

## 要件定義書からの ファンクションポイント自動計測の試み

山田涼太 山田悠斗 楠本真二 栢本真佑 肥後芳樹  
大阪大学大学院情報科学研究科

{r-yamada, y-yuto, kusumoto, shinsuke, higo}@ist.osaka-u.ac.jp

### 要旨

一般にソフトウェア開発プロジェクトの見積りでは、まず開発規模が見積もられ、それをを用いて工数や予算の見積りが行われる。開発規模の尺度として、最近ではファンクションポイント (FP) の利用が重要視されている。一方で、FPを導入する上での幾つかの課題があり、それらの対策として計測の自動化が期待されている。

本稿では、開発の上流工程で作成される要件定義書を対象とした、FP自動計測の試みについて報告する。提案手法では、要件定義書のうち機能仕様についての記述を提案するテンプレートを用いて書き換え、書き換えられた仕様を解析し、計測に必要な要素を抽出することでFP計測を行う。4種類の小規模な要件定義書を対象に提案手法を適用し、手動での計測と比較することで精度を確認した。その結果、3つの要件定義書では手動計測と同様の値が、残りの1つの要件定義書では手動計測より若干大きな値が計測された。

### 1. はじめに

一般に、ソフトウェア開発においては初期段階でのソフトウェア開発規模の見積りが重要である [1]。不正確な見積りはプロジェクト失敗の原因になることも報告されている [2]。正確な見積りの実現を目指し、ソフトウェア開発規模の見積りに関する研究が盛んに行われている。

見積りの基となる開発規模の尺度としてはソースコードの行数がよく使われていたが、最近ではファンクションポイント (FP) [3] の利用が重要視されている。政府情

報システムの開発においても、要求内容に設計または開発に関する工程が含まれる場合には、原則FPの見積り及びその根拠を示すことが必須となりつつある [4]。その一方、FPの導入における様々な課題のため、ベンダー・ユーザ企業ともにFPは十分に普及していない。1つ目の課題は、見積りのための基礎データが必要であることである。新規プロジェクトに対するFPを計測したとしても、その値を工数に変換するためには、ある程度、過去プロジェクトのFP値と工数の情報が必要となる。次に、計測者の判断による誤差が生じる点である。詳細な部分のFPの計測には測定者の判断が必要となる。同一のソフトウェアからのFP計測であっても、計測者によって誤差が生じてしまうという問題点が指摘されている [5]。例えば、同じ組織の人間が同じアプリケーションに対して測定した場合は12%、違う組織の人間が即していた場合は30%以上の誤差が出るという報告もされている [6]。最後に、導入教育が必要である。FP法を用いるためには、FP計測方法や、見積りに関する知識を身につけなければならない。それらの課題の1つの対策として、計測の自動化が考えられる。

FP計測の自動化に関する研究としては、UML(Unified Modelling Language) 図で記述された設計図からの計測 [7, 8]、Webアプリケーション・プログラムからの計測 [9, 10] 等があり、幾つかの制約を置いた上である程度正確なFP値の計測が可能となっている。しかし、開発の上流工程で作成される主に自然語で記述された要件定義書を対象としたFP自動計測手法は著者らの知る限りは存在しない。

本研究では、要件定義書 [11] を対象とした、FP自動計測手法について検討する。まず、小規模な自然語で書かれた仕様に対して、仕様がどのように記述されてい

ば計測が可能かを確認・調査する。その結果に基づいて、自然語記述からのFP計測方法を提案する。提案手法では、機能仕様をFP計測のためのテンプレートを用いた書き換え、書き換えられた仕様を解析してFPを計測する。更に提案手法に基づいた計測ツールを試作し、様々なプロジェクトの仕様書で検証することで評価を行った。以降、2章では研究の背景となる諸用語や関連研究について述べる。3章では今回提案する自動計測の手法について述べる。4章では実際に自動計測を行った結果と考察について述べる。最後に5章で本報告のまとめを述べる。

