

## 文章の類似度を用いた影響分析方式の提案

塚本良太<sup>1</sup> 徳本修一<sup>1</sup> 高橋加寿子<sup>1</sup> 磯田誠<sup>1</sup> 中島毅<sup>2</sup>

**概要:** 長期に渡って流用開発を繰り返してきた製品では、要求仕様や設計に関する知識が影響分析の精度に大きな影響を与える。特に開発プロセス上流の設計文書はソースコード等のモデル化された設計書とは異なり、自然言語で記載されているため、経験年数の浅い開発者が影響分析を実施する場合、知識不足により影響範囲の確認漏れが発生するリスクが高くなる。本研究では経験年数によらずに影響範囲の確認漏れを防止するために、文章の類似度による影響範囲の候補抽出方式を開発した。既存のプロジェクトで評価したところ、文字数ベースで54%の工数削減を確認できた。

**キーワード:** 流用開発, 影響分析, 変更要求, 自然言語処理

## A Proposal for a Change Impact Analysis Method Using Sentence Similarity

RYOTA TSUKAMOTO<sup>†1</sup> SHUICHI TOKUMOTO<sup>†1</sup> KAZUKO TAKAHASHI<sup>†1</sup>  
MAKOTO ISODA<sup>†1</sup> TSUYOSHI NAKAJIMA<sup>†2</sup>

**Abstract:** For products that have been repeatedly modified over a long period of time, the accuracy of impact analysis largely depends on knowledge of requirement specifications and design. Because the design documents in the upstream of the development process are written in natural language, when developers with less experience perform impact analysis, there is a high risk of omission of the identification of the affected part due to lack of knowledge. In this study, we developed a method to extract candidate affected part based on the similarity of sentences in order to identify the impact areas regardless of the number of years of experience. We evaluated the method in an existing project, and confirmed a 54% reduction in man-hours based on the number of characters.

**Keywords:** Diversion development, Change Impact Analysis, Change Request, Natural Language Processing

### 1. 背景

近年のソフトウェアを含む製品開発は、既存の成果物(設計書やソースコード)を部分的に修正する流用開発が主流[1]である。流用開発では、利用する成果物の変更による影響分析が重要である。流用する製品の既存機能、アーキテクチャおよび検証結果の確認が不足すると、想定外の障害や性能悪化などの品質低下を引き起こす場合が多い。

また、長期に渡って流用開発を繰り返してきた製品では、要求仕様や設計に関する知識が影響範囲を特定する精度に大きな影響を与える。特に開発プロセス上流の設計文書はソースコード等のモデル化された設計書とは異なり、自然言語で記載されているため、経験年数の浅い開発者が影響分析を実施する場合、知識不足により影響範囲の確認漏れが発生するリスクが高くなる。

個々の要求仕様に識別子を付け、関連する箇所にタグ付けを行うことによって、要求のトレーサビリティを記録し、影響分析を支援する手法もある[2]。一方で、自然言語で記載された仕様間の関連は、ソースコードのように容易にモデル化ができないため、明確には定義できない。また、長

期に渡って流用開発を繰り返してきた製品では、タグ付けに抜けや誤りが入り込む可能性が高くなる。

そのため、開発者の経験の多寡によらず、従来のタグ付けをせずに、影響範囲を特定する方法が必要である。我々は流用開発における変更要求に対し、自然言語で記載された設計書からソースコードに至る抜け漏れの無い影響分析を支援するツールを提案する。このツールにより後工程の手戻りリスクが減り、開発コストを削減できると考える。また、このツールを活用することで影響分析自体の工数、設計レビューの工数などの削減や品質の向上も期待できる。

### 2. 課題

設計書における影響分析は一般的には要求トレーサビリティを利用する。上流工程の成果物において、要求仕様や機能に識別子を付け、下流工程の成果物においてその識別子を参照することで要求トレーサビリティを設定する。そうしておくことで、上流工程の成果物に変更が生じたときに下流工程の成果物の同一識別子を探すことで変更すべき箇所を特定することができる。

