに含まれるため、これを人物として抽出する。

3.5　台詞の発話者の特定

小説から抽出した全ての台詞に対し、その話者を特定する。まず、いくつかの用語を定義する。

本研究では、台詞を埋め込み型台詞と独立型台詞の 2 つの種類に分類する。埋め込み型台詞とは、文の途中に出現する台詞である。一方、独立型台詞とは、それだけで一文を構成する台詞である。表 3.5 にそれぞれの例を示す。

表 3.5: 台詞の種類

埋め込み型台詞	『僕は活動役者の清水茂です。』と、客は正にそんな風な職業らしい愛想のいい微笑や言葉つきで挨拶した。
独立型台詞	『ありません。』 清水はきっぱりと云った。

また、台詞の話者を「明示的な話者」と「暗黙的な話者」の 2 つの種類に分類する。明示的な話者とは、ある台詞を発話した登場人物が小説のテキストの中に明確に記述されている話者を指す。一方、暗黙的な話者は、小説中には存在するが台詞を発っしたことが明示されていない話者である。暗黙的な話者は台詞の近傍に出現することが多い。表 3.6 にそれぞれの例を示す。下線を引いた「清水」が明示的な話者、「方太太」が暗黙的な話者となっている。

表 3.6: 話者の種類

明示的な話者	『ありません。』 清水はきっぱりと云った。
暗黙的な話者	「じゃ、あしたは出入の商人の方はどうしましょう」 方太太は突然押掛けて来て床の前に突立った。

話者を特定する手続きを図 3.10 に示す。まず、明示的な話者を特定する。次に、台詞の種類を判定する。埋め込み型台詞に対して明示的な話者が見つからないときは、その台詞は台詞ではないものとみなす。次に、暗黙的な話者を特定する。最後に、残った台詞に対して、話者交替パターンを繰り返し適用し、その話者を特定する。

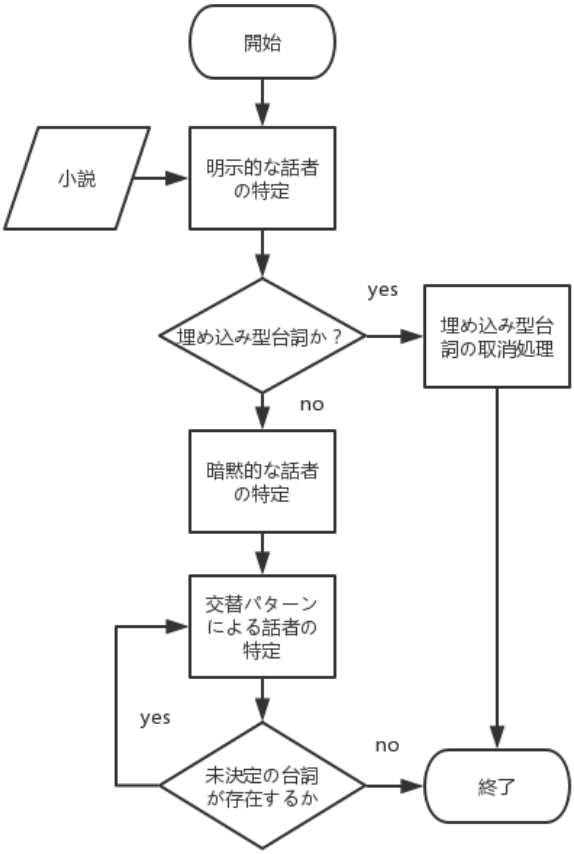

図 3.10: 台詞の話者の特定の流れ

3.5.1 明示的な話者の検出

明示的な話者は、表 3.7 に示す PE_1〜PE_7 の「明示的話者特定パターン」を用いて特定する。これらのパターンにおいて、U は 3.3 節で述べた手法で検出された台詞を、P は 3.4 節で述べた手法で検出された登場人物を表す。J はトピックを表す係助詞である。本研究で使用したトピックを係助詞の一覧を表 3.8 に示す。

一方、SV は発話を提示する動詞 (speech verb) である。インターネット上のシソーラスや類語辞典を参考に、SV に該当する 1,171 個の動詞のリストを作成した。表 3.9 にその一部を示す。また、SV を取得したウェブサイトの一覧を表 3.10 に示す。

明示的話者特定パターンでは、指定するパターンにマッチしたとき、台詞 U の話者を人物 P と特定する。パターン PE_3、PE_4、PE_7 では、台詞 U_1 と U_2 の話者をそれぞれ P_1、P_2 と特定する。

表 3.7: 明示的話者特定パターン

PE_1: $P\ J * U$ と $* SV$
PE_2: U と 、 $P\ J * SV$
PE_3: $P_1\ J * P_2$ に $* U_1$ と $* SV$ U_2
PE_4: U と 、 $P_1\ J * P_2$ に $* SV$ U_2
PE_5: $P\ J * SV$ 。 U
PE_6: U $P\ J * SV$ 。
PE_7: U_1 $P_1\ J * P_2$ に $* SV$ U_2

(P: 人物，J: トピックを表す係助詞，U: 台詞，SV: 発言を表す動詞)

表 3.8: トピックを表す係助詞

「は」「も」「では」「には」「や」「が」

表 3.9: 発言を表す動詞 (抜粋)