## 2. 準備

本章では研究の背景となる諸用語や関連研究について簡単に述べる。

### 2.1. ファンクションポイント法

ソフトウェアの規模を見積もる手法の1つにファンクションポイント(FP)法[3]がある。FP法は、ユーザから見た機能の量を計測する手法で、A.J.Albrechtによって1979年に提案された。画面や帳票、ファイルなどを通じた情報の入出力に着目し、それらを種類別に数え上げ、種類数を加重合計した値を機能量としている。以降、IFPUG法[12]、MarkII法[13]、COSMIC法[14]等、様々なFP法が提案されてきている。本研究ではIFPUG法を対象とし、以降、FP法とはIFPUG法を意味することとする。IFPUG法は、Albrecht版の使い勝手や曖昧な部分を改良したバージョンであり、ビジネスアプリケーションソフトウェアを対象として欧米で広く使用されている。IFPUG法では、ソフトウェアの持つ機能から5種類の基本機能要素を以下に示す[15]。

#### 内部論理ファイル(ILF)

計測対象のアプリケーション内でデータが更新される論理的な関連を持ったデータの集合

#### 外部インターフェイスファイル(EIF)

計測対象のアプリケーションによってデータが参照されるデータの集合(データは更新されない)

#### 外部入力(EI)

計測境界外からのデータ入力によってILFの更新を行う処理

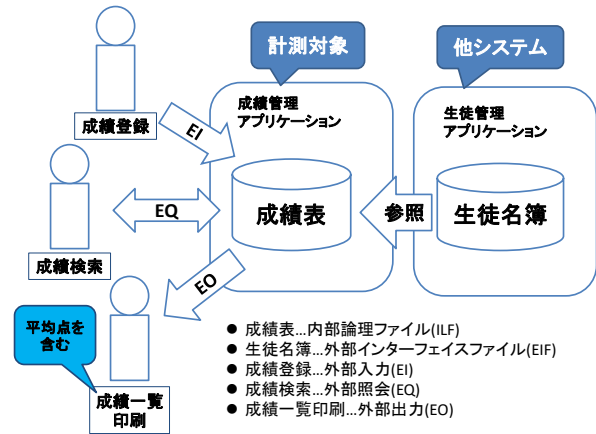


図 1. 基本機能要素の例

#### 外部出力(EO)

計測境界外へのデータ出力を含む処理のうち、出力データに導出データを含むもの

#### 外部照会(EQ)

計測境界外へのデータ出力を含む処理のうち、出力データに導出データを含まないものであり、処理がILFを更新しないもの

例として、図1に学生の成績管理アプリケーションとそこに含まれる基本機能要素を示す。計測対象である成績管理アプリケーション内でデータが更新される成績表はILF、成績管理アプリケーション内にはなく、成績管理アプリケーションから参照される生徒名簿はEIFとなる。また、データ入力によって成績の登録を行う成績登録処理はEI、成績表から平均点などの派生データを含まないデータを出力させる成績検索はEQ、成績表から平均点などの派生データを含むデータを出力させる成績一覧印刷はEOとなる。上記の5種のうち、ILFとEIFはデータファンクション(以降、DF)、EIとEQとEOはトランザクションファンクション(以降、TF)と呼ばれ、それぞれ計測に必要な要素が異なる[15]。

#### ● DF

- － 種別…ILF, EIF
- － DET(Data Element Type)…アプリケーション上で参照される項目の数
- － RET(Record Element Type)…データ群の属性内のサブグループの数

- TF
  - 種別…EI, EQ, EO
  - DET…重複を除いた入出力項目の数に、該当処理において下記のものがあれば1ずつ増やしたもの
    - \* きっかけ(処理が行われるトリガー、ボタンなど)
    - \* メッセージ(エラーメッセージなど)
  - FTR(File Type Reference)…TFに関わるDFの数

上述した要素を用いて各機能の複雑度を計測し、その上でFPの値を求める。

- DFの複雑度

表1を参照しDFの複雑度を決定する。例えばRETが1、DETが26のILFがあった場合、表1より複雑度は低となる。

- TFの複雑度

EI, EO, EQのそれぞれに対して用意されている対応表を用いて、TFの複雑度を決定する。例えば、EIは、表2を参照し複雑度を求める。FTRが2、DETが14のEIの場合、複雑度は中となる。