しかし、現実の開発では前述の通り識別子に抜けや誤り

1 三菱電機株式会社  
Mitsubishi Electric Corporation  
2 芝浦工業大学

Shibaura Institute of Technology

が入り込むリスクが有り、維持にコストをかけている。一般的には成果物の内容を確認し、開発者の知識を頼りに関係すると考えられる箇所を独自に確認し、レビューなどで抜け漏れが無いことを再確認する。この開発者の知識によるレビュー方法は、経験の浅い開発者が担当した場合、抜け漏れのリスクは大きく、リスクを減らすための有識者とのコミュニケーションコストも大きい。

本稿では、開発者の経験年数によらず、従来のタグ付けによる方法を補完するような影響分析の方法として、文章の類似度に基づく影響分析方式を提案する。

### 3. 影響分析方式

提案方式について以降で示す。提案する方式は成果物内の文章に書かれている内容から影響範囲を特定する。

#### 3.1 前提条件

本方式の前提条件を以下に示す。

##### (1) 成果物の品質

影響分析の対象とする成果物間に仕様の不整合といった誤りは無く、品質は保証されていると仮定する。文章の類似度を基礎としているため、記述に不整合がある場合は本方式の結果も誤った内容となる可能性があるためである。

##### (2) 成果物の記載内容

下流工程の成果物ではまず、上流工程の成果物の説明を記載するケースが多く、そうでない場合でも固有名詞等の重要な単語は揃えられていると想定する。

##### (3) 成果物の記述構成

設計書等の技術文書は構成等が標準化されている。例えば機能に関する記述は見出しの深さ3のセクションにあるなど決まった範囲に記載される。ここでいう深さは章節項などの見出しのレベルである。

##### (4) 開発プロセスと成果物の対応

開発プロセスとして要求分析、システム設計、ソフトウェア設計等があり、それぞれの成果物として要求仕様書、システム設計書、ソフトウェア設計書等が存在する場合のプロジェクトを対象としている。

#### 3.2 基本方式

本方式はプロジェクトの開発プロセスにおける成果物毎に影響分析を実施する。図1にその方式を示す。入力として対象プロセス=要求分析と変更要求文章を入力した場合の処理を示している。出力となる影響範囲候補一覧には要求仕様書の表示単位(3.3にて説明)が類似度順にリストアップされる。

本方式では事前分析が必要である。まず、全ての成果物は事前にベクトル化し、ベクトル化のためのモデルも生成しておく(3.5にて説明)。

入力は対象とする成果物が作成されたプロセスとそのプロセスの成果物に対する変更要求を示す文章である。出

力は指定したプロセスの成果物に記載されている内容から表示単位毎に影響範囲候補としてリストを表示する。このリストは成果物を表示単位に分割し、表示単位毎に変更要求を示す文章と類似度を計算し(3.4にて説明)、類似度の高い順にソートすることで生成する。

対象プロセスを指定する方式としたのは、表示された候補を開発者が確認する観点を揃えるためである。対象プロセスを指定しない方式の場合、全成果物から影響範囲の候補を抽出することになる。その結果、要求仕様やソフトウェア設計の内容が混在して一覧化されることになる。この場合、開発者は要求仕様の観点での確認やソフトウェア設計の観点での確認を頻りに切り替えなければならなくなる。これは影響範囲を確認する効率が悪いと考えた。

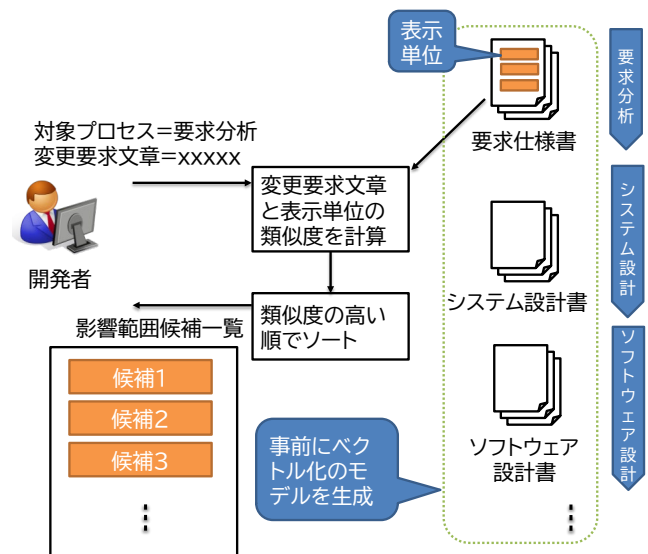


図1 基本方式

Figure 1 Basic method of impact analysis

#### 3.3 表示単位

本方式では影響範囲の候補として表示される範囲は開発者がパターンを選択する方式としている。こういった単位は機械的に検出することも考えられるが、今回は前提条件に記載したとおり、記述構成が標準化されている前提で以下のパターンのみを扱うこととした。

