

ソフトウェア再利用に向けた共通ゴール判別手法の提案

中村 祐貴 本田 耕三 中川 博之 田原 康之 大須賀 昭彦

既存システムの再利用開発手法を用いたシステム開発プロセスの効率化に対する期待が高まっている。顧客の要求を正しく反映し再利用すべき機能を容易に選択できるよう、過去に複数の既存システムに共通して存在するゴールを集約して作成された結合ゴールモデルを用いる手法が提案されている。しかし大規模システムのゴールモデル中から結合ゴールモデルの核となる共通ゴールを正しく判別することは困難である。そこで本研究では、効率よく共通ゴールを判別するための手法を提案し、それをツールに実装した。その際手法の精度を上げるために、ゴール間の類似度だけではなくゴールモデルの構造に基づいた判別ルールを提案している。テレビ・SNS・就職支援システムのゴールモデルを用いた実験の結果から、提案ルールが有効な場合とそうでない場合があることがわかった。またカメラのゴールモデルを用いた実験結果から、提案手法を使うと効率よく共通ゴールを判別できることが確認された。

The approach using the joint goal model is proposed for software reuse. Common goals are important to build the joint goal model. However the related works do not mention techniques for identifying the common goals. In this paper, we propose a technique to identify common goals. They are using the similarity of goals and the three proposal rules. The experiment using the goal models of the domain of television and social network system and employment support system shows the accuracy of the proposal rules. The experiment using the goal models of the domain of camera shows the accuracy of the proposal technique.

1 はじめに

近年、システムへの要求が多様化しておりシステムのライフサイクルが短縮化してきている。たとえば同じようなシステムでも、要求は発注者によって異なり、また大規模あるいは複雑なシステムの場合、各発注者ごとの要求も多様となる。一方で開発規模は大型化しており、開発者は今まで以上に効率よくシステムを開発できるようになる必要がある。こういった背景に対応できるよう、より効率よくシステム開発を行うための方法論が求められている。

このような状況において、近年 Software Product

Line Engineering (SPLE) という方法論が注目されている [1]。米国カーネギーメロン大学ソフトウェア工学研究所の定義 [18] によると、「Software Product Line とは、(中略) 共通かつ管理された特性の集合を共有する、ソフトウェア中心のシステムの集合であり、共通のコア資産の集合から決められた方法で開発されるもの」である。さらに [18] では、SPLE の効果として、生産性、品質、費用、労力、出荷までの時間、および新市場への参入能力が向上し、資源不足による組織の問題を解決できるとしている。SPLE では可変性分析と呼ばれる既存システムの機能を共通機能と可変機能に分類するための手法が重要である。これにより、必ず再利用すべき必須機能と発注者の要求に応じて再利用するかどうかを選択できる任意機能を判別できるようになるからである。一般的に SPLE では、システムの機能をモデル上に表現したフィーチャモデルや既存システムのソースコードなどを用いて可変性分析を行う [2]。

しかし、発注者側の要求が曖昧な場合、その要求が

Common Goal Identification for Software Reuse.

Yuki Nakamura, Kozo Honda, Hiroyuki Nakagawa, Yasuyuki Tahara, Akihiko Ohsuga, 電気通信大学大学院情報システム学研究科, The Graduate School of Information Systems, The University of Electro-Communications.

コンピュータソフトウェア, Vol.31, No.2(2014), pp.67-83.

[研究論文] 2013 年 1 月 14 日受付.

抽象的となり、開発者がフィーチャモデル上から発注者の要求を満たすために再利用すべき任意機能を選択することができない [3]。階層化された要求モデルであるゴールモデルには発注者の抽象的な要求とシステムの具体的な機能が表現されているため、ゴールモデルを用いて可変性分析を行うと発注者は達成したい要求を選択することで再利用すべき機能を選択することができるようになる [4]。ゴールモデルを用いた SPLE の研究の中に、複数のゴールモデルに共通して存在する共通ゴールを起点に結合ゴールモデルを作成する手法を提案しているものがある [5]。この手法は、ゴールモデルからフィーチャモデルを導出する手法 [6] やゴールを選択することで対応するフィーチャを判別する手法 [3] などと組み合わせることで、発注者の要求を正確に反映したフィーチャモデルの作成が効率的に可能となるため、SPLE における製品開発の際のリードタイム短縮につながる。

しかし、それら一連の研究を実現するためには結合ゴールモデルの核となる共通ゴールを判別するための手法が必要である。大規模システムの膨大な数のゴールの中から人手で共通ゴールを正確に判別することは困難であるが、それらの研究ではそのための手法が自明でない。

