箇条書きを含む文書に対する構文解析

松崎拓也[a]　小林義行[b]　藤尾正和[b]　待井君吉[c]　川端薫[c]

a.　東京大学大学院 情報理工学系研究科 コンピュータ科学専攻

b.　株式会社 日立製作所 中央研究所　　c.　株式会社 日立製作所 日立研究所

matuzaki@is.s.u-tokyo.ac.jp, {yoshiyuki.kobayashi.gp, masakazu.fujio.kz, kimiyoshi.machii.td, kaoru.kawabata.mn}@hitachi.com

1.　はじめに

仕様書、論文、報告書などの技術文書には通常のテキストに加え、表や箇条書きなどレイアウト自体が意味を持つ文書要素が頻出し、特に重要な情報がこれらのレイアウト付き文書要素で記述されることも多い。本稿では、特に箇条書きに着目して、実際の技術文書に現れる箇条書きの用法について分析を行い、さらに、箇条書きが埋め込まれた文に対する構文解析技法を提案する。

以下、第 2 節でレイアウト解析によるテキストの抽出、第 3 節で技術文書における箇条書きの特徴について述べる。第 4 節では箇条書きの実例についての分析結果を示し、第 5 節で箇条書きを含む文に対する構文解析手法を提案する。第 6 節で実験結果を示し、第7節でまとめと考察を述べる。

2.　レイアウト解析によるテキストの抽出

PDF 等の実文書からテキストを抽出し、言語処理を行うには、カラム・段組み構造、ヘッダ・フッタ領域、章立て、図・表領域といったレイアウト情報を抽出し、文や箇条書きテキストのブロックを読み順で抽出する必要がある。この処理は、大まかには以下のプロセスからなる。1) 文字コード・配置情報の取得、2) ヘッダ・フッタ領域判定、3) ブロック構造抽出、4) 文字行抽出。図 1 の例では、外枠の表から、ヘッダ・フッタ部分を除き、本文が含まれる枠を抽出し、文と箇条書きからなる本文領域を抽出している。

3.　技術文書における箇条書き

箇条書きは、関連する項目を垂直に並べて書く、レイアウト上の工夫のひとつである。各項目の先頭に、数字や「・」などの記号をおくこともある。

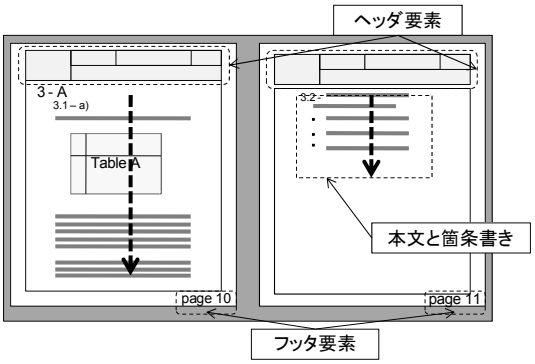

図 1　レイアウト解析例

箇条書きを使うことで、仕様項目や仕様値、構成部材など技術文書において重要な情報を目立たせることができる。本論文では、WWW から収集した 8 件の技術文書（仕様書、サーベイ、論文）から収集した 111 箇所の箇条書きを分析した。

技術文献における箇条書きの特徴のひとつは、ほとんどの場合、箇条書きが現れる前に、箇条書きを導入する文（以下、先行文）があることである。先行文は、箇条書きの文章中での役割を示すために必要な文と考えられる。

以下の例 1 では、先行文の最後の部分（前置詞 of の目的語）が欠落することで箇条書きが導入されており、列挙された各項目は The NLP system と consist of の関係にあるモノであることを表している。

（例 1）欠落型の箇条書き

The NLP system consists of:

・Sentence splitter

・POS tagger

・Syntactic analyzer

以下では、この書き方を欠落型と呼ぶ。上の例では、前置詞の目的語が欠落しているが、他動詞の目的語が欠落している場合や、I think that: の後に箇条書きがある場合のように、that 節内の埋め込み文が欠落している場合などがある。