- 複雑度からのFP計測

抽出された基本要素の複雑さ毎の個数を表3に代入し、それぞれに設定されている重み数を掛けて足し合わせた値がFPとなる。前述の例であれば、複雑度「低」のILF(RET:1, DET:26)であれば重みは10となる。また、複雑度「中」のEI(FTR:2, DET:14)であれば重みは4となる。

## 2.2 自然言語解析ツール cabocha

cabochaはSupport Vector Machines(SVM)に基づく日本語係り受け解析器である[16]。入力として与えられた文中の各単語の品詞と各文節の係り先を取得する。

図2に解析例を示す。矢印はその文節の係り先を表している。図2からは、『このデータ群は、』という文節は『データ群です。』に係り、『依頼番号と』という文節が間接的に『持つ』という文節に係っていることが分かる。

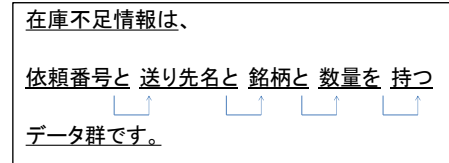


図 2. cabocha による解析例

## 3. 提案手法

本章では今回提案する計測手法について述べる。

### 3.1. 提案するFP計測の流れ

2. で述べた通り、FP法はユーザから見た機能の量を計測する手法であるため、要件定義書に記載されている情報を用いることで、計測することは可能である。しかし、一般には、必要な情報が省略されている場合も考えられ、必ずしも全ての情報が記載されているとは限らない。そこで、幾つかの小規模な仕様書に対して、どの程度詳細に記述されていれば、自動計測が可能かどうかを確認した。その結果、以下のような仮説を仮定した。

**A1** DF計測の基となるアプリケーションで扱われるデータは、アプリケーションが行う処理(機能)での入力や処理の中で使われる項目として表れている。また、表の形で記載されることもある。

**A2** DFにおいて、DFが持つ項目についての情報は仕様明記されている必要がある。

**A3** DFの種別は、TFにおいて参照するDFが明記されていれば更新されるか否かが分かる。

**A4** DFのDETは、TFにおいて参照するDFと入出力項目が明記されていれば特定することができる。

**A5** TF計測の基となるアプリケーションが提供する機能は、処理を表す文章や図を用いて表現されることが多い。

**A6** TFの種別は、仕様に含まれた特定の用語および出力項目と参照するDFから判別できる。

**A7** TFにおいて、全てのTFに「きっかけの有無」が存在する。

表 1. ILF と EIF の複雑度

RET \ DET	1-19	20-50	> 50
1	低	低	中
2-5	低	中	高
> 5	中	高	高

表 2. EI の複雑度

FTR \ DET	1-4	5-15	> 15
0-1	低	低	中
2	低	中	高
> 2	中	高	高

表 3. FP の算出のための計算表

	低	中	高	合計
ILF	$\square \times 7 = \square$	$\square \times 10 = \square$	$\square \times 15 = \square$	
EIF	$\square \times 5 = \square$	$\square \times 7 = \square$	$\square \times 10 = \square$	
EI	$\square \times 3 = \square$	$\square \times 4 = \square$	$\square \times 6 = \square$	
EO	$\square \times 4 = \square$	$\square \times 5 = \square$	$\square \times 7 = \square$	
EQ	$\square \times 3 = \square$	$\square \times 4 = \square$	$\square \times 6 = \square$	
			FP	

A8 TFにおいて、DETの根拠となる「入出力項目」「メッセージの有無」およびFTRの根拠となる「関わるDF」についての情報は仕様で明記されている必要がある。

以上の仮説に基づいて、以下の4つのステップを提案する。また、概要を図3を示す。

1. 要件定義書における機能を説明する部分の文章を自動計測用テンプレートに合わせて計測用の仕様へと変換する。
2. 変換した仕様から計測に必要な要素を抽出する。
3. 解析結果を用いてFPを計測する。

