

システム開発のテストケース作成支援ツールの開発

村上 神龍† 村上 響一† 青山 裕介† 久代 紀之†

九州工業大学†

1. はじめに

システムテストを実行する際、テスト設計者はシステム仕様書からテスト仕様書を作成する。テスト設計者は自然言語で記載されたシステム仕様書内の文章から自らの知識・経験に基づいてテストケースの作成を行うため、作成されたテストケースがテスト設計者に依存してしまう。さらに、テストケース作成する際、テスト設計者が手作業で行うため時間がかかり、人為的ミスが起りやすくなる。

また、テスト設計のプロセスが暗黙的であり、作成したテストケースの設計根拠がレビューをする際に理解できない場合がある。これらの問題を解決するために、システム仕様書記載の文章又は表からテストケースの作成を支援するツールの開発を行う。

2. ツールの概要

テストケース生成・支援ツールについて説明する。テストケースの生成支援を行うため図1のようなテスト設計プロセスを提案する。

仕様書の欠落情報を抽出するため文章をセミ形式記述(関係詞(主体, 対象, [制約,{“, ” 制約})) [1]に変換する。次に変換したセミ形式記述を条件文と動作文に分離する。判定方法として、「場合」、「時」など条件文となる語を持つ文節や、「一たら」、「一ので」など条件文となる接続語を持つ文節を条件文とし、それ以外は動作文とする。セミ形式記述に漏れがあった場合は、「??」で表示されることで、記述漏れの発見を促すことが可能である。

セミ形式記述の変換により、関係詞ごとに単文化を行うため前後の論理関係を表現出来ない場合がある。条件文と動作文を命題ネットワーク [1]で可視化することによりセミ形式記述間の論理関係を可視化する。

さらに、分離した条件・動作文をデシジョンテーブル [2]として記述する。

例として「水温が100度を超えた場合、ヒータ用電源をoffにする。」という文章を入力した際、図2のセミ形式記述、図3の命題ネットワーク、図4のデシジョンテーブルを出力する。

テストケースには、条件文、動作文等の論理関係に加えて、テスト実行の際に用いる値域が必要となる。本研究では、値域を表現する語を抽出し、同値・境界値法を用いて具体的な値を決定する。抽出した値をテストケースとして出力できるようにすることで実行可能なテストケースを作成できるようにする。

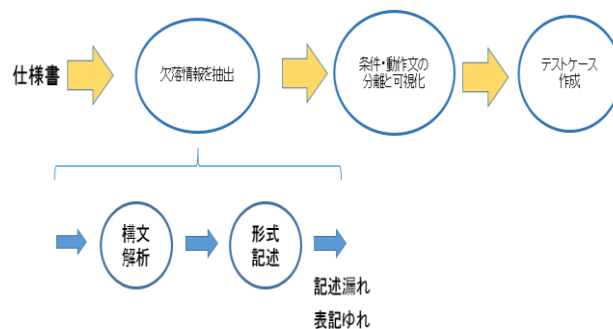


図1 テスト設計プロセス

is(水温, 100度超える) -> 制御する (ポット, ヒータ用電源, OFFに).

図2 セミ形式記述の例

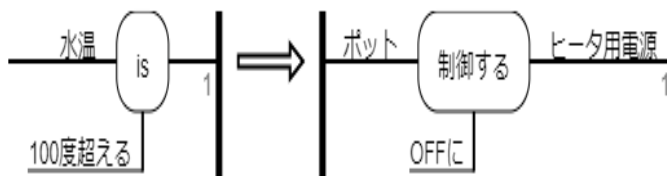


図3 命題ネットワークの例

	0	1
is(水温, 100度超える)	0	1
制御する(ポット, ヒータ用電源, OFFに)	-	1

図4 デシジョンテーブルの例

Development of test case generation support tool from system specification

†Shinryu Murakami †Kyoichi Murakami †Yusuke Aoyama

†Noriyuki Kushiro

†Kyushu Institute of Technology

3. 値域の調査

システム仕様書に記載されている値域を表現する語は、様々な表現で記載されているため仕様書内の文章で値域を表現する語としてどのような表現があるかというのを第1ステップとして調査した。調査方法として、文章をセミ形式記述に変換した際、文章内の値域にあたる表現は制約に記載されるはずである。したがって、ランダムに選定した仕様書20種をセミ形式記述に変換し、その制約に記載されるべき単語を形態素解析器[3]・構文解析器[4]を用いて言語解析を行った。調査の結果、以下の属性が値域を表現する語の判別に必要であることが判明した。