そこで本研究では、一企業内で開発された各製品系列内のゴールを共通・可変に分類するというシナリオを想定し、効率よく共通ゴールを判別するための手法を提案するとともに、それを実装したサポートツールを作成する。本手法では 2 つのゴールモデル中からツールが自動で「自動判別共通ゴール」と「自動判別可変ゴール」を判別する。その後、共通か可変かの判断が難しいゴールを「共通ゴールランキング」として出力し、開発者はそれを用いて手作業で共通ゴールの判別を行う。開発者の労力を削減させるためには、できるだけ多くの共通ゴールと可変ゴールをツールが自動でミスなく判別できる必要がある。ツールはゴール間の類似度に基づいて判別を行うが、ゴールモデル中に「類似度の高い非共通ゴール」と「類似度の低い共通ゴール」が存在するとうまく判別を行うことができない。しかし、それらのゴールはゴールモデルの構造を用いることで見分けることが

可能である。そのため、「1 ゴール 1 共通」、「下位ゴール優先」、「子ゴール共通性の反映」というゴールモデルの構造に基づいた 3 つのルールを提案した。

提案ルール及びそれを含めた提案手法全体の有効性を評価するために、様々な既存システムを対象としたゴールモデルを用いて実験を行った。テレビ・SNS・就職支援システムのゴールモデルを用いた実験の結果から、提案ルールが有効な場合とそうでない場合があることがわかった。またカメラのゴールモデルを用いた実験結果から、提案手法を使うと全て人手で判別を行うよりも早く正確に共通ゴールを判別できることが確認された。

2 章で一般的な SPLE の手法について説明し、3 章で特にゴールモデルを用いた SPLE の手法とその課題について説明する。4 章で提案手法について説明し、5 章で手法に対する評価実験の結果及び考察を示す。最後に 6 章でまとめと今後の課題について説明する。

2 Software Product Line Engineering

第 1 章で述べたように、Software Product Line Engineering (SPLE) は、システム開発における各種の問題点を解決可能にする方法論である。

SPLE では、再利用し易い再利用資産を構築するために複数の類似既存システムを用いた可変性分析が重要となる。可変性分析とは、類似システムすべてに共通して存在する共通機能と各システム固有の可変機能を分類することである。これにより、共通機能をすべて再利用し、それに発注者の要求に応じた可変機能を選択して追加することで、派生システムを効率よく開発できるようになる。

一般的に、可変性分析ではフィーチャモデルを用いることが多い [2][10][11][12]。フィーチャモデルとはフィーチャと呼ばれるシステムの機能を木構造で表現したモデルである。図 1 はフィーチャモデルを用いた可変性分析の結果を再利用開発に利用している例である。

フィーチャモデル上には、必須フィーチャを表す mandatory 関係や、いずれかもしくは全部を選択できる or 関係、いずれか 1 つのみしか選択できない alternative 関係や、任意フィーチャを表す optional

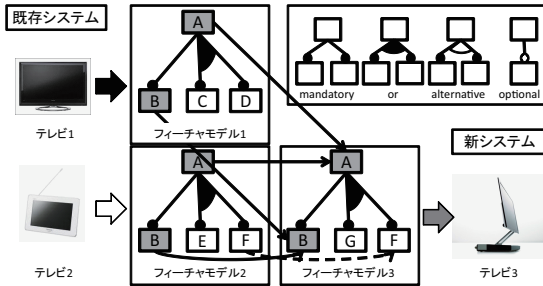


図1 フィーチャモデルを用いた SPLE の例

関係を表現できる [6][10][12]. 図の例では, A, B は mandatory ですべて再利用する必要がある, C, D, E, F は or でいずれかもしくは全部を選択して再利用できる. そのため, A と B は再利用し, C, D, E, Fの中から F を選択して再利用することでテレビ 3を開発している.

しかし, 発注者にはシステム開発の経験がなく, システムの機能に関する知識がない場合が多い. そのため, 発注者の要求が抽象的なものになってしまい, 開発者が発注者の要求を満たすために C, D, E, Fの中からどのフィーチャを再利用すればいいのかわからない可能性がある.

3 関連研究

ゴールモデルの特性を利用することで, 抽象的な要求から具体的な機能要求を導出することを可能とし, それにより再利用すべきフィーチャを特定可能にする手法が注目を集めている.

ゴールモデルとは, ゴール指向要求分析で用いられる階層化された要求モデルである [4][13]. ゴール指向要求分析では顧客がシステムに求める要求をゴールとして表現する. 抽象的なゴールをそれを達成するための具体的なゴールへと分解していくことでゴールモデルを作成する. このとき分解元のゴールを親ゴールと呼び, 分解先のゴールを子ゴールと呼ぶ.

ゴールの分解は AND 分解と OR 分解の 2 種類に分けられる. AND 分解はその子ゴールすべてが達成された場合にその親ゴールも達成される場合で, OR 分解はその子ゴールのいずれかが達成された場合にその親ゴールも達成される場合である.