### 3.2. 提案手法の前提

今回の提案手法は、3.の最初で述べた小規模な仕様書に対する試行結果に基づくものである。いわゆる在庫管理等のビジネスアプリケーションに関する要件定義書が対象となる。また、要件定義書では、機能を説明する部分の文章において以下の条件を満たすものとする。

- 主語、述語、目的語が整った誤解の生まれない文であること。

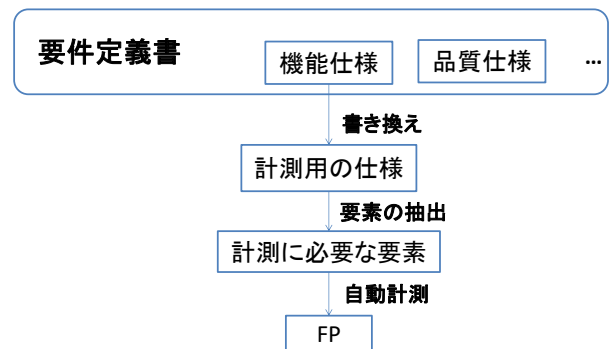


図 3. 提案する FP 計測手法の概要

- 一般には図表参照と記載されている部分について、文章で記述されていること。

図4にて例を示す。例えば「ログインは図4の画面を用いて行う。」と記載されている場合、「ログイン画面では、名前とパスワードを入力し、ログインボタンを押すことでログインを行う。」と図を参照せずに、文章として記載する。

### 3.3. テンプレートを用いた書き換え

計測には大きく分けて、DFとTFの計測が必要であり、それぞれの計測において必要な要素が存在する。そ

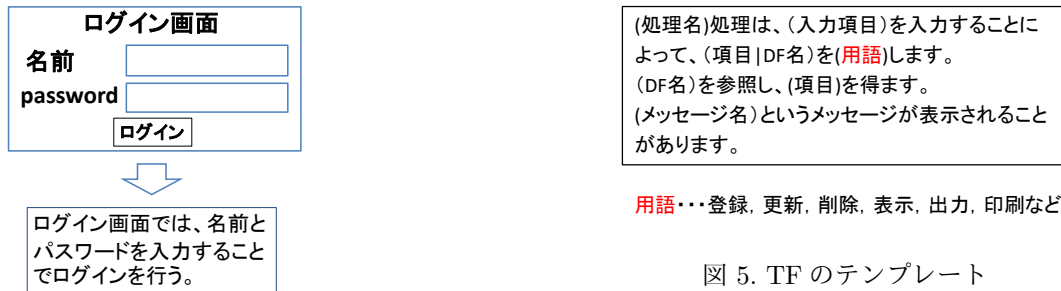


図 4. 図から文章への変換

これらの要素を持った仕様であれば、理論的には自動計測は可能であるが、前述の「主語、述語、目的語が整った誤解の生まれない文」であっても自由度が存在するため、今回はテンプレートを用いて書き換えを行う。

### 3.3.1 TF のテンプレート記述

TF に必要な要素は以下の通りである。

- 種別…EI, EQ, EO
- DET…入出力項目の数(重複は除く)+1(きっかけの有無)+1(メッセージの有無)
- FTR…関わる DF の数

今回の計測では全ての TF にきっかけがあると仮定しているので、テンプレートに沿って書き換えられた文章から FP 計測を行うためには、「種別」「入出力項目」「メッセージの有無」「関わる DF の数」といった情報が得られなければならない。計測に必要な要素を含み、かつ判別の容易な文章を作るためのテンプレートが図 5 である。

図 5 のテンプレートに変換する際、処理に入力項目や参照、メッセージが存在しない場合はそれぞれ「(入力項目)を入力することによって、」や「(DF 名)を参照して、(項目)を得ます。」「(メッセージ名)というメッセージが表示されることがあります。」の部分に記載しない。

実際に要件定義書内の文章をテンプレートへと書き換える。交通費登録内容の照会という機能 [17] を用いて説明する。当該機能は、図 6 のような文章で記述されている。これをテンプレートに変換した場合、図 7 のようになる。上記の通り、「UI に」などのような FP 計測に関係ない部分を取り除かれ、計測に必要な要素のみが残ったことが分かる。また、基本的に「何を入力したか」「何