ささやく、しゃべりたてる、しゃべる、叫ぶ、 仰しゃる、言う、語る、告白する、説明する

表 3.10: 発言を表す動詞を取得したウェブサイト

サイト名	URL
Weblio	https://thesaurus.weblio.jp/content/言う
連想類語辞典	https://renso-ruigo.com/word/言う
基本動詞ハンドブック	http://verbhandbook.ninjal.ac.jp/headwords/iu/
(個人のブログ)	http://w73t.com/iu/

以下、それぞれのパターンとその適用例を説明する。

- パターン PE_1

文 1: $P\ J * U$ と $* SV$

(適用例)

文 1: 三郎$_P$は$_J$、水を呑んだと見えて、霧をふいて、ごほごほむせて、泣くやうにしながら、「おいらもうやめた。こんな鬼っこもうしない。」$_U$ と云った$_{SV}$。

- パターン PE_2

文 1: U と 、 $P\ J * SV$

(適用例)

文 1: 「片手に書物を抱えて片手に銭を要求するのははなはだ高尚でない」$_U$ と、彼$_P$は$_J$ この時、初めて彼の夫人に対して不平を洩した$_{SV}$。

- パターン PE_3

文 1: $P_1\ J * P_2$ に $* U_1$ と $* SV$
文 2: U_2

(適用例)

文 1: しばらくすると、和尚さん$_{P_1}$は$_J$ 帰って来て、小僧$_{P_2}$ に、「留守にだれも来なかったか。」$_{U_1}$ とたずねました$_{SV}$。

文 2: 「お隣のおばあさんが、お重箱を持って来ました。おひがんだから和尚さんに上げて下さいといいました。」$_{U_2}$

- パターン PE_4

文 1: U_1 と 、 $P_1\ J * P_2$ に $* SV$
文 2: U_2

(適用例)

文 1: 『けれども、まだ三年しか経たないんですものね。』$_{U_1}$ と、なにか大きな、彼女$_{P_1}$には$_J$ わからないけれども、なにか大きな希みを彼$_{P_2}$ に話さ$_{SV}$ なければならないやうに瞳を輝かした。

文 2: 『さうだ、一日々々いろ／＼なことに疲らされなやまされ苦しまされても、二年はもう過ぎたんだからな。もうしばらくすると、坊やも歩るくやうになるんだから。』$_{U_2}$

- パターン PE_5

文 1: $P\ J * SV$
文 2: U

(適用例)
文 1: 方太太$_P$は$_J$ 慌てて語をついだ$_{SV}$。
文 2: 「節句が過ぎて八日になったら、わたしゃ……いっそのこと富籤でも買った方がいいと思いますわ」$_U$

- パターン PE_6

文 1: U
文 2: $P\ J * SV$

(適用例)
文 1: 『ありません。』$_U$
文 2: 清水$_P$は$_J$ きっぱりと云った$_{SV}$。

- パターン PE_7

文 1: U_1
文 2: $P_1\ J * P_2$ に $* SV$
文 3: U_2

(適用例)
文 1: 「すっかり〆め上げると百八十円。この払いが出来ますか」$_{U_1}$
文 2: 彼女$_{P_1}$は$_J$彼$_{P_2}$ に目も呉れずに言った$_{SV}$。
文 3: 「フン、乃公はあすから官吏はやめだ。...中略...金は要らない、役人もやめだ。これほどひどい屈辱はない」$_U$

パターン PE_1〜PE_7 をこの順序で適用し、最初にマッチしたパターンによって話者を特定する。また、同じパターンで複数の登場人物にマッチしたときは、台詞 U との距離が一番近い登場人物を特定する。

3.5.2 埋め込み型台詞の取消処理

明示的話者特定パターンによって話者を特定できない台詞が埋め込み型台詞のとき、それを台詞として検出した処理を取り消し、台詞ではないものとみなす。予備調査の結果、表 3.7 のパターンで話者を検出できない埋め込み型台詞は、括弧で囲まれていても台詞ではない場合がほとんどであったためである。一方、独立型台詞については後続の処理で話者を特定する。

例えば、図 3.11 の場合、「授業をすればお金をやる」は 3.3 節で述べた台詞抽出のパターンにマッチし、台詞として抽出される。この文は明示的話者特定パターンのいずれにもマッチしない。このときは「授業をすればお金をやる」は台詞ではないとみなす。実際、これは政府の声明の内容を表すものであり、台詞ではない。

政府は「授業をすればお金をやる」と声明したが、 この言葉は彼にとっては非常に恨めしかった。

図 3.11: 埋め込み型台詞の取消処理の例

3.5.3 暗黙の話者の検出

暗黙的な話者は、表 3.11 に示す PI_1〜PI_4 の「暗黙的話者特定パターン」を用いて特定する。U、P、J は台詞、登場人物、トピックを表す係助詞である。これらのパターンは、基本的に、台詞の前後の文に出現する登場人物を話者として特定している。ただし、「は」「ては」などトピックを表す係助詞の前に出現する人物を優先して特定する。すなわち、パターン PI_1〜PI_4 をこの順序で適用し、最初にマッチしたパターンによって話者を特定する。また、同じパターンで複数の登場人物にマッチしたときは、台詞 U との距離が一番近い登場人物を特定する。