また、先行文の書き方には、“the following” など、箇条書きで列挙される項目を参照する表現を含む以下のような例もある。

（例２）参照型の箇条書き

The NLP system consists of the following:

・Sentence splitter
・POS tagger
・Syntactic analyzer

こちらの書き方（以下、参照型と呼ぶ）では、参照表現が先行文中で果たす文法的・意味的な役割を認識することで、文書において箇条書き項目に対して述べられている情報を理解できる。参照表現としては “the following programs” のように following に名詞を付与することで、箇条書き項目のあいだの性質が共通であることをより強く示す書き方もある。

箇条書きの特徴のもうひとつは、各箇条書き項目は、文法的・意味的な特徴が類似しており、等位接続詞を用いて書き換えても意味が変わらない場合が多いことである。例えば先の例１であげた文章は、“The NLP system consists of a sentence splitter, a POS tagger and a syntactic analyzer.” のように、適宜冠詞を補った上で等位接続詞 “and” で結びつけた形式に書き換えても意味が変わらない。代表的な英文スタイル・ガイドの１つである The Chicago Manual of Style [1] でも、箇条書きの各項目が文法的に類似していることを一般的な原則としている。

なお、ここまで、箇条書き項目が名詞句である例のみを示したが、動詞句や文、あるいは複数文からなる文章がひとつの項目となる場合もある。項目が複数文からなる場合、上記のように等位接続構造へ書き換えることはできないが、適切な接続表現を各項目の先頭に補うことで、先行文と箇条書き部分全体をひと続きの文章として書き替えることはほぼ可能だろう。

4. 現実の文書における箇条書きの分析

WWW から収集した 111 箇所の箇条書きについて分析した結果を以下に示す（収集した箇条書きは 80 箇所だが、その中に入れ子の箇条書きがあるため、合計 111 箇所）。箇条書き項目の文法カテゴリを集計した結果を表 1 に、先行文の特徴について分類した結果を表 2 に、文法的性質が等しいか否かを分類した結果を表 3 に示す。

表１　箇条書き項目の文法カテゴリ

項目の文法カテゴリ	箇条書き箇所数
名詞句	62
文・文章	22
名詞句＋文・文章	18
名詞句と文・文章が混在	9
合計	111

(注)「名詞句＋文・文章」は箇条書き項目の先頭に名詞句（タイトルのような役割を果たす）があり、この名詞句の説明が続く形式

表 2　先行文の分類

先行文の書き方	箇条書き箇所数
参照型	59
欠落型	33
上記以外	5
無し	14
合計	111

表 3　箇条書き項目の関係

等位接続詞で結ぶ	箇条書き箇所数
可能	106
不可能	5
合計	111

箇条書き項目が名詞句だけであれば、箇条書きから属性と属性値を抽出する手法（例えば吉田の方法[2]など）が適用できるが、表１から、実際には文や文章が項目となっていることがあり、単純な属性と属性値の抽出では箇条書きの意味の解析として不十分であることが分かる。また、表２に見られるように、先行文を持たない箇条書きは少数であり、先行文の分類としては、例１、例２として示した参照型と欠落型で全体の 95%を占めていた。最後に、表３に見られるように、箇条書きの各項目を等位接続詞で結ぶことができない事例は少数であり、それら 5 つの事例は、すべて、先行文の書き方が参照型、欠落型のいずれにも該当しないものであった。

