

テストケース作成自動化のための意味役割付与方法

増田 聡 松尾谷 徹 津田 和彦

ソフトウェアの仕様書における条件と動作の記述からテストケースを作成する技法に、デシジョンテーブルテスト技法がある。この技法に着目し、テストケース作成自動化のための条件と動作の意味役割付与方法を提案すると共に実験評価を行った。提案方法は、自然言語処理を利用し、前処理で構文的類似度比較により仕様書を対象に振り分け、次に対象文の係り受け関係と格解析の結果から条件と動作を抽出する方法である。記述スタイルの違う文書に対して実験評価を行い、Precision が 0.901 から 0.988、Recall が 0.946 から 0.974 となった。また、処理時間も手作業と比べ約 6 分の 1 となり、提案方法の有効性を確認した。

Decision table testing is a technique to develop test cases from descriptions of conditions and actions in software specification documents. We propose, experiment and evaluate a semantic role labeling technique of conditions and actions for automatic software test cases generation. Our approach uses natural language processing to select sentences from the specification based on syntactic similarity, and then to determine conditions and actions through dependency and case analysis. We got experiment results that precision reached from 0.901 to 0.988, recall reached from 0.946 to 0.974 for different style of descriptions, and the workload was reduced to one-sixth of manual work. Our results on case studies show the effectiveness of our technique.

1 はじめに

企業活動に使用される、いわゆるエンタープライズシステムにおいてテストは重要である。システムテストやユーザ受け入れテストにおいては、テストケースは設計書や要求仕様書から作成される。設計書などステークホルダーで共有される文書の多くは、自然言語で記述されている。そのため、多くのテスト技法 [3] におけるテストケースの作成では、テスト対象の文章に対する理解力を含む、作成者のスキルに依存している。このため、必要なテストケースの抜け漏

れの問題が生じている。この問題の解決方法のひとつとして、仕様書を形式言語で記述する試みもある。しかし、形式言語の書式構築や理解することなどが困難であり、汎用的な手法にはなっていない。人のスキルに依存せず、抜け漏れの無いテストケースを作成するためには、仕様書を機械的に分析しテストケースを自動的に作成する方法が考えられる。

本論文では、自然言語処理技術を利用して仕様書の分析を行い、デシジョンテーブルテスト技法 [3] における“条件”と“動作”という意味的役割を付与する方法を提案し、実験評価をする。前処理として、仕様書の文から、条件と動作を含む文の雛形文体知識を構築した。この雛形文体知識と、テキストの構文的類似度 [13] を用い、仕様書の文を比較することにより、適切な文体に振り分ける。また、自然言語で記述されている文書を対象とするため、文の記述スタイルの違いが結果に影響することが予想される。この課題についても、記述スタイルの違う文書を用いた実験を行い、評価する。これにより、条件と動作の抽出精度の向上

Semantic Role Labeling for Automatic Software Test Cases Generation.

Satoshi Masuda, 日本アイ・ピー・エム株式会社, IBM Research - Tokyo, IBM Japan.

Tohru Matsuodani, 法政大学, Hosei University.

Kazuhiko Tsuda, 筑波大学, Tsukuba University.

コンピュータソフトウェア, Vol.34, No.2 (2017), pp.16–27. [研究論文] 2016 年 2 月 8 日受付.

本論文は第 22 回ソフトウェア工学の基礎ワークショップ (FOSE2015) の発表論文を発展させたものである。

に取り組む。

2 デシジョンテーブルテスト技法への 自然言語処理の適用と課題

2.1 デシジョンテーブルテスト技法

デシジョンテーブルは条件と動作を構成要素とする。この条件と動作というテストの深層格を、自然言語処理による文の係り受け情報と表層格の情報から同定することが本論文の提案である。デシジョンテーブルテスト技法とは、テスト対象とされた要件や機能などの仕様[2]から条件と動作を抽出し、デシジョンテーブルにマッピングし、条件と動作の組合せを挙げていくことによりテストケースを作成する技法である[3]。このテストケース作成の条件と動作の抽出において、仕様の何が条件で何が動作かという判断には経験や知識などスキルに依存する部分がある。本論文のアプローチは、自然言語処理を利用し、仕様の条件と動作の判断を機械的に行うことである。複数の条件と動作の間で、ある条件がある値をとる場合に他の条件や動作の存在そのものやとりうる値に影響を与える、という制約も仕様書には存在する。このような制約の抽出は、仕様とは別の仕様の情報へのアクセスや知識が必要となるため、本論文では対象とせず今後の課題とする。

2.2 自然言語処理の流れ

自然言語処理は、大きく次の4つの解析ステップに分けることができる[12]。

1. 形態素解析：テキストを単語に分割する。また、単語が語形変化している場合は、原形へ戻す。さらに、単語の品詞を決定する。
2. 構文解析：単語間の構文的関係を決定する。
3. 意味解析：単語、文の意味を決定する。
4. 文脈解析：複数の文にまたがる処理を行う。

単語間の関係を格といい、構文的な表層格と意味的な深層格に分けられる。また、文中の名詞と動詞に単語間の関係、つまり格がある場合、その格の深層格を同定することを意味役割付与という[12]。

意味役割付与の研究[17]では、動詞項構造ソーラスに掲載されている動詞語義と意味役割を、文の中か

ら見つけて付与している。本論文では、文中の単語間に格がある場合、それがデシジョンテーブルテスト技法における条件なのか動作なのか同定することを、テストケース作成のための意味役割付与としている。例えば、単語間の関係はヲ格やガ格などの表層格であるが、その単語間の係り受けも含めた判断ルールから、条件か動作かを同定している。