表 3.11: 暗黙的話者特定パターン

PI_1: $*\ P\ J\ *$ U
PI_2: U $*\ P\ J\ *$
PI_3: $*\ P\ *$ U
PI_4: U $*\ P\ *$

(P: 人物，J: トピックを表す係助詞，U: 台詞)

以下、それぞれのパターンとその適用例を説明する。

- パターン PI_1

文 1: $*\ P\ J\ *$ 文 2: U

(適用例)

文 1: 清水$_P$ は$_J$ 顔色を変えてとび上がった。

文 2: 『違いない！——そ、それを西村さん。あなたは御存知なのですか！……』$_U$

- パターン PI_2

文 1: U
文 2: $*\ P\ J\ *$

(適用例)

文 1: 「じゃ、あしたは出入の商人の方はどうしましょう」$_U$

文 2: 方太太$_P$ は$_J$ 突然押掛けて来て床の前に突立った。

- パターン PI_3

文 1: $*\ P\ *$
文 2: U

(適用例)

文 1: 西村$_P$ の眼には深くあわれみの色が満ちた。

文 2: 『では、お気の毒ながらやっぱり遺言状をお作りしてあげなけりゃなりますまい……僕にはどうも、それ以上、お力になる事は出来ません。相手は象牙菊花倶楽部ですもの。どうしたって——左様、金輪際君の命は助かりませんね。』$_U$

- パターン PI_4

文 1: U
文 2: $*\ P\ *$

(適用例)

文 1: 「全くそうよ、お金なしではお米が買えません、お米なしでは御飯が焚けません……」$_U$

文 2: 彼女$_P$ の両方の頬べたがふかふか動き出した。

3.5.4 話者交替パターン

明示的話者特定パターンと暗黙的話者特定パターンでも話者を特定できない場合は、図 3.12 に示す「話者交替パターン」を用いて話者を特定する。He らの先行研究 [9] でも論じられているように、複数の台詞が連続して出現するとき、その台

表 3.12: 話者交替パターン

PA_1

(話者)	(台詞)
人物 B	U_1
	$*$
人物 A	U_2
?	U_3

? → 人物 B

PA_2

(話者)	(台詞)
?	U_1
人物 A	U_2
	$*$
人物 B	U_3

? → 人物 B

詞の話者は交替することが多い。話者交替パターンはこの性質を利用して話者を特定するものである。

図 3.12 のパターン PA_1 において、(台詞) は小説中の台詞を、(話者) は既に特定された台詞の話者を表す。いま、U_2 の話者は「人物 A」と決まっているが、その次の台詞 U_3 の話者は決まっていない。このとき、U_2 の直前に出現する台詞の話者が「人物 B」と特定されていれば、U_2 から話者が交替すると仮定し、U_3 の話者を「人物 B」と特定する。なお、台詞が連続している場合だけでなく、間に短い文が挿入されているときでも、同様に話者交替のパターンが適用できると考えられる。そのため、U_2 と U_3 は連続した台詞だが、U_1 と U_2 の間には台詞以外の文が存在してもよいものとする。PA_1 における U_1 と U_2 の間の $*$ は任意の文の出現を許すことを表す。つまり、U_1 は U_2 の前に出現する一番近い台詞である。

図 3.12 のパターン PA_2 も同様の考え方で話者を決定する。いま、U_2 の話者は「人物 A」と決まっているが、その前の台詞 U_1 の話者は決まっていない。このとき、U_2 の後に出現する (間に文が挿入されていてもよい) 台詞 U_3 の話者が「人物 B」と特定されていれば、話者交替のパターンから U_1 の話者を「人物 B」と特定する。

話者交替のパターン PA_1 と PA_2 はこの順序で適用する。また、話者交替パターンは、全ての台詞の話者が特定されるまで、あるいは話者交替パターンによって話者を特定できる台詞がなくなるまで、繰り返し適用する。

話者交替パターンによって話者を特定する処理の例を図 3.12 に示す。図 3.12 の 110 行目の台詞の話者は、暗黙的話者特定パターン PI_2 によって「方太太」と特定されている。また、114 行目の台詞の話者も、暗黙的話者特定パターン PI_2 によって「彼」と特定されている。112 行目、113 行目の台詞は明示的な話者特定パターンでも暗黙的話者特定パターンでも特定できなかったため、話者交替パターンを適用する。110 行目の台詞と 113 行目の台詞を同一人物の発話と特定する。また、112 行目の台詞と 114 行目の台詞を同一人物の発話と特定する。すなわち、話者交替パターンによって、112 行目の台詞の話者は「彼」と特定され、113 行目の台詞の話者は「方太太」と特定する。