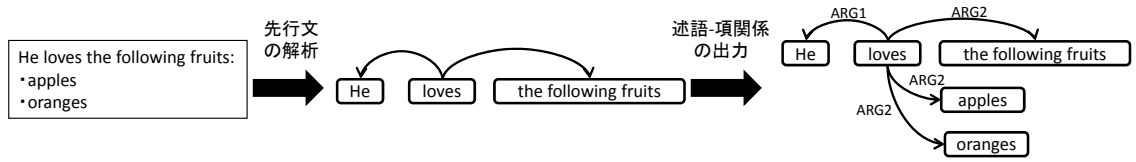

図２　参照型の先行文をもつ箇条書きの解析

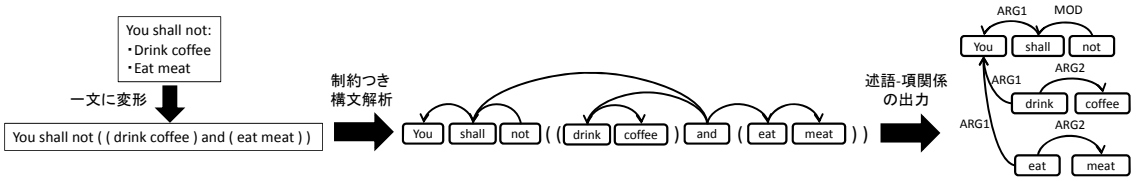

図３　欠落型の先行文をもつ箇条書きの解析（図中の括弧は句境界の制約を表す）

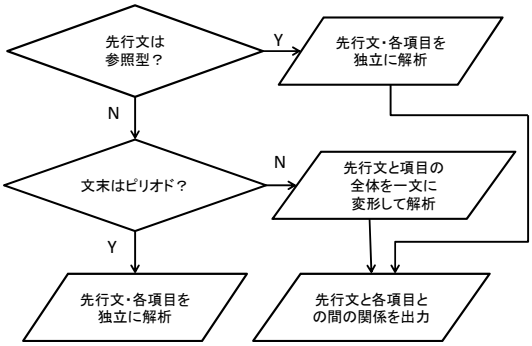

図４　解析アルゴリズム

5.　先行文と箇条書きの構文解析

先行文と箇条書き項目の意味的関係を認識するための構文解析手法を提案する。入力としては、HTML における<ul>や<li>のようなタグで箇条書き全体と各項目の範囲がタグ付けされた文書を仮定し、さらに、文と文の境界も認識済みとする。本手法は単一文を入力とする構文解析器をサブルーチン的に利用し、欠落型の箇条書きに対しては先行文と箇条書き項目を一文に書き替え、参照型の箇条書きに対しては先行文中での参照表現の意味役割を解析することで箇条書き項目の文章中での役割を認識する。アルゴリズムの概要を図４に示す。構文解析器としては Enju HPSG Parser [3] を利用したが、係り受け解析など、他の構文解析枠組みを利用することもできる。

解析アルゴリズムの詳細は、以下の通り：

(1)　先行文のタイプの同定

まず、先行文が /following */、/as follows/、/(listed|shown) below/ などの参照表現パターンにマッチするか調べる。パターンにマッチした場合は参照型の先行文、マッチせず、文末がピリオド等（“.”、“?” または “!”）でない場合は欠落型の先行文と判断する。それ以外の場合は箇条書き項目と先行文の関係は不明とし、先行文および各項目を独立に構文解析する。

(2)　先行文が参照型の場合

先行文を構文解析し、参照表現が関与する述語-項関係を、箇条書きの各項目（複数文を含む場合は、その最初の文）についても出力する（図２）。

(3)　先行文が欠落型の場合

各箇条書き項目（の最初の文）S1、S2、...、Sn を等位接続詞 “and” で結合し、先行文 T に付加する。このとき、各項目 Si の末尾にもともと接続表現（/and|or|as well as/）がある場合は、挟みこんだ “and” を適宜省く。このようにして形成した文 “T S1 and S2 and ... and Sn” をそのまま構文解析した場合、箇条書きの構造から明らかな句境界と合致しない解析が出力される場合がある。例えば以下のような箇条書き

You shall not:
・Drink coffee
・Eat meat

に対し、(You shall not drink coffee) and (eat meat) [コーヒーはよせ、肉を喰え] は誤った解釈である。このような解析誤りを防ぐため、各項目 Si が句となること、および、項目全体を結合した “S1 and S2 ... and Sn” 全体が句となることの２つを構文解析時に制約として用いる。解析の結果として、各項目 Si について先行文 T との間の述語-項関係が得られる（図３）。