1. 時間：時相名詞，時間辞，時間名詞を含む
2. 強時間：「3時に」などの強い時間表現
3. 数量：数詞，助数辞を含む
4. 未来句：明日，来週，3日後などの未来を表現する
5. 時数ノ：「3時の」「3つの」など
6. 強数量修飾：「このうち3つ」などの数量修飾
7. 回数：「～回」，「～度」などの回数

一方で、例えば「温度が100度以下」のような場合の「以下」は値域を表現するが、「以下のように示す」の「以下」はそれよりも後のことを述べるという意味で使用されている。このように上記の属性を持つ語のすべてが値域を表現しているわけではないため、現状では、テストケース生成を行う前に値域を表現する語を確認する必要がある。

上記作業の支援のため、文章をセミ形式記述に変換したものに値域を表現する語と修飾する語をハイライトで表示し、誤りがあれば修正できるセミ形式記述修正ツールを作成した。

4. セミ形式記述修正ツール

セミ形式記述修正ツールは、VisualStudio Codeの拡張機能として作成した。因みに、橙色のハイライト部分が値域を表現する語、緑色のハイライトの部分がその値域の示す対象となる語を示す。これにより文章中のどの部分に値域を示す語があるかの発見の支援し、値域を表現する語の修飾関係がアルゴリズムにより正しく変換できているかどうかを確認することを容易にする。

このツール上での変換結果を元に自然言語からセミ形式記述への変換アルゴリズムのブラッシュアップを行うことで、自然言語からセミ形式記述の変換アルゴリズムの精度を向上させることが将来的には可能である。

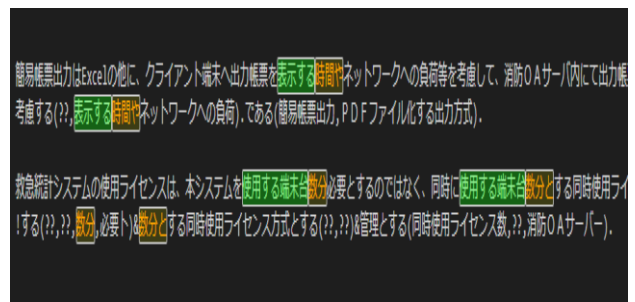


図5 ツールの実行画面

5. まとめと今後

本稿では、システム仕様書記載の文章又は表からテストケースの作成を支援するツールの開発を行った。しかし、現状ではテストケースを生成する際に必要な値域を抽出することが出来ない。そこで、第1ステップとしてシステム仕様書の文章を言語解析し、値域を表現する語として7つの属性の語が選定した。この値域を表現する語の修飾関係の確認を行うために、構文解析した結果を蓄積し、その蓄積した結果を元に自然言語からセミ形式記述へ変換するアルゴリズムのブラッシュアップを行うためのセミ形式記述修正ツールを作成した。

今後は、セミ形式記述修正ツールを使用し、セミ形式記述の変換アルゴリズムのブラッシュアップを行う。そして、セミ形式記述から抽出した値域から同値・境界値法を用いて具体的な値を抽出する。抽出した具体的な値を条件として、デシジョンテーブルに反映することで実行可能なテストケースの生成ができるようにツールの拡張を行う。

参考文献

- [1] 久代紀之，鳥飼諒一．“システムデザインにおける論理的枠組みとデータ活用戦略．” ヒューマンインタフェース学会誌．Vol.17No.2 20.
- [2] 増田聡，松尾谷徹，津田和彦．“テストケース作成自動化のための意味役割付与方法．” 日本ソフトウェア学会．
- [3] 京都大学黒橋・河原研究室，“日本語形態素解析システム JUMAN”
<http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN>
- [4] 京都大学黒橋・河原研究室，“日本語形態素解析システム KNP”
<http://nlp.ist.i.kyoto-u.ac.jp/?KNP>