110:	「じゃ、あしたは出入の商人の方はどうしましょう」
111:	方太太は突然押掛けて来て床の前に突立った。
112:	「商人？……八日の午後来いと言え」
113:	「わたしにはそんなことが言えません。 向うで信用しません、承知しません」
114:	「信用しないことがあるもんか。向うへ行って聞けばわかる。 役所じゅうの人は誰一人貰っていない。皆八日だ」
115:	彼は人差指を伸ばして蚊帳の中の空間に一つの半円を画いた。
方太太：	「じゃ、あしたは出入の商人の方はどうしましょう」
unknown1：	「商人？……八日の午後来いと言え」
unknown2：	「わたしにはそんなことが言えません。 向うで信用しません、承知しません」
彼：	「信用しないことがあるもんか。向うへ行って聞けばわかる。 役所じゅうの人は誰一人貰っていない。皆八日だ」

unknown1 → 彼; unknown2 →方太太

図 3.12: 話者交替パターンの例 (魯迅「端午節」(井上紅梅訳) より)

3.6 対話の抽出

本研究の目的は対話コーパスの自動構築である。そのため、単独で出現する台詞は抽出せず、2 つ以上の台詞が連続して出現するとき、それらを一つの対話として抽出する。

小説中の全ての台詞の話者を特定した後、連続する台詞を対話として抽出する。小説内の対話では台詞の間に地の文が出現することもあるので、台詞間に出現する文の数が 2 以下のときは連続した台詞であるとみなす。対話を抽出する際には、その話者の情報も一緒に抽出する。

図 3.13 は前処理が完了した段階での小説の例である。台詞と登場人物を特定し、台詞の話者を特定し、連続した台詞を抽出すると、図 3.14 のような対話が抽出される。この図の各行は、行番号、話者、台詞の順に並んでいる。また、「*****」という行は対話の境界を表す。80〜86 行目、94〜102 行目の台詞の列が対話として抽出される。一方、91 行目の台詞は他の台詞と並んでいないため、対話として抽出しない。

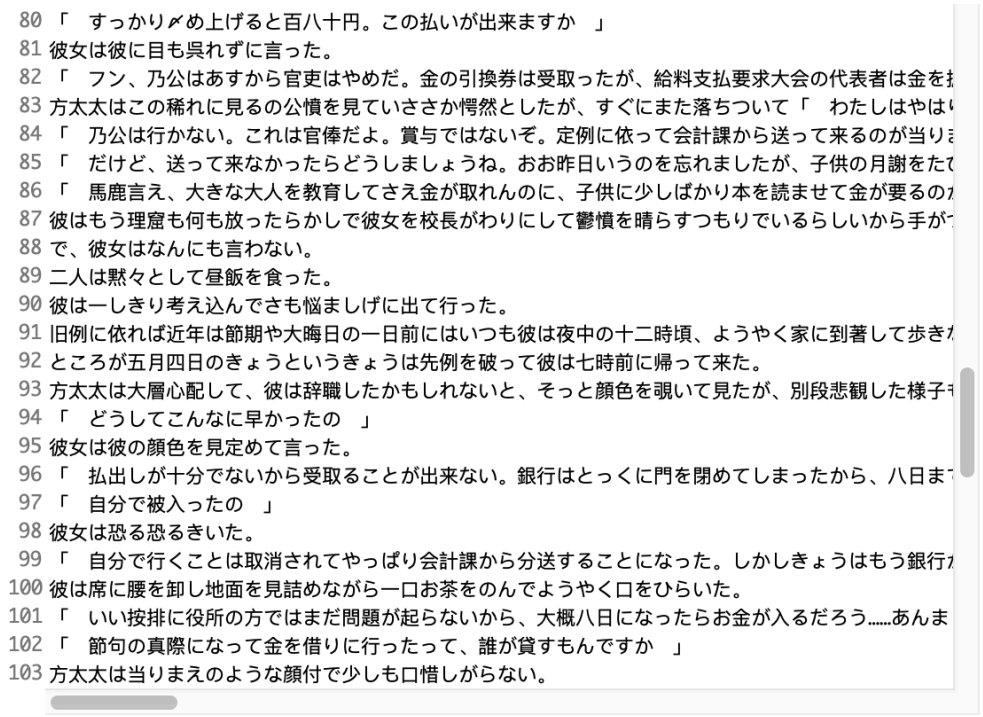

図 3.13: 前処理が完了した小説の例 (魯迅「端午節」(井上紅梅訳) より)

```
80:彼女:「　すっかり〆め上げると百八十円。この払いが出来ますか　」
82:彼:「　フン、乃公はあすから官吏はやめだ。金の引換券は受取ったが、給料支払要求大会の代表者
83:方太太:「　わたしはやはり御自分で取りに被入る方がいいと思います。これじゃしようがありませ
84:彼:「　乃公は行かない。これは官俸だよ。賞与ではないぞ。定例に依って会計課から送って来るの
85:方太太:「　だけど、送って来なかったらどうしましょうね。おお昨日いうのを忘れましたが、子供
86:彼:「　馬鹿言え、大きな大人を教育してさえ金が取れんのに、子供に少しばかり本を読ませて金が
**********************************************************
91:彼:「　おい、取って来たよ　」
**********************************************************
94:彼女:「　どうしてこんなに早かったの　」
96:彼:「　払出しが十分でないから受取ることが出来ない。銀行はとっくに門を閉めてしまったから、
97:彼女:「　自分で被入ったの　」
99:彼:「　自分で行くことは取消されてやっぱり会計課から分送することになった。しかしきょうはも
101:彼:「　いい按排に役所の方ではまだ問題が起らないから、大概八日になったらお金が入るだろう…
102:方太太:「　節句の真際になって金を借りに行ったって、誰が貸すもんですか　」
**********************************************************
```