図 5. TF のテンプレート

を出力したか」「何を参照したか」のみをテンプレートに合わせて記せばよいため、FP 計測の知識がない者であっても変換は容易である。

### 3.3.2 DF のテンプレート記述

DF の計測に必要な要素は以下の通りである。

- 種別…ILF, EIF
- DET…アプリケーション上で参照される項目の数
- RET…データ群の属性内のサブグループの数

RET とは、1 つの DF の中に混在している異なる意味合いを持つデータのまとまりの数を指す。しかし、仕様から得られる要素を用いて対象の DF がどの DF のサブグループであるかを判別することは、仕様の中で直接的な言及が行われていない限り不可能である。そのため、本研究では RET を常に 1 として数えることとした。よってテンプレートに沿って書き換えられた文章のみで FP 計測を行うためには、その文章から「種別」「アプリケーション上で参照される項目の数」といった情報が得られなければならない。計測に必要な要素を含み、かつ判別の容易な文章を作るためのテンプレートが図 8 である。実際に図 8 のテンプレートを用いて変換を行うと図 9 上部にある文章は、図 9 下部のように変換される。

### 3.4. 自然言語解析による要素の抽出

先述したテンプレートに沿って書き換えた仕様から FP 計測に必要な要素を以降で述べる方法で抽出する。

#### 3.4.1 DF からの要素抽出

図 8 のテンプレートにおける (項目) は当該 DF が持つ全項目である。しかし DF の FP 計測に必要なものは

「所有する項目の数」ではなく「アプリケーション上で参照される項目の数」であるため、このテンプレートからでは DET が数えられない。また、種別についての言及もない。そこで、前述した TF の情報を用いて以下のように判別を行う。

- DET の判別

当該 DF のテンプレートにて示された項目のうち、TF の中で一度以上使用された項目のみを DET として数え上げる。

- ILF と EIF の判別

当該 DF が一度でも EI 内にて登録や更新が行われていた場合は ILF、そうでない場合は EIF として判別する。例として「A を更新する。」と記述されていれば A は ILF と判別される。

### 3.4.2 TF からの要素抽出

図5のテンプレートにおける(入力項目)、(出力項目)は DET の一部である入出力項目に相当し、(データ群)は処理に関わる DF を示している。また、「～というメッセージが表示されることがあります」という文によってメッセージの有無が判別できる。従って、このテンプレートで、「入出力項目」「メッセージの有無」「関わる DF の数」は文章から直接読み取れる。

一方で「種別」については明記が行われていない。そこで、このテンプレートから該当 TF の種別が EI, EQ, EO の3種の内どれに該当するのかを以下の2段階で判別する。

- EI と EQ, EO の判別方法

EI と EQ, EO の判別に使われている用語で行う。具体的には、「登録」、「更新」といった外部入力に関する用語であれば EI、「表示」、「出力」といった用語が用いられているなら EQ, EO とした判別を行う。判別のための用語は、辞書に別途登録しておく。

- EQ と EO の判別方法

EQ と EO の判別には、動詞の目的語となる項目を用いて判別を行う。具体的には、FP 計測規則 [12] に則って、DF に含まれていないデータ(項目)が出力されていれば、トランザクションにおいて何らかの処理を行っているかと判断して EO と判別し、

精算データの照会は、UIに社員番号と登録日付を入力することで、画面中央に利用日と目的と金額と交通機関とFromとToを表示する。エラーが起きた際にエラーメッセージが表示される。社員番号と登録日付を基に、交通費精算ファイルを参照することで利用日と目的と利用金額と交通機関とFromとToを得る。

図 6. 変換前の文章

精算データの照会処理は、社員番号と登録日付を入力することによって、利用日と目的と利用金額と交通機関とFromとToを表示します。交通費精算ファイルを参照し、利用日と目的と利用金額と交通機関とFromとToを得ます。また、入力データに誤りがあった際エラーというメッセージの表示を行います。