2.3 課題とアプローチ

デシジョンテーブルテスト技法における条件と動作の抽出に、自然言語処理を利用することを考えた場合、これらを直接抽出する方法には課題がある。自然文から自然言語処理により「(A)が(B)の場合(C)は(D)である」という論理を抽出している関連研究[7]がある。この研究では、(A)が条件の変数、(B)が条件の値、(C)が動作の変数、(D)が動作の値としている。しかし、このアプローチは条件と動作はそれぞれ1個であり、複数の条件と動作は抽出していない。他のアプローチとして、条件と動作の抽出は、日本語自然文を論理式に変換するという課題に置き換えることと考えられる。これについて、日本語文から拡張型述語論理式への自動変換ツール[15]が発表されている。しかし、文型と論理式ラベルの対応表など仕様書やテストに沿った辞書の作成が必要となり、本論文では対象としておらず今後の課題である。

本論文では、自然言語処理による条件と動作の抽出を意味役割付与の方法と捉え、自然言語処理の解析結果から得られる情報を基に意味役割付与を行うアプローチを提案する。現実のソフトウェアの仕様書においては、曖昧な記述や論理不明瞭な文も存在する。これに対しては、条件と動作が明確に記述された雛形文体と構文的類似度を比較することで、条件と動作の抽出に適した文を振り分けることとする。また、箇条書きのリスト構造の文のように複数の文から機能を定義することがある。複数の文の文間関係や含意関係の研究[8]もあるが、本論文では、一文ごとの解析精度向上に取り組む。

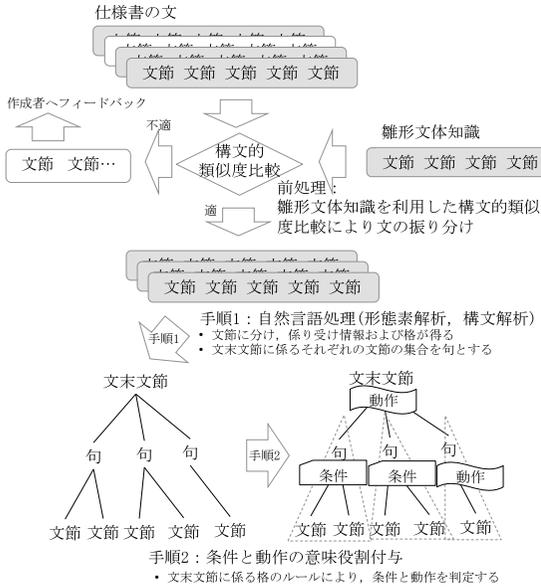


図1 条件と動作の抽出方法

3 条件と動作を抽出する意味役割付与の方法

3.1 抽出方法概要

条件と動作を抽出する意味役割付与の方法を図1に示す。まず、前処理で仕様書の文と雛形文体との間の構文的類似度を算出し、条件と動作の抽出に適した文に振り分ける処理を行う。これにより、条件と動作の抽出の精度向上を計る。文体とは、文における品詞と係り受けの構造を指す。仕様書の文には、条件と動作の抽出に適した文とそうでない文が存在する。前者の抽出に適した文は条件と動作の抽出処理へ、後者の抽出に適していない文は作成者フィードバックをすることで記述の向上が見込まれる。次に、対象文に対して自然言語処理の係り受けと図1手順1の格解析の結果から、図1手順2の判断ルールにより条件と動作の意味役割を付与する。

3.2 構文的類似度比較

構文的類似度比較の目的は、仕様書の文の文体を、条件と動作が明確に記述された雛形文体と比較することにより、条件と動作の抽出に適した文を振り分けることである。条件と動作が明確に記述された雛形文は、「年齢が20歳以上の場合、入場料1000円を表

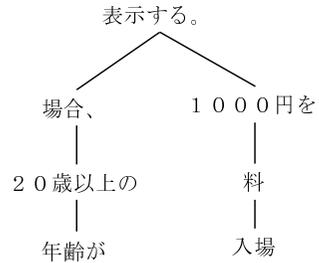


図2 雛形文体の係り受け構造

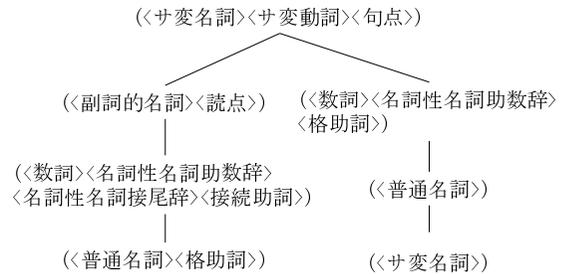


図3 雛形文体の品詞の構造

示する。」とした。この文は図2のように「年齢が20歳以上の場合、」という条件と「入場料1000円を表示する。」という動作の記述を持つ、本論文の意味役割付与方法に適した文体である。雛形文体知識は、雛形文体の集合であり、条件と動作の記述を持つ文体の品詞とその係り受けの木構造表現とした。本論文では、図3の雛形文体とした。この構造をBNF記法で表現すると次のとおりである。

```

<雛形文体> ::=
(
  ( <普通名詞><格助詞> )
  ( <数詞><名詞性名詞助数辞><名詞性名詞接尾辞><接続助詞> )
  ( <副詞的名詞><読点> )
)
(
  ( <サ変名詞> )
  ( <普通名詞> )
  ( <数詞><名詞性名詞助数辞><格助詞> )
)
<サ変名詞><サ変動詞><句点>