図 3.14: 対話の抽出例 (魯迅「端午節」(井上紅梅訳) より)

3.7　対話コーパスの整備

本節は、対話コーパスを整備する将来の構想について述べる。ここで述べる処理はまだ実装していない。

上記の手続きで得られた連続した台詞ならびにそれぞれの台詞の話者を整理して、対話コーパスを構築する。話者の情報として小説の人物名をそのまま付与するのではなく、同一人物を「話者 A」のような記号 (話者 ID) に置き換える。

図 3.15 では、「方太太」を「A」、「彼」を「B」に置き換えることにより、話者の情報として ID が付与された対話コーパスを作っている。

方太太：	「じゃ、あしたは出入の商人の方はどうしましょう」
彼：	「商人？……八日の午後来いと言え」
方太太：	「わたしにはそんなことが言えません。向うで信用しません、承知しません」
彼：	「信用しないことがあるもんか。向うへ行って聞けばわかる。役所じゅうの人は誰一人貰っていない。皆八日だ」
	方太太 → A; 彼 → B ↓
A：	「じゃ、あしたは出入の商人の方はどうしましょう」
B：	「商人？……八日の午後来いと言え」
A：	「わたしにはそんなことが言えません。向うで信用しません、承知しません」
B：	「信用しないことがあるもんか。向うへ行って聞けばわかる。役所じゅうの人は誰一人貰っていない。皆八日だ」

図 3.15: 話者の記号への置き換えの例

第4章　実験・評価

本章では、本研究の提案手法の評価実験について述べる。4.1 節では、台詞の話者を特定する手法を評価する。4.2 節では、青空文庫の小説から提案手法を用いて対話コーパスを構築した結果を報告する。

4.1　話者特定手法の評価

3.2 節から 3.5 節では、小説における台詞の話者を特定する手法を提案した。ここではその手法を評価する。人手で正解の話者の情報をタグ付けたテストデータを用意し、正解の話者と自動認識した話者を比較する。

4.1.1　実験データ

本研究では、既に述べたように、対話コーパスを自動構築するために、青空文庫で公開されている小説を用いる。青空文庫で公開されている小説からランダムに 4 つの小説を選択し、テストデータ 1 とする。3.3 節で述べた手法で抽出したそれぞれの台詞に対して、その話者を人手でタグ付けする。テストデータ 1 の概要を表 4.1 に示す。

テストデータ 1 は、提案手法を検討する際に参照した小説である。すなわち、テストデータ 1 の小説を観察し、話者を特定する手法を開発した。したがって、テストデータ 1 を用いた評価はクローズドテストである。

提案手法を公正に評価するために、テストデータ 1 とは別の評価用データを用意する。青空文庫から「不思議な島」「和尚さんと小僧」「可哀相な姉」「父を失う話」の 4 つの小説を選択し、同様に台詞の話者を人手でタグ付けした。これをテストデータ 2 と呼ぶ。テストデータ 2 の概要を表 4.2 に示す。テストデータ 2 を用いた評価はオープンテストである。

4.1.2　評価基準

今回の実験では、適用率と正解率を評価として、台詞の話者の認識手法を評価する。

表 4.1: テストデータ 1 の概要

タイトル	独立型台詞	埋め込み型台詞	合計
晩餐	11	5	16
端午節	31	26	57
象牙の牌	51	10	61
さいかち淵	17	10	27
(全て)	110	51	161

表 4.2: テストデータ 2 の概要

タイトル	独立型台詞	埋め込み型台詞	合計
不思議な島	72	0	72
和尚さんと小僧	4	21	25
可哀相な姉	37	11	48
父を失う話	16	13	29
(全て)	129	45	174

適用率は、小説に含まれる台詞のうち、提案手法によって話者を特定できた台詞の割合である。このとき、特定された話者が正解か不正解は問わない。適用率の定義を式 (4.1) に示す。

$$\text{適用率} = \frac{\text{話者を特定できた台詞の数}}{\text{小説に含まれる台詞の数}} \tag{4.1}$$

一方、正解率は、提案手法によって話者を特定できた台詞のうち、正しく話者を特定できた台詞の割合である。正解率の定義を式 (4.2) に示す。

$$\text{正解率} = \frac{\text{正しく話者を特定できた台詞の数}}{\text{話者を特定できた台詞の数}} \tag{4.2}$$

4.1.3 クローズドテストの結果

表 4.3: 話者特定手法の評価結果 (クローズドテスト)

タイトル	台詞数	適用率	正解率
晩餐	16	1.00	0.88
端午節	57	1.00	0.82
象牙の牌	61	1.00	0.74
さいかち淵	27	1.00	0.37
(全て)	161	1.00	0.72

テストデータ 1 に対する提案手法の評価結果、すなわちクローズドテストの実験結果を表 4.3 に示す。提案手法の適用率は全ての小説で 1 となった。つまり、全ての台詞について話者を特定できた。一方、正解率は、3 つの小説については 70% から 90%となり、比較的高い値となった。一方、「さいかち淵」については 0.37 と低かった。4 つの小説全体での正解率は 0.72 となった。