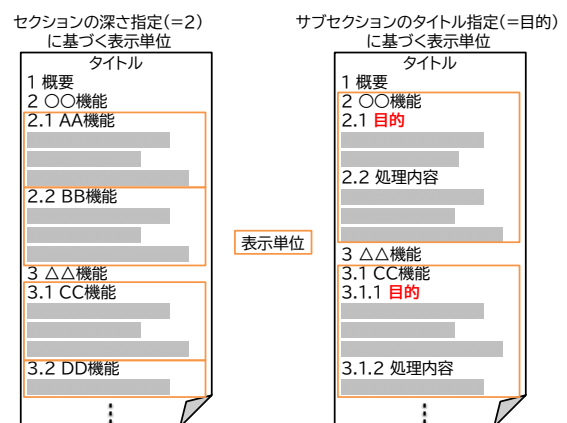


図2 表示単位

Figure 2 Display unit

(1) セクションの深さ指定

一定の深さのセクションを表示単位とする。例えば図 2 の左では深さを 2 と指定した場合、1.x, 2.x, 3.x 等のセクションが表示単位として指定される。

(2) サブセクションのタイトル指定

特定のタイトルのサブセクションを持つセクションを表示単位とする。例えば図 2 の右では「目的」を指定した場合、「1.x.y 目的」をサブセクションに持つ 1.x 等のセクションが表示単位として指定される。

3.4 表示単位の類似度計算方式

変更要求を示す文章と表示単位との類似度を計算する手順を図 3 に示す。変更要求の文章は  $n$  個の段落で構成されており、各段落を  $\{R_i | i = 1, 2, \dots, n\}$  とする。表示単位の文章は  $m$  個の段落で構成されており、各段落を  $\{D_j | j = 1, 2, \dots, m\}$  とする。

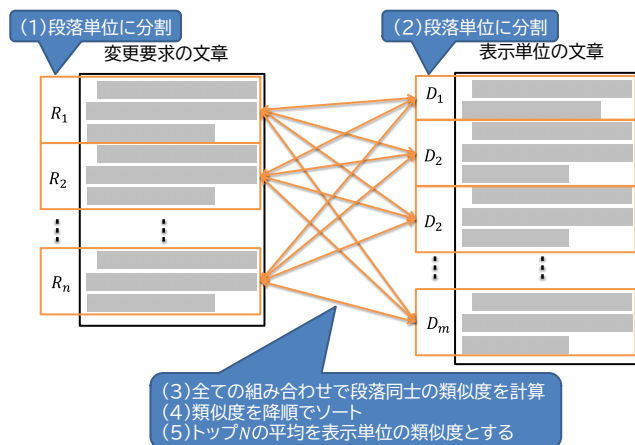


図 3 表示単位の類似度計算手順

Figure 3 Display unit similarity calculation procedure

(1) 変更要求を示す文章を段落単位に分割

入力された文章を段落に分割する。改行を段落の区切りとして分割する。空行は無視する。

(2) 表示単位の文章を段落単位に分割

成果物を解析し得られた表示単位を段落に分割する。段落の区切りは成果物のファイル形式に応じて表 1 のようにする。空行は無視する。また、一般的なプロジェクトでは Microsoft Excel<sup>a</sup> による成果物も存在する可能性があるが、後述の評価で用いた題材 (4.1 を参照のこと) には存在しなかったため、今回は定義しない。

表 1 ファイル形式に応じた段落の扱い

Table 1 Handling of paragraphs according to file format

ファイル形式	段落
Microsoft Word <sup>b</sup>	改行区切り
HTML	<p>タグ内

<sup>a</sup> Microsoft Excel は Microsoft Corporation の米国、その他の国での登録商標である。

<sup>b</sup> Microsoft Word は Microsoft Corporation の米国、その他の国での登録商標

(3) 変更要求の段落と表示単位の段落同士で

類似度を計算

変更要求の文章の  $n$  個の段落と表示単位の文章の  $m$  個の全ての組み合わせ  $\{R_i \cdot D_j | i = 1, 2, \dots, n, j = 1, 2, \dots, m\}$  を求める。ここで「 $\cdot$ 」は段落単位の類似度を計算することを意味する (段落単位の類似度の計算方法は 3.5 を参照のこと)。