ゴールモデルの上位ゴールにはビジネス的な「組織/人物の目指す理想的な状態」である顧客の抽象的な要求が表現され, 下位ゴールにはシステムの具体的な機能が表現される [6]. つまり, ゴールモデルの上位ゴールから下位ゴールに向かって達成したいゴールを選択していくことで, 最終的にシステムに必要な機能要求が判明する.

こういった理由から, ゴールモデルを用いて SPLEを実現しようとしている研究がある. それらの研究では, 複数のゴールモデルから結合ゴールモデルを作成 [5] し, そこからフィーチャモデルを導出 [6] している. さらに, 顧客が結合ゴールモデルの中から新システムを用いて達成したいゴールを選択していくことで, 再利用すべきフィーチャを特定 [3] できるようにしている.

Uno らは複数のゴールモデルを結合することで, 結合ゴールモデルを作成する手法を提案している [5]. これにより, 複数の類似既存システムに共通して存在する共通ゴールと各システム固有の可変ゴールを 1 つのモデル上に表現できる (図 2).

Yu らはゴールモデルからフィーチャモデルを導出する手法を提案している [6]. そこでは, ゴールモデルの下位ゴールからフィーチャを導出している. その後, ゴールモデルの AND, OR 関係とフィーチャモデルの mandatory, or, alternative, optional 関係を対応付けることで, 図 3 のようにフィーチャモデルを導出している.

結合ゴールモデルからフィーチャモデルを導出できるようになったため, Mohsen らは達成したいゴールを選択することで必要なフィーチャのみを選択するた

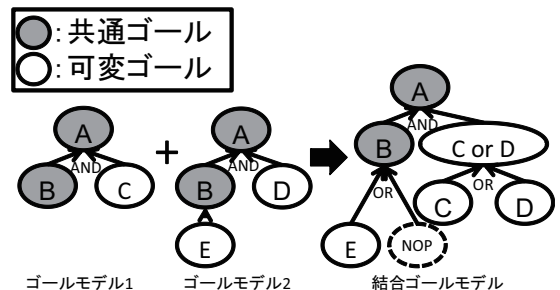


図2 結合ゴールモデル作成の例

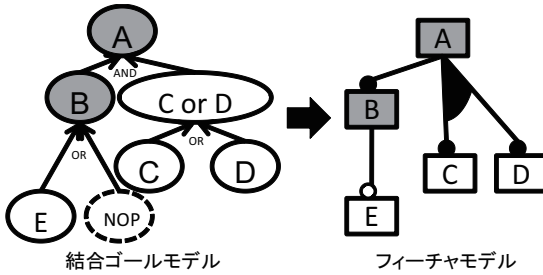


図3 ゴールモデルからのフィーチャモデル導出例

めの手法を提案している [3]。そこでは、ゴールに満足度を設定し、AND/OR の論理演算に従い自動で必要なフィーチャだけを残す。これにより、たとえば図4のように顧客がゴールに重みづけをすることで自動的に必要なフィーチャを選択できる。

説明してきた関連研究の中で、結合ゴールモデルを作成する際に重要となる共通ゴールの判別手法が自明でない。共通ゴールとは、複数のゴールモデル内に共通して存在するゴールである。2つのゴール同士が共通かどうかを判別するためには、ゴール名だけではなくそのゴールが意味していることや親子関係などを考慮しなければならない。また、現場で扱う大規模システムはゴール数が膨大な数となるため、その中から人手で共通ゴールを判別することは困難である。

4 提案手法

4.1 研究の目的

本研究では、開発者が効率良く共通ゴールを判別するための手法を提案する [15]。また、それを実装したサポートツールを作成することで開発者の労力を削減させる。

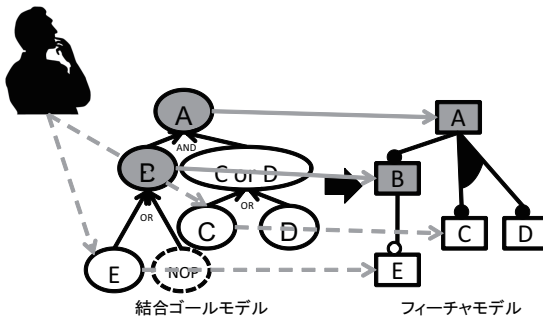


図4 再利用すべきフィーチャの特定例

研究の最終目標は、すべての共通ゴールを過不足なくツールに自動で判別させることである。その第一段階として本研究では、ツールが自動で判別できるゴールの範囲を広げることで開発者の労力削減を目指す。