6.　実験結果

第 4 節で分析した箇条書きと先行文の実例に対して提案手法を適用し、その結果を調べた。なお、

提案手法の開発は第 4 節と類似の分析を基にして行ったが、手法開発のための分析に利用したデータはこの実験で用いたものとは別のものである。

全 111 の箇条書きの解析中、構文解析器がいかなる解析も出力しない例が 6 例あった。このうち 5 例は、パターンがカバーしていない参照表現を含む参照型の先行文や、箇条書きの見出しである名詞句が、誤って欠落型の先行文として処理されたため、先行文と箇条書きを一文に変形した結果が非文あるいは不自然な文となったためである。特に、列挙される項目のクラス等を表す名詞句（例えば 「カレーの材料」）が箇条書きの見出しとなっている場合、現在のアルゴリズムでは見出し部分が欠落型の先行文として処理されるため、構文解析ができた場合も誤った結果が出力される。表 2 で先行文「無し」に分類された 14 例は、全てこのような見出しを持つ箇条書きであった。これらに対しては、先行文に定動詞が含まれない場合は見出しであると判断する等の改善が可能である。

参照型の先行文を持つ箇条書き 59 例のうち、先行文中で参照表現が関与する述語-項関係が正しく認識できたのは 43 例であった。これらに対しては、各箇条書き項目と先行文との関係が認識できたといえる。失敗例のうち最も多かったのは、パターンがカバーしていない参照表現を含む場合（9 例）であり、このうち半数以上が、定冠詞を持たず、数を明示した複数形名詞句（例：“we used three data sets:”）によって、列挙される項目全体を参照するものであった。次に多かったのは契約書などに頻出する “X includes, but is not limited to, Y” という表現を含む例で（5 例）、これはこの構文（right-node raising）が、解析器で用いる文法でカバーされていないためである。

欠落型の先行文を持つ箇条書き 33 例のうち、29 例について箇条書き項目と先行文の述語-項関係が正しく認識できた。失敗した 4 例のうち 3 例は、現在の実装では対応していない、入れ子になった箇条書きが正しく解析できなかったためである。

7. おわりに

本稿では技術文書に現れる箇条書きに着目して、実データに対する分析を示し、箇条書きを含む文に対する構文解析技法を提案した。データの分析から、ほぼ全ての箇条書きは先行文ないし見出しをもつこと、また、先行文は箇条書き項目に対する参照表現を含むものと、先行文と箇条書き項目を併せて一文として解釈できるものに大別できることが分かった。また、実験結果から、先行文を持つ箇条書きの大多数は、先行文中に参照表現が含まれるかどうかを正しく認識できれば、各項目と先行文との意味関係がほぼ正しく解析できることが分かった。

本稿で提案した解析技法に対する今後の改良としては、先行文ではなく見出しを持つ箇条書きの扱いや、入れ子になった箇条書きへの対応、先行文中の参照表現の認識率の向上が挙げられる。特に、先行文中の参照表現の認識は、共参照解析問題の特殊なケースであるが、後方で明示的に列挙されている要素を指す参照表現を見つけるという興味深い部分問題である。

提案法の応用としては、技術関連の契約書を解析することにより、潜在的に契約上のトラブルとなりうる要注意箇所を自動的に発見するシステムを想定している。待井ら[4]が示すように、単純なキーフレーズ検出による方法では、箇条書きなどレイアウトをもつ文書要素内の要注意箇所を発見することは難しい。提案法のように箇条書き周辺の文と箇条書き項目の意味的関係を解析することで、それら発見の難しかった要注意箇所の検出が容易になることが期待できる。

謝辞

本研究の前身となる研究において多大な貢献を頂いた、辻井潤一氏（Microsoft Research Asia）および原忠義氏（国立情報学研究所）に感謝いたします。

参考文献

[1] Chicago Manual of Style (15th ed.). 2003. University of Chicago Press

[2] 吉田稔. 2002. “表形式と箇条書き形式からの情報抽出”, 東京大学博士論文.

[3] Takashi Ninomiya, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2007. ”A log-linear model with an n-gram reference distribution for accurate HPSG parsing”, IWPT-2007

[4] 待井君吉，横田毅，尾花充，後藤仁一郎. 2010. “英文契約書評価支援システムの開発”, 情報処理学会第 197 回自然言語処理研究発表会（NL197-1）.