```

この雛形文体と仕様書文の文体の構文的類似度比較を行うことにより、例えば、「個数が10個以上の場合、配送料500円を徴収する。」のような、文体が似ている文を振り分けることができる。

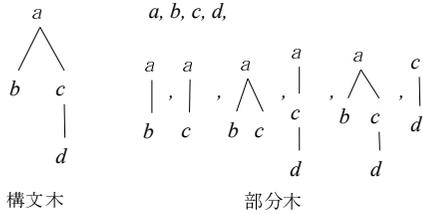


図4 木カーネルにおける構文木と部分木

構文的類似度の比較には木カーネルを用いた。木カーネルは、構文木の類似度を計算する方法である。木カーネルの類似度の計算は、図4のように構文木を部分木に分け、2つの構文木の間で部分木を共通に持つ数の計算により測定することができる[13]。類似度の計算式は式(1)のように、木カーネルの内積からコサイン類似度を求めた[1][11]。

$$\cos \theta = \frac{k(T_i, T_j)}{\sqrt{k(T_i, T_i)k(T_j, T_j)}} \quad (1)$$

構文木を T_i, T_j とする。カーネルを k とする。 T_i, T_j の成す角を θ とする。

3.3 条件と動作抽出と意味役割付与

提案方法では、システムの仕様書に記述される文章は、第3者に仕様を伝えるために標準化された記述であると想定している。例えば、下記のような文である。

- 「在庫量が設定値を下回った場合発注を行う。」
- 「重複予約が発生した場合、その内容をエラーメッセージとして表示する。」

このような仕様書の文には「～の場合～とする」のように「～とする」の文末文節に係り受けている句が存在する。文末文節に係る場合を除いては文中の文節は直後の文節に係ることが最も多い性質[20]がある。つまり、文末文節に係る文節は関係があると考えられる。よって、本論文では、文末文節に係り受けている句の中で、動作を表すものは動作句、それ以外のは条件句から成っていると捉える。これらの関係は自然言語処理の格解析の結果として抽出される。この格解析結果の表層格のうち、まず、ガ格とヲ格は格の定義から動作主を表すので動作と判断する。なお、ヲ格が他動詞を取る場合は動作主を表すものではないが、

テストケースとしては動作と判断する。

また、本論文で使用した、格解析方法[6]では、格に「連格」と「文節内」がある。「連格」は被連体修飾詞の連体修飾節に対する格解析結果である。「文節内」は複合名詞の主辞基本句以外に付与されるものである。よって、「連格」と「文節内」は文末文節に連なり語句を形成するのでこれも動作と判断する。これら以外の格は条件と判断する。

条件と動作を抽出する意味役割付与方法を定義する。

- 文を S とする
- 文 S の文節を $m(i)$ とする
- 文末の文節番号を T とし文末文節を $m(T)$ とする
- $m(i)$ の係り受け先の文節番号を $D(i)$ とする
- $m(i)$ の係り受け先との格を $C(i)$ とする
- $m(i)$ を係り受け先とし、係り受け関係のあるすべての文節の集合を句 $B(i)$ とする
 - この $m(i)$ を句 $B(i)$ の終端文節とする
- $B(i)$ の属性として、条件と動作を C_n と A_c とする
- 条件と動作の判断ルールの集合を R とする
 - R_a は動作と判断されるルールとする
 - * $Ra1$. 文末文節である
 - * $Ra2$. 動作主を表す「係:ヲ格」, 「係:ガ格」, 「係:連格」, 「係:文節内」である
 - R_c は条件と判断されるルールとする
 - * $Rc1$. 動作と判断されない

アルゴリズムは図5のとおりである。

図6に動作と条件の抽出方法の適用例を例文(1)を用いて示す。

例文(1)：

- 「平成23年12月31日以前に締結した契約であっても、平成24年1月1日以後に更新・特約中途付加などを行った場合は、異動日以後、契約全体に対して新制度の控除区分が適用されます」([4]を基に編集)

この例文の文末文節「適用されます」に係る句およびその格は、形態素および構文解析の結果により、4つである。それぞれ、「場合は」は(係:未格), 「以後」は(係:無格), 「対して」は(係:複合辞連用), 「区分が」

条件と動作の抽出アルゴリズム	
入力:	形態素解析, 構文解析, 意味解析結果
1:	文末文節 $m(T)$ を探す
2:	$m(T)$ に係る文節 $m(i):D(i)=T$ を探す
3:	for all $m(i):D(i)=T$
4:	$m(i)$ を句の終端文節とする句 $B(i)$ を作成する
5:	if $C(i)$ が Ra に含まれる
6:	then $B(i)$ を動作と判定する
7:	else $B(i)$ を条件と判定する
8:	endif
9:	end for

図5 条件と動作の抽出アルゴリズム

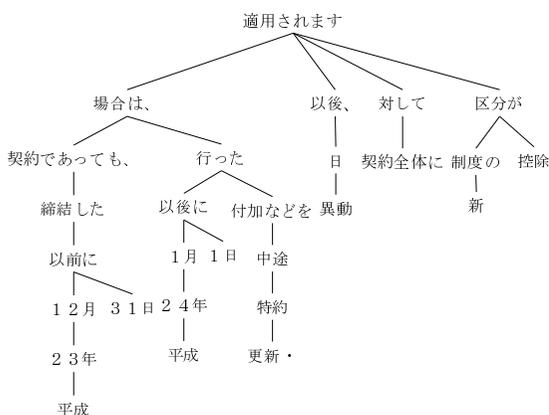


図6 例文の係り受け構造

は(係:ガ格)である。判断ルール $Rc1$, $Ra1$, $Ra2$ により、条件と判断される句は、「契約全体に対して」、「異動日以後」、「平成23年12月31日以前に締結した契約であっても、平成24年1月1日以後に更新・特約中途付加などを行った場合は」の3個である。動作と判断される句は「新制度の控除区分が」、「適用されます」の2個である。この例のように仕様書の文に対して、条件と動作の意味役割を付与する。