4.2 研究の前提

本研究では「一企業内で開発された各製品系列内のゴール (〇〇シリーズの中に納まる製品だけ) を共通・可変に分類する」というシナリオを想定している。ここで、現場では企業内の仕様書の用語や記述方法などを可能な限り統一し誤解を招かないようにしている。また、不統一な用語を自動で見つけ出し統一するための技術 [16] や、統一された形式に自動で変換するための入力フォーマット [17] が研究・開発されている。こういった背景から、ゴールの用語・記述方法・粒度が既に統一されていることを前提として手法を提案する。

また、本手法で判別された共通ゴールを起点に Uno らの手法により結合ゴールモデルが作成され、Yu らの手法により下位ゴールからフィーチャが導出されるというシナリオを想定している。それに合わせるため、本研究では「組織/人物の目指す理想的な状態」である上位に位置するゴールではなく、システムごとの違いが顕著に現れフィーチャを導出する部分でもある具体的な下位に位置するゴールを用いて手法を提案する。

4.3 提案手法の概要

本手法ではツールが自動で「自動判別共通ゴール」と「自動判別可変ゴール」を判別し出力する。その後、残った共通可変の判別が難しいゴールを「共通ゴールランキング」として開発者に提示し、手作業で判別を行う。

図5のようにツールでは「提案ルール」の比較順序に従ってゴールを比較しながらゴールモデル間の「ゴール名の類似度」を計測し、閾値を基準に「自動判別共通・可変ゴール」を出力する。その後、ツールが自動で共通・可変の判別を行うのが困難なゴールを「共通ゴールランキング」として出力する。

本章では、まずキーワードとなる「自動判別共通・

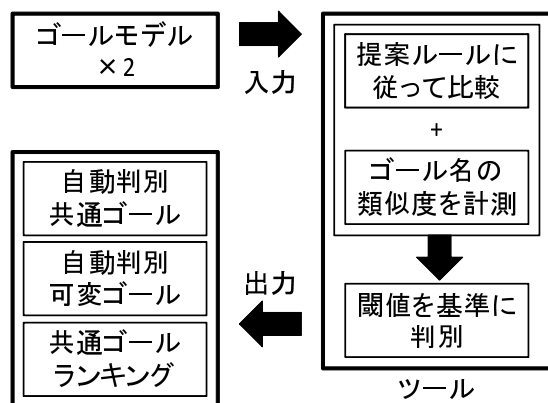


図 5 提案手法の概要

可変ゴール」,「共通ゴールランキング」について説明する。次に,提案手法の精度が下がる可能性のある場合について説明し,それらを解決するための「提案ルール」について説明する。その後,手法のベースとなる「ゴール名の類似度の算出方法」について説明する。最後に,「自動判別共通・可変ゴール」を判別する際の基準となる「閾値の設定方法」について説明する。

4.3.1 自動判別共通・可変ゴール

本手法では、共通ゴールの中でツールが自動で判別するゴールを「自動判別共通ゴール」と呼ぶ。「自動判別可変ゴール」も同様に定義される。これらのゴールは「提案ルール」の比較順序に従って「ゴール名の類似度」が算出され「閾値」を基準に判別・除外される。これらの多くをツールが自動で判別しゴールモデルから除外できると、共通ゴールを判別するために照合しなければならないゴール数が大幅に減少する。

4.3.2 共通ゴールランキング

ゴールモデルから「自動判別共通・可変ゴール」が除外されると、ゴールモデル内にはツールが自動で共通・可変の判別を行うのが困難なゴールのみが残る。これらのゴールは「ゴール名の類似度」に基づいてランク付けされ、「共通ゴールランキング」として開発者に提示される。ランキングの上位に出現するゴールほど、共通ゴールの可能性が高い。正解の共通ゴールが上位に出現すると、それ以外のゴールは見なくてもよくなるため照合しなくてもよいゴール数が減

少する.

4.3.3 ゴール名の類似度のみを用いて判別を行った場合の問題点

ゴール名の類似度のみを用いて判別を行うと、「類似度の高い非共通ゴール」を自動判別共通ゴールとして出力してしまう。同様に、「類似度の低い共通ゴール」を自動判別可変ゴールとして出力してしまう。これらのゴールの存在により、提案手法の精度が下がる。言い換えると、これらのゴールを見分けることができるようになれば提案手法の精度が上がることになる。ここでは「類似度の高い非共通ゴール」と「類似度の低い共通ゴール」について説明し、それらを解決するにあたっての方針について説明する。

類似度の高い非共通ゴールが出現する場合

類似度の高い非共通ゴールとは、ゴール名は似ているが意図している内容が異なるゴールのことである。

例えば、図 6 ではゴールモデル G1 内の (1-2) とゴールモデル G2 内の (2-1) と (2-2) の類似度が非常に高い。(1-2) と (2-2) が正しい共通ゴールのセットなので、(1-2) にとって (2-1) は類似度の高い非共通ゴールとなる。これをツールが自動で見分け、誤りである (1-2) と (2-1) ではなく、正解である (1-2) と (2-2) を共通ゴールのセットとして判別できる必要がある。