図 7. 変換後の文章

そのようなデータがない、つまり DF の項目をそのまま出力していれば EQ と判別する。

### 3.4.3 TF でない処理について

今回の計測上、仕様の書き換えの際に TF ではない処理を TF のテンプレートを使って変換してしまう場合があり得る。今回の手法では、EI, EQ, EO の判断のため辞書に登録している「更新」や「表示」「出力」等の用語が存在しない場合、EI, EQ, EO ではないとして TF としては扱われないため、計測には問題ない。

### 3.4.4 重複処理について

FP 計測において、処理内容が同一の処理は重複処理と呼ばれる。この重複処理は複数存在しても1つしか計測しないルールとなっている。そのため、図10のように同一の項目を持つ処理が複数存在した場合、それらを重複処理と判断し、1つしか数えないようにする。

## 3.5. FP の計測

ここまでのステップにて FP 計測に必要な要素が全て揃うため、2. 1. で記述したように要素から複雑度および FP を算出する。

(DF名)は、(項目)を持つデータ群です。

図 8. DF のテンプレート

データベース名	データ項目
図書情報	出版社, 題名, 著者名
利用者情報	氏名, 利用者番号

図書情報は、出版社と題名と著者名を持つデータ群です。  
利用者情報は、氏名と利用者番号を持つデータ群です。

図 9. DF のテンプレート記述への変換

### 3.6. 計測ツールの実装

これまでに述べた方針に基づく FP 計測ツールを Java 言語を用いて試作した。計測ツールの役割は「cabocha を用いた自然言語解析による計測に必要な要素の抽出」および「DF や TF の種類などの cabocha では抽出できない要素の判別」、そして「要素を用いた FP の算出」である。図 11 に本研究における FP 計測の流れと計測ツールの関係を示す。

計測の流れは以下の通りである。

1. 要件定義書のうち、機能に関わる部分の文章を意味内容が明確となるよう手動で変換する。
2. 書き換えた文章をテンプレートを用いて計測用の仕様を手動で変換する。
3. 計測用の仕様に cabocha による自然言語解析を行い、計測に必要な要素を抽出する。
4. 3. で抽出した要素を用いて、FP 計測を行い結果を出力する。

## 4. ケーススタディ

本章では、対象となる 4 つの仕様書に対して自動計測を行った結果及び考察をまとめる。

処理名	種別	入力項目	出力項目	参照した DF	メッセージの有無
図書検索A	EQ	図書番号	題名, 著者名	図書情報	有り
図書検索B	EQ	図書番号	題名, 著者名	図書情報	有り

処理名以外の全項目の内容が一致  
→「図書検索A」と「図書検索B」は重複処理と判断

図 10. 重複処理となる例

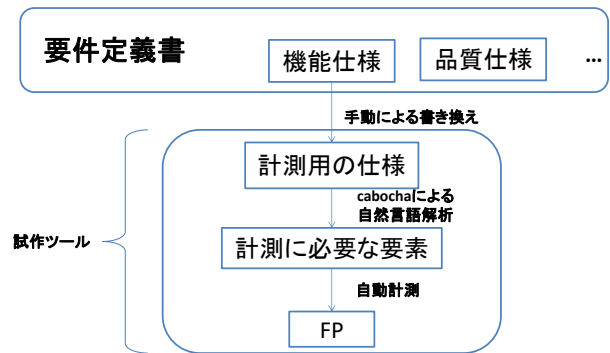


図 11. 計測フローと計測ツールの役割

### 4.1. 対象システムの概要

今回自動計測の対象とした 4 つのシステムの概要を記す。

#### 1. 旅費精算システム

交通費の管理を行うシステムである。社員が出張にて支払った交通費を登録しておくことで上司がそれを確認、承認できる。また、承認されたものについては銀行への支払い命令が行われる [17]。