以下、解析誤りの主な原因について述べる。

> しゅっこは、舜一なんだけれども、みんなはいつでもしゅっこといふ。
> ……
> しゅっこも、大きな白い石をもって、淵の上のさいかちの木にのぼってゐたが、それを見ると、すぐに、石を淵に落して叫んだ。「おゝ、発破だぞ。知らないふりしてろ。石とりやめて、早くみんな、下流へさがれ。」

(『さいかち淵』より抜粋)

図 4.1: 登場人物の抽出に失敗した例

『さいかち淵』に対する正解率が低いのは、登場人物がニックネームで表現されていて、固有表現抽出やシソーラスを用いた手法では人名と認識できなかったためである。図 4.1 は『さいかち淵』の一部である。「しゅっこ」は人物のニックネームで、最後の台詞の話者であるが、登場人物として抽出されなかった。小説のメタ情報として登場人物リストのような情報があれば、登場人物抽出の再現率が上がり、話者を正確に特定できるようになると考えられる。

> 彼女は、すぐに嬉しさうに、『坊や。』と大きな声を出した、子供はそれと同時に大きな叫声を上げて、母親の顔を見ながら、『うま／＼／＼／＼。』とスプーンをテーブルにたゝきつけた。

(『晩餐』より抜粋)

図 4.2: 1 つ文に 2 つの台詞があるときの解析誤り例

1 つの文に 2 つの台詞が存在するとき、話者を特定することができなかった場合があった。図 4.2 は『晩餐』の中の一文である。この文には「坊や。」と「うま／＼／＼／＼。」という 2 つの台詞があり、それぞれの話者は「彼女」と「子供」である。これらの話者は台詞を発っしたことが明示されていると言えるが、本研究で用意した 7 つの明示的話者特定パターンのいずれにもマッチしないため、話者を特定できなかった。この問題に対しては、明示的話者特定パターンを追加することで解決できる可能性がある。

複数の台詞が連続するときに、同じ話者が 2 回連続で台詞を発言するときがあり、このときに誤った話者が特定された。図 4.3 は『象牙の牌』の一部である。67

67:	『　つまり、君の死はもう、思いのほか間近に的確に迫って来ていたと云うことですよ。　』
68:	西村は落ちつきはらった調子で静かにこう云った。
69:	『　？……　』
70:	清水は流石に狼狽してあたりを見まわした。
71:	『　その証拠は――　』
72:	西村はそう云いながら、立って部屋の一隅に置かれた典雅な書棚の抽斗を開けて、しばらくゴソゴソやっていたが、軈て、ひとふりの抜き身の支那型の短剣を取り出して来た。
73:	『　これですよ……　』
74:	『　おお!!　』
75:	清水は突き出されたその短剣のつかに目をやると、うめいた。
西村:	『　つまり、君の死はもう、思いのほか間近に的確に迫って来ていたと云うことですよ。　』
清水:	『　？……　』
西村:	『　その証拠は――　』
unknown:	『　これですよ……　』
清水:	『　おお!!　』

(『象牙の牌』より抜粋)

図 4.3: 話者交替パターンによる解析誤り例

行目の台詞の話者は明示的話者特定パターンによって「西村」と特定された。また 69 行目の台詞は暗黙的話者特定パターンによって「清水」と特定された。71 行目の台詞の話者は明示的話者特定パターンによって「西村」と特定された。74 行目の台詞の話者は明示的話者特定パターンによって「清水」と特定された。一方、73 行目の台詞の話者はパターンマッチでは特定できなかったため、話者交替パターンを用いて特定を試みる。2 つ前の台詞の話者が「清水」なので、話者交替パターンでは 73 行目の話者は「清水」と特定された。しかし、実際には 71 行目と 73 行目の台詞は同じ話者が 2 回続けて発言しており、73 行目の正しい話者は「西村」である。この場合、連続する発話の話者は常に交替するという原則にしたがっていないため、話者交替パターンによって誤った話者が特定された。

話者交替パターンでは、話者が台詞毎に必ず交替することを仮定していたが、図 4.3 の例のように例外的にそうでない場合があるので、対処が必要である。同じような解析誤りは『端午節』でも見つかった。また、現在の話者交替パターンは対話の参加者が 2 名であることを仮定しているため、3 名以上の人物が対話している場面では正しい話者を認識できない。

4.1.4 オープンテストの結果

テストデータ 2 に対する提案手法の評価結果、すなわちオープンテストの実験結果を表 4.4 に示す。

表 4.4: オープンテストの評価結果

タイトル	台詞数	適用率	正解率
不思議な島	72	1.00	0.86
和尚さんと小僧	25	1.00	0.72
可哀相な姉	48	1.00	0.52
父を失う話	29	1.00	0.76
(全て)	174	1.00	0.72

適用率は全ての小説で 1 となった。つまり、テストデータ 2 に対しても、提案手法により全ての台詞の話者を特定できた。一方、正解率は、『可哀相な姉』に対する正解率は 0.52 に留まったものの、他の 3 つの小説についての正解率は 70%以上と比較的高い値となった。全体の正解率は 0.72 となり、これはクローズドテストであるテストデータ 1 に対する正解率と変わらない。したがって、本研究で提案する台詞の話者を特定する手法は、手法の開発に用いたデータに特化せず、ある程度汎用的であることが確認された。