4 実験評価

4.1 実装

実験評価のため提案方法を実装した。自然言語処理の実装は日本語形態素解析および構文解析ツール[5]を用いた。構文的類似度の比較は木カーネルのア

ルゴリズム[13]を用いた。アルゴリズムの実装はスクリプト言語[9]を用いた。提案方法は特定の自然言語処理の実装に依存するものではない。しかし、係り受け解析や格解析結果は、それぞれの自然言語処理の辞書や解析方法の違いなどの実装によって、異なる結果が出てくることが考えられる。これらの自然言語処理の実装による係り受け解析や格解析結果の違いに対する評価実験は、本論文では対象とせず今後の研究課題とする。

実験は、雛形文体知識による構文的類似度比較を用いた前処理が無い場合と、前処理が有る場合の実験を行った。構文的類似度を用いた振り分けの閾値は、自然言語処理で一般的[10][14]な0.30とした。この閾値は、表1に示す、本論文の例文における有識者の適・不適判断と構文的類似度の比較においても妥当であることを確認した。

また、実験の対象文書として、記述スタイルの異なる2つのタイプの文書に対して行った。ひとつはユースケース記述で記述されたシステム設計書で、全部で871個の文を含む。この文書を文書タイプA^{†1}と呼ぶ。もうひとつは、インターネット上で入手可能なシステム設計書であり、例えば消防システム基本設計書[18]など、主に公的組織の23個の調達仕様書に記述されている機能記述を抜き出したもので、全部で711個の文を含む。文書タイプBの文書一覧を本論文 Appendix A. の表9に載せている。これらの文書群を文書タイプBとする。

例文はそれぞれ以下の通りである。

- 文書タイプAの文例:

- 「システムは、申請者の指定したファイルを取得し、添付資料を開く。」
- 「入力された項目数が正しくない場合、システムは、項目数が正しくない旨を表示する。」

- 文書タイプBの文例:

- 「あらかじめ設定した時間以上システムが操作されていないと判断される場合は、当該利用者を自動的にログアウトすること(自動ログアウト機能).」

^{†1} 文書タイプAは企業内の公開できない資料である。

表 1 有識者判断と構文的類似度の比較

例文	有識者判断	構文的類似度
年齢が 20 歳以上の場合、入場料 1000 円を表示する。	適	1.000
個数が 10 個以上の場合、配送料 500 円を徴収する。	適	1.000
ログオン・ユーザーが対象会社ではない場合、システムは、対象ユーザーではない旨を表示する。	適	0.594
年齢が 20 歳以上で女性の場合、500 円を表示する。	適	0.563
ユーザーは、ログインし、イントラネット ID およびパスワードを入力する。	適	0.500
納期が 3 日未満の場合、追加料 800 円を徴収し、係員が手配をする。	適	0.313
その際に担当が代わる場合は、「担当メンテ」を使用し、発行の担当者名を変更する必要あり。	不適	0.292
当日受付確認処理された患者情報一覧を表示すること (主に医師が診察室で使用する)。	不適	0.271
時間管理が出来ること (オーダー時間、受付時間、採血時間等)	不適	0.200
何もせずに終わる場合→終了。	不適	0.125

表 2 前処理が無い場合と有る場合の対象文書の数

前処理	文書タイプ A	文書タイプ B
無し	871	711
有り	844	667

- 「(依頼に基づく場合、管理上必須一覧の項目に番号が追加されることが必要な場合等) 会社等一覧表を抽出、出力する。」

この文書タイプ A と文書タイプ B それぞれに対して、構文的類似度による振り分けの前処理が無い場合と有る場合の対象文書の数は、表 2 のとおりである。

4.2 精度と再現率

デジジョンテーブルテスト技法の有識者が、条件と動作それぞれの抽出に対して実験結果の評価を行った。実験で条件または動作と判断された句で、有識者によって条件または動作と判断された句は真陽性 (TP: True Positive) とした。実験で条件または動作と判断された句で、有識者によって条件または動作ではないと判断された句は偽陽性 (FP: False Positive) とした。実験で条件または動作と判断されなかった句で、有識者によって条件または動作と判断された句は偽陰性 (FN: False Negative) とした。また、提案方法によって条件と判断された句に条件と動作の

両方の文節が含まれていると評価された場合は、条件の抽出は真陽性 TP とし、動作の抽出は偽陰性 FP とした。それぞれの件数を数え、式 (2), (3), (4) に示す計算式のとおり、Precision (P: 精度), Recall (R: 再現率) および F-Measure (F: 調和平均) を算出した。なお、この評価は抽出された条件と動作に対して行ったものであり、振り分けの結果から対象文とならなかった文については評価対象には入っていない。

$$Precision = \frac{TP}{(TP + FP)} \quad (2)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (3)$$

$$F = 2 \times Precision \times \frac{Recall}{(Precision + Recall)} \quad (4)$$

表 3 および表 4 に評価結果を示す。前処理無しの場合、Precision が 0.888 から 0.985, Recall が 0.944 から 0.973, F は 0.923 から 0.964 の値となった。