#### 2. 図書システム

図書の管理を行うシステムである。図書の登録や増冊、検索や削除などが行える。ある企業の社内勉強用資料に記載されているものである。

#### 3. 酒屋問題

酒の在庫管理を行うシステムである。積荷として運ばれてきた酒の銘柄や本数の登録や、出庫指示などが行える [18]。

#### 4. 健康情報サイト

病院や健康情報について扱うサイトである。健康診断が行われている病院の検索や健診情報の登録が行える [19]。

表 4. 計測結果

		旅費精算				図書システム				酒屋問題				健康情報			
		手動		自動		手動		自動		手動		自動		手動		自動	
DF	ILF	17	7	17	7	14	14	14	14	14	14	14	14	21	21	21	21
	EIF		10		10		0		0		0		0		0		0
TF	EI		12		12		13		13		9		12		30		30
	EQ	52	20	52	20	40	7	40	7	24	3	27	3	75	41	75	41
	EO		20		20		10		10		12		12		4		4
FP		69		69		44		44		38		41		96		96	

#### 4.2. 提案手法の計測結果と手動計測結果の比較

先述した4つのシステムの要件定義書に対して、FP計測の熟練者（一部はFP計測における国際資格であるCFPS[20]の有資格者）による手動計測結果および計測ツールによる自動計測結果を表4に示す。表4より、手動で計測した結果と比べ、酒屋問題においてTFとFPが3ずつ多く計測されていることが分かる。

#### 4.3. 考察

##### 1. 誤差についての分析

旅費精算システム・図書システム・健康情報サイトの要件定義書に対しては、手計算で得られたFPと同様の値を得ることができたが、酒屋問題に関しては誤差が生じた。この原因について考察する。

酒屋問題における誤差の原因を調べたところ、本来無視すべき付随処理を計測していることが原因であった。付随処理とはTFとして扱われる処理において、本来の処理に付随して行われる処理の事を言う。酒屋問題においては酒の在庫情報を更新する「在庫情報更新処理」が相当し、この処理は本来出庫の指示を担う出庫指示票を出力する処理である「出庫指示票出力処理」に付随して行われる処理であったが、今回の提案手法での計測では別々の処理として計測された。FP計測においてこの付随処理は無視される、つまり計測されないものであるが、その処理が付随処理であるかどうかはFP計測の知識がある者でしか判断できず、またツール側から判断することもできない。そのため、FP計測の知識があまりない者がテンプレートへの変換を行った場合、この付随処理も含めて記述してしまう可能性が

高く、結果として今回のように付随処理の分多く計測される可能性も高い。この問題に関しては、自動計測後の結果をファイル出力し、利用者の手によって修正・変更を行う方法による解決が望ましい。

##### 2. DFが明記されていない場合の対応

今回はDFやTFについて記述が正確に行われているという条件でFP計測手順を提案したため、有資格者が計測した際の値と近いものを得ることができた。しかし、実際の要件定義書の中にはDFについて明示的に記載されていないケースが存在する。その場合、今回の研究にて提案した手法をそのまま適用できないため、一つのアイデアとして、TFで扱われているデータ項目の集合をDFの候補とするということを考えており、今後その詳細を検討する予定である。

#### 5. あとがき

本研究では、自然言語で書かれた仕様書からのFP自動計測の実現を試みた。その方法として、テンプレートを用いて仕様を書き換えた上で、自然言語解析を用いてFP計測に必要な要素を抽出しFP計測を行う手法を提案した。4つの仕様書を対象とし、提案手法を用いて自動計測を行った。その結果、3つの仕様書においては手計算で見積りが行った場合と同様のFPを算出することができた。残り1つの仕様書では手計算で得られたFPと異なる値が得られたことから、その原因に対する考察を行った。

本研究の今後の課題としては、下記のような事が考えられる。

##### 1. 適用事例の追加



今回取り扱った2つの要件定義書は最大でもFPが100程度のものであり、また、FPが小さいために要件定義書内で扱われる処理も種類が少ない。そのため、FPが大きく、また、多種多様な要件定義書への適用を行いそこで生じた問題点を解決することで、本研究の実用性を高める。

## 2. 手法の改善

テンプレートを用いた書き換えコストの調査やテンプレート記述以外のより計測に適した記述方法の提案、および穴埋め式記述を用いたツールの開発によるより記述量の少ない計測手法の提案などによって本研究における手法そのものの改善を行い、本研究およびツールの有益性を高める。