類似度の低い共通ゴールが出現する場合

類似度の低い共通ゴールとは，ゴール名は似ていないが意図している内容が同じゴールのことである。

例えば、図 7 では 2 つのゴールモデル間のゴールの類似度があり高くはない。そのため記述の違いにより、(1-1) と (2-1) は類似度の低い共通ゴールとなる。これをツールが自動で見分け、自動判別可変ゴールと判別しないようにする必要がある。

解決の方針

解決の方針として、2つの方法を検討した。

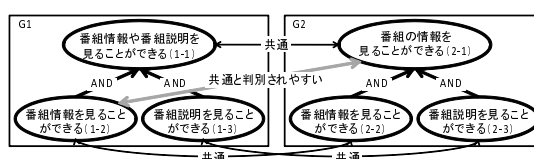


図 6 類似度の高い非共通ゴールの図

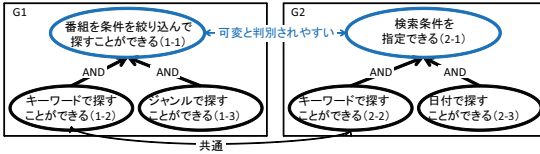


図7 類似度の低い共通ゴールの図

- (1) 自然言語処理 [7] を用いて類義語を定義する
- (2) ゴールモデルの構造 [15] を利用する

(1) は、ドメインに特化したオントロジーを作成する [8] などの方法により、単語に意味を与えることでゴール同士の類似度をより繊細に計測するという方法である。本論文でも、4.2 に挙げた本研究の前提において、[16] のような本カテゴリに属する手法を利用している。しかし一般にはドメインに特化したオントロジーを各ドメイン毎に開発する必要がある。また、「類似度の低い共通ゴール」を見分けることは可能だが、「類似度の高い非共通ゴール」を見分けることは困難である。

(2) はゴール単体だけで判別を行うのではなく、その親子ゴールの情報も活用するという方法である。この方法では、親子ゴールの情報から「類似度の低い共通ゴール」と「類似度の高い非共通ゴール」の両方を見分けることができる。また、ゴールモデルの構造を利用するためドメイン毎に特別な準備を行う必要がない。

以上により本論文では、(2) の方法を中心とする方がより汎用的に問題を解決できると考え、ゴールモデルの構造に基づいた方法を提案することにした。

4.4 ゴール名の類似度の算出方法

自動判別共通・可変ゴールの判別や、共通ゴールランキングのランク付けは「ゴール名の類似度」に基づいて行われる。類似度はジャカード係数という既存の手法 [9] を用いて算出される。ジャカード係数とは、集合 X と Y の共通要素数を少なくとも一方にある要素の総数で割ったものである。

$$sim = \frac{|X \cap Y|}{|X \cup Y|}$$

このとき比較を行う単語は名詞と動詞のみで、動詞は基本形に変換してから比較する。例えば、「g1:見

たい番組を探すことができる」「g2:見たい番組を番組表で探すことができる」という2つのゴール間の類似度を算出する際には、まず $g1$ を「見る」、「番組」、「探す」、 $g2$ を「見る」、「番組」、「番組表」、「探す」のようにゴールの記述を単語に分割する。その後、 $g1$ 、 $g2$ 間のジャカード係数を算出し類似度とする。

例の場合、 $|X \cap Y|$ は $g1$ 、 $g2$ に共通して存在する「見る」、「番組」、「探す」の3単語で、 $|X \cup Y|$ は $g1$ 、 $g2$ の少なくとも一方に存在する「見る」、「番組」、「番組表」、「探す」の4単語である。つまり、 $g1$ 、 $g2$ 間の類似度は $sim = 3 / 4 = 0.75$ となる。

4.5 提案ルール

ゴールモデルの構造を利用した、「1 ゴール 1 共通」、「下位ゴール優先」、「子ゴール共通性の反映」という3つのルールについて説明する。

4.5.1 1 ゴール 1 共通・下位ゴール優先

「1 ゴール 1 共通」とは、1つのゴールに対して複数のゴールを共通ゴールにしないというルールである。このルールにより共通と判別されたゴールのセットはその他のゴールと共通ではないとして比較対象から除外することができる。1 ゴール 1 共通のルールを適用するためには、ゴールモデル内に同一ゴールが存在してはいけないため、本手法では1つのゴールモデル内に同一のゴールが複数存在しないことを前提とする。