同様の既存研究 [16] では、日本語ユースケース記述の Action 抽出の評価結果が Precision は 0.815, Recall は 0.946 である。既存研究と比較し同等以上の結果となった。既存研究 [16] は、自然言語で記述された仕様書から、条件や動作を抽出するという点で本論文と同様である。この既存研究と本論文では、条件

表 3 条件と動作の抽出の評価件数

前処理	文書タイプ	条件の抽出評価件数			動作の抽出評価件数		
		TP	FP	FN	TP	FP	FN
無し	A	1,545	102	43	1,634	25	96
	B	1,113	140	47	1,215	19	72
有り	A	1,489	99	39	1,614	20	92
	B	1,039	114	43	1,187	18	56

表 4 Precision-Recall-F 評価

前処理	文書タイプ	条件の抽出			動作の抽出		
		Precision	Recall	F	Precision	Recall	F
無し	A	0.938	0.973	0.955	0.985	0.945	0.964
	B	0.888	0.959	0.923	0.985	0.944	0.964
有り	A	0.938	0.974	0.956	0.988	0.946	0.966
	B	0.901	0.960	0.930	0.985	0.955	0.970

や動作の判断ルールが異なる。既存研究の日本語に対する判断は、「場合」など文中のキーワードに対して、単語の文中位置の前後で判断している。本論文では、文中の係り受けおよび格で判断している。この判断ルールが既存研究に比べて新しい点である。

前処理無しの文書タイプ B の条件の抽出の Precision が他と比べて低い結果となっている。前処理有りの場合は、Precision が 0.901 から 0.988, Recall が 0.946 から 0.974, F は 0.930 から 0.970 の値となった。前処理有りの文書タイプ B の条件の抽出の Precision が向上している。これについては、考察で述べる。

4.3 構文的類似度比較を用いた前処理

本論文では、構文的類似度を木カーネルの内積のコサイン類似度とし、閾値を例文による類似度から 0.30 で振り分けることとした。その結果、表 2 に示すように、対象文について文書タイプ A は 871 個から 844 個に、文書タイプ B は 711 個から 667 個になった。比率では、文書タイプ A は 3.1%、文書タイプ B は 6.2% が対象とならなかった。構文的類似度比較は、仕様書の文の文体を条件と動作が明確に記述された雛形文体とすることで、条件と動作の抽出に適した文を振り分けることである。表 4 に示すように、構文的類似度比較を用いた前処理が有る場合は無い場合と比較し、同等以上の解析精度となった。特に、文

書タイプ B の Precision の向上が前処理無しの場合 0.888 に対し、前処理有りで 0.901 と向上した。よって、本論文での構文的類似度比較は一定の効果があったと判断する。構文的類似度の閾値について考察する。この閾値の値を上げれば、より条件と動作の抽出に適した文が対象となるので、その次の処理である条件と動作の意味役割付与の Precision は上がると考えられる。一方、振り分け前の文を全体とした場合の対象範囲は下がり、有用性に影響が出ると考えられる。振り分けられ対象とならなかった文については、構造の類似度情報を付けて作成者に早期にフィードバックし修正を促すことが挙げられる。このフィードバックをすることにより、有用性は高まると考えられる。

4.4 実験対象の文書タイプ

本論文の提案方法は、対象とする文の記述スタイルの違いによって結果が異なることが予想される。句の終端文節の格情報を用いて文の句パターンを

<句パターン> ::= (<句の終端文節の格情報>)

| {(<句の終端文節の格情報>)} <文末>

と表現し、文書タイプ A と文書タイプ B の違いを評価した。句パターンは提案方法による解析情報から自動的に作成されたものである。例えば、図 6 の例文は、1 番目の句が未格、2 番目が無格、3 番目が複合辞連用、4 番目がガ格、そして文末である。これは <未

表 5 文書タイプ A の句パターン上位 10 件

文書タイプ	句パターン	個数	個数割合	累積割合
A	<連用> <未格> <ヲ格> <文末>	177	20.4%	20.4%
	<未格> <ヲ格> <文末>	95	10.9%	31.3%
	<未格> <連用> <ヲ格> <文末>	80	9.2%	40.6%
	<連用> <ヲ格> <文末>	58	6.7%	47.2%
	<連用> <未格> <連用> <ヲ格> <文末>	43	5.0%	52.2%
	<ヲ格> <文末>	23	2.6%	54.8%
	<ヲ格> <連格> <文末>	18	2.1%	56.9%
	<未格> <カラ格> <ヲ格> <文末>	18	2.1%	59.0%
	<連格> <文末>	12	1.4%	60.4%
	<未格> <ニ格> <ヲ格> <文末>	12	1.4%	61.8%

表 6 文書タイプ B の句パターン上位 10 件

文書タイプ	句パターン	個数	個数割合	累積割合
B	<未格> <連格> <文末>	88	12.6%	12.6%
	<連用> <連格> <文末>	70	10.0%	22.6%
	<連用> <未格> <連格> <文末>	21	3.0%	25.6%
	<連用> <ヲ格> <文末>	18	2.6%	28.1%
	<未格> <ヲ格> <文末>	18	2.6%	30.7%
	<ノ格> <文節内> <文末>	17	2.4%	33.1%
	<連格> <文末>	13	1.9%	35.0%
	<ガ格> <連格> <文末>	12	1.7%	36.7%
	<未格> <連用> <ヲ格> <文末>	9	1.3%	38.0%
	<連用> <デ格> <連格> <文末>	9	1.3%	39.3%