正解率が一番低い『可哀相な姉』における誤りの例を図 4.4 に示す。116 行目の台詞の話者は 117 行目の文に出現する「私」であるが、本研究の明示的話者特定パターンでは「姉」を話者と誤って認識した。また、正解の話者「私」を話者として特定できるパターンはない。さらに、話者交替パターンによって 119、121、123 行目の台詞の話者を特定するときに、116 行目の話者の解析誤りが伝播した。

また、『和尚さんと小僧』には、『晩餐』と同じように、1 つの文に 2 つの台詞が存在する場合があり、このときに話者を特定することができなかった。

『父を失う話』と『不思議な島』については、正解率が高かったが、暗黙的話者特定パターンと話者交替パターンを使用する順序について問題が見つかった。図 4.5 の例では、明示的話者特定パターンによって、76 行目の台詞の話者は「役人」、77 行目の台詞の話者は「私」と特定できる。もしこの時点で話者交替パターンを用いれば、73 行目の台詞の話者は、2 つ後の台詞の話者が「役人」であることから、「役人」と正しく特定される。しかし、提案手法では話者交替パターンよりも暗黙的話者特定パターンを先に用いる。したがって、73 行目の話者は暗黙的話者特定パターンによって「私」と誤って特定された (図 4.5 における †)。暗黙的話者特定パターンは台詞の周辺に出現する登場人物を話者として特定するという制約の緩いルールなので、誤りも多い。場合によっては話者交替パターンの方が信頼性が高いことも考えられる。この問題に対して、台詞間に出現する文がない連続した台詞に対しては、暗黙的話者特定パターンよりも話者交替パターンを優先して利用することで解決できる可能がある。

115: ―― 可哀相な姉よ！
116: ―― 姉さん、どうしたのです？」
117: 姉は、さも憎々しげに私を睨みつけながらうなずいていた。
118: ―― オマエ、ヒゲヲ、ハヤス、ツモリカエ？」
119: ―― だって、僕はもう大人になったのですから生やしたいのです。」
120: ―― オトナハ、ワタシ、キライダ！」
121: ―― そんなことを云ったって、無理ですよ。僕は大人になって、姉さんを広い家に住まわせて、仕合せにして上げようと思うのです。」
122: ―― イイヨ。カッテニ、スルガイイ。ワタシハ、アノクスリヲノムカラ！」
123: ―― 薬ですって？」
124: 姉は首を横に振って、机の上の黒い本を開いて見せた。

(『可哀相な姉』より抜粋)

図 4.4: 『可哀相な姉』の解析誤り例

70:	到頭金釦をつけた空色の制服を着ている税関の役人が私の肩を敲いた。
71:	「　どうしたんです？まさか、身投げをするつもりじゃないでしょうね。　」
72:	私は急に悲しくなってむせび泣いた。
73:	「　おやおや、困りますね、一体どうしたって云うのでしょう。泣いてちゃわかりません。わけをお話しなさい。　」
74:	「　お父さんが、いなく、なった、のです！……　」と私はようやく答えた。
75:	そして、それから、父のためにどんな風にしてあざむかれてしまったかを語った。
76:	「　お父さんはどんな様子の人です？　」と役人はきいた。
77:	「　よく思い出せないのです。そう、恰度あなたみたいな人です。髭がなくなってつるつるした顔をしていました。そして、しかもやっぱりそんな大きな眼鏡をかけていました。ああ、ほんとにあなたとそっくりです！　」と私は叫んだ。
役人:	「　どうしたんです？まさか、身投げをするつもりじゃないでしょうね。　」
私:†	「　おやおや、困りますね、一体どうしたって云うのでしょう。泣いてちゃわかりません。わけをお話しなさい。　」
私:	「　お父さんが、いなく、なった、のです！……　」
役人:	「　お父さんはどんな様子の人です？　」
私:	「　よく思い出せないのです。そう、恰度あなたみたいな人です。髭がなくなってつるつるした顔をしていました。そして、しかもやっぱりそんな大きな眼鏡をかけていました。ああ、ほんとにあなたとそっくりです！　」

(『父を失う話』より抜粋)

図 4.5: パターンの適用順序で誤りが生じた例

4.2 対話コーパス構築

青空文庫に掲載されている 4,836 件の全ての小説に対して、提案手法によって対話を抽出した。結果を表 4.5 に示す。

表 4.5: 対話コーパス構築の実験結果

小説数	4,838
うち台詞を検出できた小説数	3,267
うち対話を抽出できた小説数	2,209
対話数	19,492
対話における発話 (台詞) の総数	201,391
一対話当たりの平均発話数	10.3

話者特定の適用率は 0.917 であった。つまり、提案手法によって台詞の話者を特定できない台詞は 1 割未満である。表 4.5 中の「対話数」は、連続して出現しかつ全ての台詞の話者を特定できた対話の数である。平均して 10 個程度の発話からなる 19,000 件の対話を抽出し、比較的大規模な対話コーパスを自動構築できた。