(4) 段落単位の類似度について降順でソート

上記で求めた全ての  $i, j$  の組み合わせの類似度  $\{R_i \cdot D_j | i = 1, 2, \dots, n, j = 1, 2, \dots, m\}$  を類似度の高い順でソートする。

(5) 表示単位の類似度を計算

上記でソートした類似度のトップ  $N$  の平均を表示単位の類似度とする。 $N$  は平均を取るための母数であり、評価で調整する。

以上の方法で表示単位の類似度を計算するが、変更要求を示す文章と表示単位を分割する単位として、段落以外に文や文章全体等の選択肢が考えられる。変更要求数件を対象とした簡易評価では段落単位で分割する方法が最も良い結果であった (類似度の高い表示単位が実際の影響範囲であった) ため段落単位での分割とした。

3.5 段落単位の類似度計算方式

段落のベクトル化として PV-DM (Distributed Memory Model of Paragraph Vectors) を使用し、類似度の計算はベクトル間のコサイン類似度を使用した。実装は gensim<sup>d</sup> (Ver.3.8.3) の Doc2Vec を用いた。

4. 評価方法

上記で説明した影響分析方式を評価するための題材、指標及び具体的な評価手順について示す。

4.1 評価題材

組込み系プロジェクトの要求仕様と要求仕様に対する変更要求を用いた。このプロジェクトは毎年前年度の成果物をベースに開発を重ねている典型的な流用開発のスタイルである。

表 2 評価題材とした機種及び変更要求数

Table 2 The model used as the evaluation subject and the number of change requests

機種	変更要求数	要求仕様に変更が発生した変更要求数
機種 A	10	3
機種 B	24	15
機種 C	53	37
機種 D	20	9

である。

<sup>d</sup> <https://radimrehurek.com/gensim/>

機種数, 変更要求数及び実際に要求仕様に変更が発生した変更要求の数を示す. また, 各機種の要求仕様の規模を示す.

表 3 機種毎の要求仕様の規模

Table 3 Scale of required specifications for each model

機種	ファイル数	総ページ数	総文字数
機種 A	16	570	483,160
機種 B	16	599	505,392
機種 C	16	654	549,964
機種 D	16	678	569,744

## 4.2 評価指標

提案した影響分析方式を評価する指標として, レビューアの確認工数面と精度面の観点で説明する. また, これらの指標を左右する調整パラメータについても説明する.

### 4.2.1 工数面の指標

工数面は影響分析を行う際に読まなければならない文章範囲を文字数で計測する. 理想は予備知識の無い開発者に影響分析を実施してもらい, その時間を計測することであるが, 予備知識の無い人員確保及び時間計測が困難であるため文字数での計測とした.

### 4.2.2 精度面の指標

精度面では網羅範囲比率と網羅範囲内正答率を使用する[3]. 図 4 はその説明図である. 影響範囲の候補一覧が全体で $W$ 個あり, 上位から $C$ 番目までに変更すべき箇所が全てリストアップされているとする.  $C$ 番目までには変更すべきでない箇所(不正解)も含まれており, 変更すべき箇所(正解)は $T$ 個含まれる.

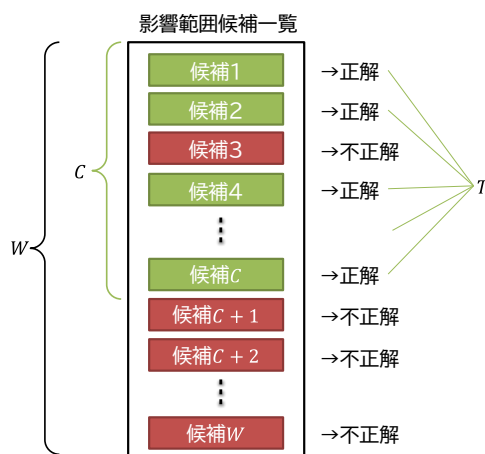


図 4 精度指標の定義

Figure 4 Definition of accuracy index

### (1) 網羅範囲比率 $F$

影響範囲の候補一覧があった場合に, 上位何%迄にすべての正解が含まれるかの割合である. 以下の式で表される. 低い値が良い値と言える.

$$F = \frac{C}{W}$$

### (2) 網羅範囲内正答率 $A$

上記網羅範囲の中に正解がどれくらいの割合で含まれているかを表す. 以下の式で表される. 高い値が良い値と言える.

$$A = \frac{T}{C}$$

## 4.3 評価手順

本方式の具体的な評価手順を以下に示す.