格><無格><複合辞連用><ガ格><文末>と表現される。

文書タイプ A の句パターン上位 10 件を表 5 に、文書タイプ B の句パターン上位 10 件を表 6 に示す。

文書タイプ A では、句パターン「<連用><未格><ヲ格><文末>」が最も多くあった。これは「～の場合～は～を～する」という文である。例えば、「会社情報を取得できない場合、システムは、会社情報が取得できなかった旨を表示する。」という文である。この句パターンが文の個数で 177 個、全文数に対しては 20.4%であった。文書タイプ A は上位 10 パターンの累計で全体の 61.8%を占めた。句パターンにおける条件と動作の抽出を考察する。文書タイプ A で最も多くあった句パターンに入る文として、上記例文「会社情報を取得できない場合、システムは、会社情報が取得できなかった旨を表示する。」を挙げる。この文は「会社情報を取得できない場合」が「係:連用」で、「システムは」が「係:未格」で、「会社情報が取得できなかった旨を」が「係:ヲ格」で、それぞれ文末文節の「表示する」に係る。よって、「会社情報を取得で

きない場合」および「システムは」を条件と判断し、「会社情報が取得できなかった旨を表示する」を動作と判断する。これは有識者の評価でも正しいと判定された。

文書タイプ B では句パターン「<未格><連格><文末>」が最も多くあり、これは「～は～すること」という文である。この句パターンでは「こと」が文末文節である。例えば、「システムは終了すること」という文である。この句パターンが文の個数で 88 個、全文数に対しては 12.6%であった。文書タイプ B は上位 10 パターンの累計で全体の 39.3%を占めた。文書タイプ B の句パターンの数で 10 番目の<連用><デ格><連格><文末>に入る文として、「立入検査指摘事項に指摘があれば、届出をシステム入力することで、自動的に履行登録されること。」を挙げる。この文は、「立入検査指摘事項に指摘があれば」が「係:連用」で、「届出をシステム入力することで」が「係:デ格」で、「自動的に履行登録される」が「係:連格」でそれぞれ文末文節の「こと」に係る。よって、「立入検査指摘

表 7 手作業との Precision-Recall の比較

		Precision	Recall	F
手作業	例 1	1.000	0.756	0.861
	例 2	1.000	0.393	0.564
提案方法	平均	0.949	0.955	0.952

事項に指摘があれば」および「届出をシステム入力することで」を条件と判断し、「自動的に履行登録される」を動作と判断する。これは有識者の評価でも正しいと判定された。

文書タイプ A は、句パターンの最大数は 177 個、種類は 157 個であり、文書タイプ B は、それぞれ 88 個、276 個である。句パターンの種類の数を記述のばらつきと捉えた場合、文書タイプ B は文書タイプ A に比べてばらつきが多いといえる。これは、文書タイプ B は、23 個の異なるシステムの文書から集められた 711 個の文であるのに対し、文書タイプ A は 1 個のシステムの文書から集められた 871 個の文であることから予想された傾向であると考えられる。しかし、このように記述にばらつきのある文に対しても、提案方法では表 4 に示すようにほぼ同等の高い Precision と Recall を確認できた。

4.5 手作業との比較

提案方法と手作業の精度と再現率の比較を行う。手作業によるテストケース作成の比較に関連研究 [19] を用いる。この手作業事例では、テスト対象として抽出すべき仕様の数 78 個に対して、手作業で抽出したテスト対象の仕様数は 59 個である。これを手作業例 1 とする。この手作業事例のもうひとつの例では、抽出すべき仕様数 56 個のうち手作業では 22 個を抽出した。これを手作業例 2 とする。手作業で作成したテストケースは、抽出すべき条件と動作を含んだ仕様であった。この結果を、それぞれ Precision, Recall, F-Measure を計算し提案方法を比較すると、表 7 のとおりである。提案方法は Precision は下回ったが Recall および F-Measure は上回った。これについては考察で述べる。

また、提案方法と手作業との作業工数の比較を行う。提案方法の実験は、ハードウェアは CPU 2.60GHz,

表 8 手作業との処理時間比較 (単位:分)

方法 文書タイプ	手作業		提案方法	
	A	B	A	B
手順 1. 自然言語処理	240	210	42	36
手順 2. 条件と動作抽出			1	1

メモリ 8GB であり、自然言語処理ツール [5] を使い、アルゴリズム実装はスクリプト言語 [9] で行った。手作業事例はシステム開発経験 20 年のエンジニアが、対象文書からテストケースの条件と動作を抽出する作業としている。処理の比較対象作業は条件と動作の抽出のみである。提案方法の手順 1 および手順 2 で行う場合と手作業で行う場合を比較した。手作業および提案方法の作業工数は表 8 のとおりである。提案方法は手作業の約 6 分の 1 の処理時間で条件と動作を抽出した。

5 考察

提案方法は、既存研究と同等以上の精度と再現率を持つことを確認した。また、構文的類似度比較を用いた前処理により、記述スタイルにばらつきがある文書タイプ B の Precision が向上した。テストケースとしては理想的にはテストすべき機能数に対して 100% のカバレッジが求められる場合もある。しかし、本論文の対象としているエンタープライズシステムにおけるシステムテストやユーザー受け入れテストなどテストケースにおいては、手作業と比較した場合の抜け漏れなどを考慮すると、今回の精度と再現率は実務上使用可能な範囲と考える。

文書タイプの違いによる精度と再現率について考察する。文書タイプ A はユースケース記述であり、そのプロジェクトの文書記述標準に則った記述となっている。これは本論文で提案した想定する文構造によく適合するものであった。文書タイプ A の文は、例えば「ログオン・ユーザーが対象会社ではない場合、システムは、対象ユーザーではない旨を表示する。」のように、条件と動作が分かるように記述されている。提案方法では、この例文では「ログオン・ユーザーが対象会社ではない場合」、「システムは」を条件、「対象ユーザーではない旨を表示する。」を動作と判断した。