「下位ゴール優先」とは、最下位ゴール(葉ゴール)からゴールを比較していき、自動判別共通だと判別されたゴールが複数ありそれらの類似度が等しいときは、下位ゴールを優先して判別するというルールである。

これは、親ゴールをより具体的な子ゴールに分解するというゴールモデルの特性を利用している。類似度が等しい場合には、より具体的なゴールから共通ゴールのセットを作ること、正しい共通ゴールのセットを作ることが可能となる。

例えば図6の場合、下位ゴール優先を適用し、まず(1-2)、(1-3)と(2-2)、(2-3)だけで比較を行う。これにより、(1-2)と(2-2)、(1-3)と(2-3)という2つの正解の共通ゴールのセットを判別できる。1 ゴール 1

共通のルールからこれらのゴールを比較対象から除外することで、(2-1)の共通ゴール候補に(1-2)を出現させないようにすることができる。

4.5.2 子ゴール共通性の反映

「子ゴール共通性の反映」とは、子ゴールの中に自動判別共通ゴールがあればその親ゴールは自動判別可変にはしないというルールである。

これも同様に親ゴールをより具体的な子ゴールに分解するというゴールモデルの特性を利用している。親ゴールが抽象的な表現をしていて、それにより共通かどうかの判断ができない場合でも、より具体的な子ゴールの共通性を反映することで共通かどうかを判断することができる。

ただし子ゴール同士が共通であっても親ゴール同士が必ずしも共通とは限らないため、自動判別共通にはせず後述する共通ゴールランキングで開発者に提示する。

例えば図7の場合、(1-1)の子ゴールである(1-2)と(2-1)の子ゴールである(2-2)は自動判別共通ゴールとなる。この場合、子ゴール共通性の反映を適用するとどんなに類似度が低くても、(1-1)と(2-1)は自動判別可変とはしない。

4.6 閾値の設定方法

提案ルールに従ってゴール名の類似度を算出した後、閾値を設定することでツールは自動判別共通・可変ゴールを判別する。閾値の設定に失敗すると、それが手戻りを発生させる原因となる。それを防ぐために、最適な閾値の求め方について説明する。

図8を用いて説明すると、類似度が閾値Aを超えた場合、それらのゴールのセットは自動判別共通ゴールと判別される。また、他方のゴールモデルのすべてのゴールとの類似度が閾値Bを超えない場合、そのゴールは自動判別可変ゴールと判別される。

類似度が閾値Aと閾値Bの間にある共通・可変の判断が難しいゴールは、共通ゴールランキングとして開発者に提示される。

このとき、自動判別共通ゴールの中に可変ゴールが含まれていたり、自動判別可変ゴールの中に共通ゴールが含まれることが極力起こらなくなるような閾値

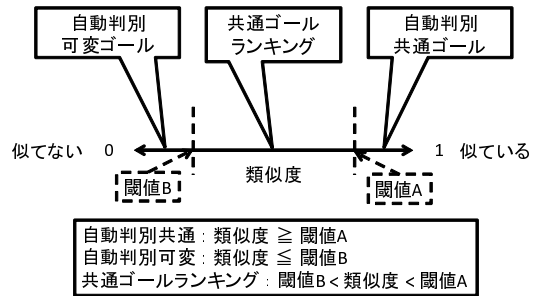


図8 類似度と閾値の図

設定が必要である。

非共通なゴールを自動判別共通ゴールとして判別してしまうと、発注者の要求に応じて選択できる機能要求を必ず選択しなければならないなくなってしまい、再利用しなくてもいい機能を開発してしまうという事態が発生する。

また、共通なゴールを自動判別可変ゴールとして判別してしまうと、発注者の要求次第では必要な機能要求を選択しそこなうことになってしまい、再利用しなければならない機能を開発しないという事態が発生する。

つまり本手法の(1)のプロセスでは「自動判別共通ゴール」のみを、(2)のプロセスでは「自動判別可変ゴール」のみを判別できるようにすることを目標として最適な閾値を見つけ出す必要がある。

最適な閾値を見つけ出すための予備実験について説明する。図9に予備実験のイメージを示す。予備実験ではまず、ゴールモデルの一部のゴールを用いて専門家が手作業で共通ゴールと可変ゴールの分類を行う。その後、5章で後述するツールを用いて、閾値を変動させながらそれらの共通ゴールと可変ゴールを適合率100%で判別できる閾値をそれぞれ見つける。

このとき、予備実験に用いるゴールはゴールモデルの中からランダムに選ばれたものである。まずツールにゴールモデルを入力し、閾値を変動させながらツールが出力するゴールのセットをランダムに選択する。これにより、閾値が高いときには共通ゴールのセットが多く選択され、閾値が低いときには可変ゴールのセットが多く選択される。すべての閾値で行うことにより、共通ゴールと可変ゴールの両方が含まれ