### (1) 調整パラメータの設定

調整パラメータを設定する. パラメータ一覧は 4.4 を参照のこと.

### (2) 評価する機種の設計書一式の事前解析

設計書一式を表示単位に分割する. また, 段落単位の分割と形態素解析も行い, Doc2Vec の学習モデルを生成する.

### (3) 本方式による影響範囲の候補一覧を出力

影響分析の対象とする開発プロセスに要求仕様, 変更要求を示す文章に変更要求を入力し, 影響範囲の候補一覧を出力する.

### (4) 上位から全ての正解が現れるまでの各種計測

上位から順に実際の影響範囲かどうか, 正解データと突き合わせながら計測を進める. 計測するのは表示単位に含まれる総文字数である. 全ての正解データが出揃うまで進める. 全ての正解データが出揃った順位までの総表示単位数の全候補に対する割合が網羅範囲比率である. 網羅範囲までに計測した文字数の合計を求める. この合計の文字数が影響分析のために読まなければならなかった文字数である. また, 網羅範囲までに現れた正解の割合が網羅範囲内正答率である.

### (5) 調整パラメータの最良値計測

上記を繰り返し, 網羅範囲比率と網羅範囲以内正答率が最も良い値となる調整パラメータを求める. 本実験では機種 B の変更要求で求めた. 最良値が計測できたら全ての機種全ての変更要求について読まなければならない文字数, 網羅範囲比率, 網羅範囲内正答率を計測する.

## 4.4 調整パラメータ

本方式で調整可能なパラメータは以下の通り.

### (1) 表示単位の類似度の集計単位

3.4 にて説明した表示単位の類似度を集計する母数 $N$ によって, 結果が変わると考えられるため, その値を調整パラメータとする. 実験では 10 と 30 で評価した.

### (2) 表示単位の設定

表示単位として設定するセクションの深さが異なると, 表示単位の類似度の計算結果も異なることが考えられるため, その値を調整パラメータとする. 実験では深さ 2 のセクション (1.x, 2.x, 3.x 等), 深さ 3 のセクション (1.x.y, 2.x.y, 等), サブセクションに「目的」を持つセ

クシオンで評価した。

### (3) 形態素解析の辞書

文章をベクトル化する際に形態素解析を行っているが本方式では Janome<sup>e</sup> (Ver.0.4.1, MeCab<sup>f</sup>と同等) を用いている。MeCab には固有名詞を充実化させるための辞書読み込み機能が備えられており、Web 上の固有名詞を集めた辞書も有志によって公開されている。固有名詞の区切り方はベクトルの値に影響し、類似度の計算にも影響すると考えられるため調整パラメータとする。実験では MeCab 標準の場合（追加で辞書を設定しない）と NEologd<sup>g</sup> (Ver.0.0.7) の辞書を追加する場合で評価した。

### (4) Doc2Vec の次元数

文章のベクトルの次元数が異なると、類似とみなされる文章にも違いが現れると考えられるため、その値を調整パラメータとする。実験では 100 と 300 で評価した。

### (5) Doc2Vec の学習回数

Doc2Vec はその実装上乱数を用いてベクトル化を行っており、学習回数を増やすことによってその精度を上げることができる。このベクトル化の精度の変化は類似度の計算に影響を与えるため調整パラメータとする。実験では 5 と 50 で評価した。

## 5. 評価結果

調整パラメータの最良値、工数面の評価、精度面の評価について説明する。

### 5.1 調整パラメータの最良値

調整パラメータの最良値を表 4 に示す。

表示単位の類似度の集計単位は 10 よりも 30 とした方が精度指標の結果が良いことが分かった。10 程度では影響範囲と関係しないが類似とみなされる段落が多く混入してしまった可能性が考えられる。よって表示単位の類似度の集計単位はある程度大きい値が良いと考えられる。