記述スタイルにばらつきがある文書タイプ B も、既存研究 [16] と比較し同等以上の結果を確認した。考察すべき点は、前処理が無い場合の文書タイプ B の条件の抽出の Precision が他と比べ低い値となったことである。これは主に動作である修飾詞を条件として抽出していたことが原因である。例えば、「あらかじめ設定した時間以上システムが操作されていないと判断される場合は、当該利用者を自動的にログアウトすること (自動ログアウト機能).」のように、文の最後に括弧で機能の要約をつけている文があった。本論文の判断方法では、この文の「(自動ログアウト機能)」を動作と判断してしまい、その他の文節は条件となってしまう。

このような文の記述パターンに対しては、前処理を行い解析対象から外すことで解析対象の文の精度は向上した。解析対象から外された文は、テストケース作成にあたり条件と動作の抽出に適していない文である。これを文の作成者に早期にフィードバックすることにより修正を促せば、実務上有益であると考ええる。

また、提案方法の結果を利用し句パターンの比較を行った。この句パターンの比較は記述のばらつき測定の要求分析にも利用できると考える。さらに、提案方法の適用により発生したエラーの分析から、第 3 者にもより分かりやすい記述方法をフィードバックするという応用も考えられる。

さらに、本論文の提案方法は、手作業に比べ Precision は下回ったが Recall, F-Measure で上回った。手作業の Precision が 1.000 という値は、抽出すべき仕様の中から少ない工数などの制約下で抽出すべき重要な仕様を抽出していると考えられる。品質まで評価が必要ではあるが、手作業の中でこれらの仕様の選択基準は適切であったかは、今後の参考資料となり得る。処理時間に関しては、約 6 分の 1 の処理時間で条件と動作の抽出を行うことを確認した。これは、テストケース作成自動化ための方法として有効な候補であると考ええる。

本論文の提案方法を発展させるための研究課題としては、複数の文による機能記述への対応がある。仕様書には、箇条書きのリスト構造の文のように複数の文から機能を定義することがある。これに対しては

複数の文間の含意関係などを分析する技術を取り入れるアプローチがある。また、条件と動作間の制約についても、発展させるために必要な研究課題である。この制約については、論理式変換 [15] や辞書の整備などのアプローチがあると考ええる。

6 おわりに

本論文では、テストケース作成の自動化のための意味役割付与の方法を提案すると共に実験評価した。提案した手法は、評価結果から、記述スタイルの違う文書に対しても、既存研究と同等以上の精度と再現率を確認した。判断のルールや条件間の制約を対応させることにより、さらに高い精度と再現率とすることが可能であると考ええる。これらのことは、提案手法が構文木の照合が基本ロジックとなっているため、拡張性が高いことに起因している。この拡張性により、精度向上と共に汎用性向上も実現可能と考える。今後は、第 3 者にもより分かりやすい記述方法をフィードバックするという、提案方法の応用にも取り組む所存である。

参考文献

- [1] Danilo, C., Alessandro, M. and Roberto, B.: Structured Lexical Similarity via Convolution Kernels on Dependency Trees, in *Association for Computational Linguistics Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011, pp.1034–1046.
- [2] ISO/IEC/IEEE JTC 1/SC 7: Software and systems engineering - Software testing - Part 1 Concepts and Definitions, in *ISO/IEC/IEEE JTC 1/SC 7*, 2015, p. 9
- [3] ISO/IEC/IEEE JTC 1/SC 7: Software and systems engineering - Software testing - Part 4 Test techniques, in *ISO/IEC/IEEE JTC 1/SC 7*, 2015, pp. 70–72.
- [4] 国税庁: No.1140 生命保険料控除, <https://www.nta.go.jp/taxanswer/shotoku/1140.htm>, 2015 年 6 月 30 日アクセス
- [5] 京都大学黒橋・河原研究室: 自然言語処理のためのリソース, <http://nlp.ist.i.kyoto-u.ac.jp/index.php?NLP> リソース, 2015 年 6 月 30 日アクセス
- [6] 京都大学黒橋・河原研究室: KNP で付与される feature 一覧, http://nlp.ist.i.kyoto-u.ac.jp/index.php?plugin=attach&refer=KNP&openfile=knf_feature.pdf, 2016 年 4 月 30 日アクセス
- [7] Masuda, S., Hosokawa, N., Iwama, F., Matsudani, T. and Tsuda, K.: Semantic analysis tech-

表 9 文書タイプ B の文書一覧

文書名	著者・作成者等
求人情報管理システム操作マニュアル	JS 出雲島根県
消防業務支援システム基本設計書	横浜市
予防接種業務支援システム開発業務委託仕様書	横浜市健康福祉局健康安全課
京都市立病院総合情報システム仕様書	京都市立病院
京都府立新総合資料館(仮称)統合情報システム 図書系システム機能要求仕様書	京都府
別紙 1 年金業務システム 基本設計書 第 1.0 版 概要	厚生労働省
WEB 公開型犯罪情報システム開発等業務仕様書	広島県警察本部
甲府市地域包括支援センター支援システム仕様書	甲府市
高知県立図書館・高知市民図書館新図書館情報システム等基本設計委託業務仕様書	高知県と高知市
新図書館情報システム構築等委託業務仕様書	高知県と高知市
「高知市家屋評価システム」仕様書 (RFP)	高知市財務部資産税課
『栄養管理システムソフトウェア一式』仕様書	国立障害者リハビリテーションセンター
佐賀市 学齢簿・就学援助システム開発要件定義書	佐賀市
佐賀中部広域連合地域包括支援センターシステム要件定義書	佐賀中部広域連合
国有財産総合情報管理システムに係る設計・開発及び移行業務一式仕様書(案)	財務省理財局管理課
山梨県共同システム基本設計書	山梨県市町村総合事務組合
地方独立行政法人市立秋田総合病院財務会計システム構築業務委託仕様書	秋田市福祉保健部病院法人移行準備室
松江市上下水道局水道施設管理マッピングシステム構築業務要求仕様書	松江市上下水道局
焼津市内部情報系システム整備事業 要求仕様書(共通)	焼津市
救急統計管理システム仕様書	笛吹市消防本部
汎用受付システム基本設計書	島根県
OSS オープン・ラボ システム開発・構築作業(第 1 次強化)	独立行政法人情報処理推進機構
脳卒中遠隔画像伝送システム仕様書	能登脳卒中地域連携協議会