A	男	『なぜ黙ってるの。』
B	彼女	『まだ本当にわづかしか経ちませんのね、結婚してから。』
A	男	『さうだなァ。』
A	男	『たった三年にしかならないんだな。けれども、俺たちはいろ／＼苦労したなァ。』
B	彼女	『本当にね。』
A	男	『さうだ、一日々々いろ／＼なことに疲らされなやまされ苦しまされても、二年はもう過ぎたんだからな。』
B	彼女	『坊や。』
B	彼女†	『うま／＼／＼／＼。』
A	男	『もう少しの間だ。』
B	彼女	『さうね、私たちは働きませうね。』
A	男	『さ、あとを片づけよう。そして寝よう、明日は早く起きようぢゃないか。』

表 4.6: 抽出された対話の例

図 4.6 は実際に抽出された対話の例である。各行は (今後付与する予定の) 話者 ID、小説から特定された話者、台詞を示している。なお、† のついた話者は解析誤りで、正しくは「子供」である。

第5章　おわりに

5.1　本研究のまとめ

本論文では、小説から対話コーパスを自動構築する手法を提案した。まず、小説から抽出した台詞に対して、その話者を特定する手法を提案した。また、連続する台詞を話者の情報とともに対話として抽出する手法を提案した。

小説における台詞に対して、その発話者を特定するために、まず、「明示的な話者」と「暗黙的な話者」を対象として、これらを特定する一般的なパターンを設定した。また、台詞が連続したときは二人の話者が交互に交替することが多いことに着目し、話者交替パターンを提案し、それを用いて連続的に話者を特定した。

評価実験では、提案手法の適用率は、クローズドテスト、オープンテストともに 1.0 になった。つまり、提案手法によって全ての台詞を特定できた。一方、話者特定の正解率は、小説によってばらつきが見られたものの、クローズドテスト、オープンテストともに 0.72 となった。

青空文庫に掲載されている小説に対して、提案手法によって台詞の話者を特定し、連続する台詞を抽出することで、19,492 件の対話を抽出した。提案手法のアプローチによって比較的大規模な対話コーパスを自動構築できることを確認した。

5.2　今後の課題

今後の課題について述べる。まず、対話コーパスの整備のために、小説中の登場人物名を話者 ID へ置き換える処理を実装することが挙げられる。この際、代名詞が抽出されたとき、その先行詞を特定し、同一人物に対して同じ話者 ID を与える工夫が必要である。

また、本論文では、小説から対話を抽出する時には、小説内の対話では台詞の間に地の文が出現することもあるため、台詞間に出現する文の数が 2 以下のときは連続した台詞であるとみなしていた。しかし、2 という閾値は必ずしも最適とは言えない。ここでの閾値の設定は、言い換えれば小説における対話の境界を推定することに相当するが、その手法は再検討すべきである。

最後に、4.1 節の実験では、オープンテストでもクローズテストでも青空文庫の小説をテストデータとした。一方、ウェブ上でアクセスできる小説は青空文庫以外にも多々ある。小説の情報源によって小説のフォーマットやスタイルが異なる

ことも考えられる。そのため、青空文庫に掲載された小説とは別の情報源から取得した新しいテストデータを用いて提案手法を評価する必要がある。提案手法が青空文庫の小説に限らずどんな小説にも適用可能な汎用的な手法となっているかを検証する。

参考文献

[1] 畑健治, 小倉卓也, 萩原将文. 言語資源を用いた非タスク指向型対話システム. 日本感性工学会論文誌, Vol.10,no.4,pp.515-522. 2011.

[2] 小林峻也, 萩原将文. ユーザの嗜好や人間関係を考慮する非タスク指向型対話システム. 人工知能学会論文誌, Vol.31,no.1,SP2-A,pp.DSF-A-1-10. 2016.

[3] 狩野芳伸. 対話システムの現在. 情報管理, 2017,vol.59,no.10,pp.658-665. 2017.

[4] 田中弥生, 柏野和佳子, 角田ゆかり, 伝康晴, 小磯花絵. 『日本語日常会話コーパス』の構築ー会話収録法に着目してー. 国立国語研究所論集 (NINJAL Research Papers) 14:pp.275-292. 2018.

[5] 野口英司 (編). インターネット図書館青空文庫, はる書房. 2005.

[6] 西原弘真, 白井清昭. 物語テキストを対象とした登場人物の関係抽出. 言語処理学会第 21 回年次大会, pp.626-631. 2015.

[7] http://taku910.github.io/cabocha/

[8] 池原悟, 宮崎正弘, 白井諭, 横尾昭男, 中岩浩巳, 小倉健太郎, 大山芳史, 林良彦. 日本語語彙大系 CD-ROM 版. 岩波書店. 1999.

[9] Hua He, Denilson Barbosa, and Grzegorz Kondrak. Identification of speakers in novels. In Proceedings of Annual Meeting of the Association for Computational Linguistics, pp. 1312-1320, 2013.

[10] 小林聡. 場・時・人に着目した物語のシーン分割手法. 情報処理学会研究報告, Vol.2007-NL-179, No.47, pp.25-30. 2007.

[11] David K. Elson and Kathleen R. McKeown. Automatic Attribution of Quoted Speech in Literary Narrative. Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10), pp.1013-1019. 2010.