表示単位の設定はサブセクションに「目的」を持つセクションが精度指標の結果が良いことが分かった。この表示単位は対象の仕様書の文書構成において機能単位に相当する単位である。そのため、表示単位はセクションの深さ等の単純な決め方よりは機能単位を示すセクション等、ある程度仕様として意味のある単位を指定するのが良いと考えられる。

Doc2Vec の次元数は 300 よりも 100 とした方が精度指標の結果が良いことが分かった。設計書に含まれる語彙は記事分類など一般の自然言語処理で必要とされる語彙より少なく、小さめのベクトル空間で十分であると考えられる。

Doc2Vec の学習回数は 50 よりも 5 とした方が精度指標の結果が良いことが分かった。Doc2Vec の次元数と同様に設計書に含まれる語彙が少ないため、語彙間のベクトルは

十分に分散しており、ベクトル化の誤差の影響が少なかったと考えられる。

形態素解析の辞書は NEologd の辞書よりも MeCab 標準の辞書の方が精度指標の結果が良いことが分かった。設計書に記載される機能名等の固有名詞は MeCab 標準辞書では分割される傾向にあり、NEologd の辞書では連結される傾向にあるが、文章間の類似度を計測する上では分割された状態の方が良い結果を得られることが分かった。

表 4 調整パラメータの最良値

Table 4 Best adjustment parameter

調整パラメータ	最良値
表示単位の類似度の集計単位	30
表示単位の設定	サブセクションに「目的」を持つセクション
Doc2Vec の次元数	100
Doc2Vec の学習回数	5
形態素解析の辞書	MeCab 標準

### 5.2 工数面の評価

本方式で影響分析を行った場合に読まなければならない文章範囲の文字数を評価した結果を表 5 に示す。全体の平均としては 54%の工数削減効果であった（全ての文章を読むのに対して 46%を読めば全ての影響範囲を特定できる）。

表 5 工数削減効果

Table 5 Man-hour reduction effect

機種	母数	総文字数	網羅範囲文字数 (平均)	工数削減効果
機種 A	3	483,160	324,151	0.67 (37%削減)
機種 B	15	505,392	192,248	0.35 (65%削減)
機種 C	37	549,964	250,395	0.50 (50%削減)
機種 D	9	569,744	257,048	0.45 (55%削減)

まず、機種 B が最も良い結果が得られている。これは調整パラメータを最適化したデータであったためと考えられる。機種 C と機種 D については機種 B ほどではないが読む文章量が半減しておりある程度良い結果が得られていると考えられる。機種 A については最も悪い結果となっているがサンプル数が 3 であったため本結果だけでは必ずしも悪い結果になると結論付けられないと考えられる。

### 5.3 精度面での評価

網羅範囲比率と網羅範囲内正答率を計測した結果を表 6 に示す。全体の平均としては網羅範囲比率が 0.33、網羅範囲内正答率が 0.14 であった。

まず、機種 A が最も良い網羅範囲比率を得られているが工数面での評価同様にサンプル数が少ないため本結果だけ

<sup>e</sup> <https://mocabeta.github.io/janome/>

<sup>f</sup> <https://taku910.github.io/mecab/>

<sup>g</sup> <https://github.com/neologd/mecab-ipadic-neologd>



では良い結果が得られると結論付けられないと考えられる。機種 B と機種 C については本方式で得られる影響範囲の候補の上位 30~40%程度を調査すれば影響範囲を網羅できる結果となっている。機種 D については網羅範囲比率と網羅範囲内正答率の両方について良い結果が得られている。機種 D のみ良い結果が得られている原因の分析として、設計書の記載内容や変更要求の記載内容を分析してみたが他と区別できるような特徴は見つけられなかった。機種 D の結果の解釈についてはより詳細な分析が必要と考えられる。

表 6 精度指標の計測結果

Table 6 Performance index measurement results

機種	母数	網羅範囲比率	網羅範囲内正答率
機種 A	3	0.09	0.11
機種 B	15	0.33	0.11
機種 C	37	0.37	0.12
機種 D	9	0.23	0.27