nique of logics retrieval for software testing from specification documents, in *IEEE Software Testing, Verification and Validation Workshops (ICSTW)*, 2015, pp. 1–6.

- [8] 水野淳太, 渡邊陽太郎, エリックニコルズ, 村上浩司, 乾健太郎, 松本裕治: 文間関係認識に基づく賛成・反対意見の俯瞰, 情報処理学会論文誌, (2011), pp. 3408–3422.
- [9] Natural Language Tool Kit (NLTK) Project: Natural Language Toolkit, <http://www.nltk.org/>, 2015/9/20 Accessed.
- [10] グェントアンドック, ボレガラダヌシカ, 石塚満: エンティティペア間類似性を利用した潜在関係検索, 情報処理学会論文誌, 2011, pp. 1790–1802.
- [11] 小田悠介, 若林茂: プログラム間の類似性の定量化手法, 神戸高専研究紀要 51 号 (平成 25 年), 2013, pp. 103–108.
- [12] 奥村学: 自然言語処理の基礎, コロナ社, 2010.
- [13] 高橋哲郎, 乾健太郎, 松本裕治: テキストの構文的類似度の評価方法について, 情報処理学会自然言語処理研究会, NL-150-7, 2002.
- [14] 高瀬翔, 岡崎直観, 乾健太郎: 高速な類似度計算手法による関係パタンのクラスタリング, 言語処理学会第 20 回年次大会 発表論文集, 2014, pp. 47–50.
- [15] 高柳俊祐, 上条敦史, 石川勉: 日本語文から拡張型述語論理式への自動変換ツール: CONV, 人工知能学会論文誌, Vol. 17(2012), pp. 271–280.
- [16] 竹内広宜, 中村大賀, 山口高平: テキスト分析技術を用いたユースケース分析, 電子情報通信学会技術研究報告知能ソフトウェア工学, 2010, pp. 55–60.
- [17] 竹内孔一, 土山傑, 守屋将人, 森安祐樹: 類似した動作や状況を検索するための意味役割及び動詞語義付与システムの構築, 言語理解とコミュニケーション研究会, NLC2009-33 pp. 1–6, 2010.1.25.

- [18] 横浜市: 消防業務支援システム基本設計書, <http://www.city.yokohama.lg.jp/shobo/koukai/ippan-nyuusatsu-pdf/shoubou.kihonnsekkeisyotou.pdf>, 2014 年 12 月 26 日アクセス
- [19] Yumoto, T., Matsudani, T. and Tsuda, K.: A Study on an Approach for Analyzing Test Basis Using I/O Test Data Patterns, in *IEEE Software Testing, Verification and Validation Workshops (ICSTW)*, 2015, pp. 1–6.
- [20] 張玉潔, 尾関和彦: 文節間係り受け距離の統計的性質とその係り受け解析への応用, 電子情報通信学会技術研究報告, 1995, pp. 61–68.

A 文書タイプ B の文書一覧

本論文で仕様した文書タイプ B の文書一覧は表 9 のとおり. 文書はインターネット上で入手可能なシステム設計書であり, 主に公的組織の調達仕様書である.



増田 聡

1991 年埼玉大学工学部機械工学科卒. 同年日本アイ・ビー・エム株式会社入社, エンタープライズ・アプリケーションにおけるソフトウェアテストを中心としたソフトウェア・エンジニアリングに従

事。2014年同社東京基礎研究所に異動，サービス・ソフトウェア・エンジニアリング研究に従事。情報処理学会，人工知能学会，IEEE Computer Societyの会員，情報処理学会情報規格調査会WG 26ソフトウェアテストワーキンググループ幹事，特定非営利活動法人ソフトウェアテスト技術振興協会理事。



松尾谷 徹

2005年筑波大学大学院博士課程修了，博士(システムズ・マネジメント)。

日本電気株式会社周辺装置やOSの開発に従事，プロジェクト支援を経て社内にモダン・プロジェクト・マネジメント・アカデミーを設立し指導を行う。2002年独立し有限会社

デバッグ工学研究所を設立，現在同社代表コンサルタント，法政大学非常勤講師，情報処理学会正会員，IEEE Computer Society 正会員。



津田 和彦

1986年徳島大学工学部情報工学科卒。1986年三菱電機，1991年住友金属工業，1994年徳島大学大学院工学研究科システム工学専攻修了，博士(工学)。

1998年筑波大学大学院助教授，2005年，同大学院教授。自然言語理解，情報検索，データベース，アルゴリズムの研究に従事。IEEE Computer Society，情報処理学会，電子情報通信学会，電気学会，人工知能学会，計測自動制御学会の会員。