表 1 実験に用いたゴールモデルの特徴

ドメイン	テレビ		SNS		就職支援システム		カメラ	
ゴールモデル名	製品A	製品B	サービスC	サービスD	E社システム	F社システム	製品G	製品H
ゴール数	51	70	31	54	33	84	57	50
正解共通ゴール数	20		9		13		23	
企業	同企業		別企業		別企業		同企業	
作成者	同作者		同作者		別作者		同作者	
言語	日本語		英語		日本語		英語	
ゴールモデル間のゴールの類似度	高		低		低		高	
ゴールモデル内のゴールの類似度	高		低		高		低	

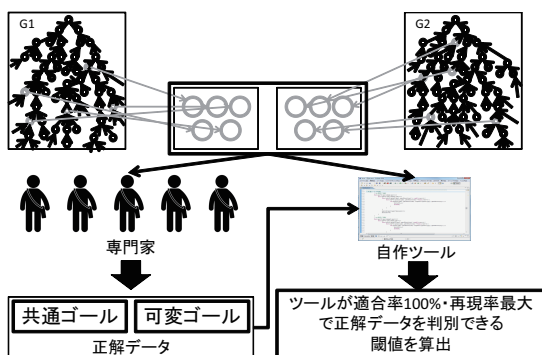


図 9 予備実験の図

たゴールをランダムに選択することができる。

5 実験と考察

提案ルール (1 ゴール 1 共通, 下位ゴール優先, 子ゴール共通性の反映) を用いた際の精度の変化を評価するために, テレビ・Social Network System(SNS)・就職支援システムに対するゴールモデルを用いて実験を行った。

さらに提案ルールが有効な場合に, 提案手法が効率化に与える影響を評価するためにカメラに対するゴールモデルを用いて実験を行った。

各ゴールモデルのゴール数や特徴, 正解データ (正解共通ゴール数) は表 1 の通りである。また, データの詳細は WEB ページ < http://www.ohsuga.is.uec.ac.jp/member/n_yuki_goals.zip > からダウンロードできる。

5.1 提案手法の実装

Eclipse3.2 上で JAVA 言語を用いて, 提案手法を実装したツールを作成した。ツールにテキスト形式でゴールモデルを入力すると「自動判別共通ゴール」, 「自動判別可変ゴール」, 「共通ゴールランキング」が

出力される。

その際に出力結果を比較できるように, 提案ルールである「1 ゴール 1 共通」, 「下位ゴール優先」, 「子ゴール共通性の反映」を用いるかどうかを選択できるようになっている。また, 出力される「自動判別共通ゴール」, 「自動判別可変ゴール」はツールの閾値を変動させることで変化する。閾値は 0.0 から 1.0 の範囲で変動させる。

「自動判別共通ゴール」では, 閾値 0.0 は両ゴールを構成する単語がまったく一致してなくても自動判別共通とする状態で, 1.0 は両ゴールを構成する単語が完全に一致している状態でなければ自動判別共通とはしない状態である。

「自動判別可変ゴール」では, 閾値 0.0 は両ゴールを構成する単語が完全に不一致である状態のみ明らかに可変だと判別する状態で, 1.0 は両ゴールを構成する単語がどれだけ一致していようが明らかに可変だと判別する状態である。

5.2 実験の目的

本実験では 4 章で提案した手法の有効性, 妥当性, 適用可能性等を検証するため, いくつかの事例を取り上げて適用評価する。

そのために, テレビやカメラのような「一企業内で開発された各製品系列内のゴールを共通・可変に分類する」という本研究のシナリオに合致している場合だけでなく, SNS や就職支援システムのような「同ドメイン別製品系列のゴールモデル同士」の場合でも共通ゴールを判別しそれらの実験結果を比較する。

また本実験で組み込み系と WEB 系のドメインを選択したのは, 身近で実験の際に被験者が直接操作できるシステムを用いる必要があったためである。

5.3 テレビ・SNS・就職支援システム

5.3.1 実験の方法

テレビと SNS ではまず開発者という位置付けで著者の一人が手作業でゴールモデルを 2 つずつ作成した。また、就職支援システムでは情報工学を専攻している修士課程の学生 2 人にそれぞれ 1 つずつゴールモデルを作成させ、さらに修士課程の学生 2 人を追加しゴールモデルをレビューさせた。

作成の際には、それらの取扱説明書を参考にするとともに実際にシステムを操作した。また、取扱説明書から要求モデルを作成している Berry らの手法 [14] を参考にしている。

本来、手法が現場で効果を発揮できることを証明するためには、そのシステムの開発に携わった現場の開発者にゴールモデルを作成してもらう必要がある。しかし本環境下では困難であった。