## 6. 今後の課題

評価結果から次のことが今後の課題と考えられる。

### (1) 変更要求の文章と対象成果物の文章の特徴の分析

本評価では工数面での有用性や精度指標などは計測できたが、今後の改善に有用となる変更要求の文章と対象成果物の文章の構成・記述内容による結果の特徴の分析はできなかった。これは文章のベクトル化の方法をブラックボックスとして扱ったためであり、今後はベクトル化の方式も踏まえた実験データの収集や結果の考察が必要である。

### (2) 機種個別の最適化

本評価では工数削減効果においては、調整パラメータの最良値を求めた機種が最も良い結果を得られた。このことから機種ごとに最良パラメータが異なる可能性が考えられ、その最適化方法を検討する必要がある。

### (3) 組込み系以外のプロジェクトでの評価

本評価は組込み系プロジェクトの成果物を対象に評価を行った。エンタープライズ系プロジェクトにおいても同様の結果が得られるとは限らないため評価が必要である。

### (4) ユースケースの拡充

本評価は上流工程の成果物の変更に対し下流工程の成果物の影響範囲を分析する想定で実施している。しかし、実開発ではソースコード等下流工程の成果物の変更を特定してから上流工程の成果物を修正するユースケースも多く存在する。これらのユースケースの整理と本方式の再評価が必要である。また、本方式の特性に応じた効果的な利用方法の模索も必要である。

### (5) 実開発での使用データの収集

本評価は過去の開発データを用いて実施している。実

開発で同等の効果が得られるか検証する必要がある。

## 7. 関連研究

影響分析は、既存機能に対する変更が他の機能等に及ぼす影響を確認することである。一般的には設計執筆時に上流の要求とのマッピングをトレーサビリティマトリックス等に記録しておき、影響分析が必要になったときにそのマッピングを参照し、影響範囲を特定する。ツールなどでは成果物にタグ付けをし検索可能な状態にして影響分析を支援する。代表的なツールとしては Reqtify[4] や microTRACER[5]等がある。また、タグ自体を構造化して影響範囲の特定を支援する方法も研究されている[2]。これらタグ付けを中心とした方法は本方式を補完する方法であり、組み合わせると効果が上がると考えられる。

一方で、変更要求と本文との類似性に着目した研究は単語レベルではこれまでに実施しており、その有用性を示してきた[6]。

## 8. まとめ

本稿では、開発者の経験年数によらず、従来のタグ付けによる方法を補完するような、影響分析方法として文章の類似度に基づく影響分析方法を示した。

組込み系のプロジェクトを題材に評価を実施したところ、工数面では設計書を読む量に換算して平均で 54%の削減効果があることが分かった。また、精度面では平均で網羅範囲比率は 0.33、網羅範囲内正答率が 0.14 であり、リストアップした影響範囲の候補の上位 3 割程に全ての変更すべき箇所がリストアップされており、工数面と合わせて本方式の有用性を確認できた。

今回の実験結果からは変更要求の文章と対象成果物の文章の特徴分析等、課題をいくつか抽出することが出来ており、今後はそれらの改善に取り組む。

## 参考文献

- [1] 社会基盤センター, ソフトウェア開発データ白書 2018-2019, (独)情報処理推進機構, 東京, 2019
- [2] 徳本修一, 高橋洋一, 田村直樹, 中島毅, “構造化タグ方式による要求トレーサビリティリンク構築ツール”, 電子情報通信学会論文誌 D, Vol.J104-D, No.2, pp.105-118, 2021
- [3] 岩崎陽也, 中島毅, “過去の変更仕様書を学習することにより要求文からプログラム変更部分の候補を推薦するシステムの提案.” 研究報告ソフトウェア工学 (SE) 2021.16 (2021): 1-6.
- [4] ダッソー・システムズ, Reqtify, <https://www.3ds.com/ja/products-services/catia/products/reqtify/>, April 2020.
- [5] DTS インサイト, microTRACER, [https://www.dts-insight.co.jp/product/eps\\_tool/microtracer/index.html](https://www.dts-insight.co.jp/product/eps_tool/microtracer/index.html), April 2020.
- [6] 高橋 加寿子, 塚本 良太, 磯田 誠, “変更要求に対するシステム設計書修正箇所抽出法の検証”, 研究報告ソフトウェア工学 2018-SE-200: 1-6