## 謝辞

本研究を進めるにあたって、貴重なご助言を頂きました株式会社NTT データ 鷗澤仁氏、一般財団法人経済調査会 大岩佐和子氏、押野智樹氏に深く感謝申し上げます。

本研究は一部、日本学術振興会科学研究費補助金基盤研究(S)(課題番号:25220003)の支援を受けている。

## 参考文献

- [1] B. Boehm, C. Abts, S. Chulani. Software development cost estimation approaches - A survey. *Annals of software engineering*, Vol. 10, No. 1-4, pp. 177-205, Springer, 2000.
- [2] R. L. Glass. *Facts and fallacies of software engineering*, Addison-Wesley Professional, 2002.
- [3] A. J. Albrecht. *Function point analysis*. *Encyclopedia of Software Engineering*, Vol. 1, pp. 518-524, Addison-Wesley Professional, 1994.
- [4] 総務省行政管理局. 政府情報システムの整備及び管理に関する標準ガイドライン実務手引書. 政府情報システムの整備及び管理に関する標準ガイドライン実務手引書, p. 122, 内閣官房情報通信技術(IT)総合戦略室, 2015.
- [5] G. C. Low, D. R. Jeffery. *Function points in the estimation and evaluation of the software process*. *IEEE Transactions on Software Engineering*, Vol. 16, No. 1, pp. 64-71, IEEE, 1990.
- [6] B. Kitchenham. The problem with function points. *IEEE software*, Vol. 14, No. 2, p. 29, IEEE Computer Society, 1997.
- [7] T. Uemura, S. Kusumoto, and K. Inoue. Function-point analysis using design specifications based on the united modelling language. *Journal of software maintenance and evolution: Research and practice*, Vol. 13, No. 4, pp. 223-243, IEEE Computer Society, Wiley Online Library, 2001.
- [8] G. Cantone, D. Pace, and G. Calavaro. Applying function point to united modeling language: Conversion model and pilot study. *Software Metrics*, 2004. *Proceedings. 10th International Symposium on*, pp. 280-291. IEEE, 2004.
- [9] T. Edagawa, T. Akaike, Y. Higo, S. Kusumoto, S. Hanabusa, and T. Shibamoto. Function point measurement from web application source code based on screen transitions and database accesses. *Journal of Systems and Software*, Vol. 84, No. 6, pp. 976-984, Elsevier, 2011.
- [10] P. Fraternali, M. Tisi, and A. Bongio. Automating function point analysis with model driven development. In *Proceedings of the 2006 conference of the Center for Advanced Studies on Collaborative research*, p. 18. IBM Corp. , 2006.
- [11] IEEE Computer Society. *Software Engineering Standards Committee, IEEE-SA Standards Board. IEEE recommended practice for software requirements specifications*, IEEE, 1998.
- [12] IFPUG: *Function Point Counting Practices Manual*, Release 4. 3. International Function Point Users Group, 2010.
- [13] C. R. Symons. *Software sizing and estimating: Mk II FPA (function point analysis)*, John Wiley & Sons, Inc. , 1991.
- [14] 山口正明(著) 調重俊(監修). *ファンクションポイント cosmic- ffp 法実践ガイド一組込み系・リアル*

タイム系に最適なソフトウェア規模・工数の見積り方法, 日科技連出版社, 2007.

- [15] 柏本隆志, 楠本真二, 井上克郎, 鈴木文音, 湯浦克彦, 津田道夫. イベントトレース図に基づく要求仕様書からのファンクションポイント計測手法. Vol. 41, No. 6, pp. 1895-1904, 2000.
- [16] Cabocha/南瓜 <https://taku910.github.io/cabocha/>.
- [17] 日本ファンクションポイントユーザ協会. ファンクションポイント計測コース演習編課題テキスト, 2014.
- [18] 山崎利治. 共通問題によるプログラム設計技法解説. 情報処理, Vol. 25, No. 9, p. 934, 1984.
- [19] 経済調査会調査研究部第二調査研究室. Web サイト制作費の見積りの現状と課題, 2015.
- [20] CFPS 概要. [http://www.jfpug.gr.jp/app-def/S-102/wp/?page\\_id=1374](http://www.jfpug.gr.jp/app-def/S-102/wp/?page_id=1374).