一方被験者である学生たちは、情報工学の修士学生であり授業の演習などを通じて上流から下流まで一連のシステム開発の流れを経験している人が多く、また扱ったシステムも身近で操作に馴染みがある。さらに扱ったシステムはいずれもある程度の取扱説明書が用意されており、上述のように Berry らの手法を参考にしてゴールモデルを作成できた。したがって本実験では、経験の浅い開発者が、体系的な手法も参考にして、PBL (Project-Baese Learning, 実際の開発プロジェクトを通じて開発手法を修得する学習法) を通じて、開発者 (より正確には要求分析者) の役割でゴールモデルを作成した、といったような状況を想定している。このため、実験の被験者に関する妥当性は、学生の経験レベル、および手法の妥当性に依存するが、いずれにしても十分であるとはいえない。そこで実際の開発者による評価は今後の課題としたい。

ゴールモデルの作成時に重要となる最下位ゴールの分解粒度を本実験では「ユースケースレベルで表現できる所まで」とした。これにより、ゴールはシステム内の処理を表すレベルにまで分解されていない。

その後、修士の学生 5~9 人にレビューアとして手作業で共通ゴールを判別してもらい、正解データを作成した。

本実験では、提案手法を実装したツールの出力と正

解データを比較することで評価を行う。評価項目は以下の 3 つである。

(1) 「1 ゴール 1 共通」, 「下位ゴール優先」を用いる前後の「自動判別共通ゴール」の適合率 100 % 時の再現率

(2) 「子ゴール共通性の反映」を用いる前後の「自動判別可変ゴール」の適合率 100 % 時の再現率

(3) 「自動判別共通・可変ゴール」を除外する前後の共通ゴールランキングで正解共通ゴールが出現する順位

(1), (2) ではツールの閾値を変動させながら、正解データを適合率 100 % で判別できたときの閾値を見つけ、そのときの再現率を評価するとともに、提案ルール適用前後で結果を比較する。適合率、再現率は以下の式で計算される。

$$(\text{適合率}) = \frac{(\text{判別した正解ゴール数})}{(\text{判別したゴール数})}$$

$$(\text{再現率}) = \frac{(\text{判別した正解ゴール数})}{(\text{全正解ゴール数})}$$

このとき適合率 100 % 時の再現率を評価するのは、共通ゴールのみもしくは可変ゴールのみをどれだけ判別できるのかを評価したいからである。ツールが自動で判別した共通ゴールの中に可変ゴールが含まれていたり、可変ゴールの中に共通ゴールが含まれていると、それが手戻りの原因になるため、適合率は極力 100 % に近づけなければならない。そこで、少なくとも実験データに対しては適合率が 100 % となる閾値を評価対象とした。

(3) では共通ゴールランキングに正解共通ゴールが出現する順位の平均を評価するとともに、「自動判別共通・可変ゴール」を除外する前後で結果を比較する。

5.3.2 実験の結果

実験結果を図 10~21 に示す。各図は、閾値を 0 から 1 まで変化させた場合の、各提案ルール適用前後のそれぞれに対する、適合率と再現率の変化を示している。各図には、対象システム (テレビ, SNS, および就職支援システム)、ゴールの種類 (自動判別共通, および可変)、ならびに縦軸の数値 (適合率と再現率) を示す表題を付記している。また灰色の線は提案ルール適用前、および黒色の線は提案ルール適用

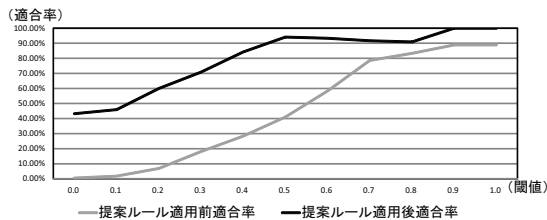


図 10 テレビ・自動判別共通・適合率

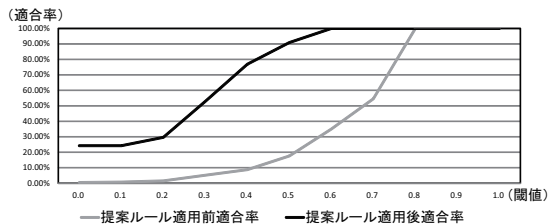


図 14 就職支援・自動判別共通・適合率

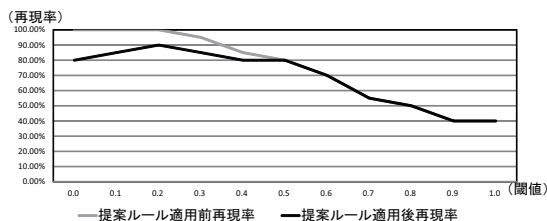


図 11 テレビ・自動判別共通・再現率

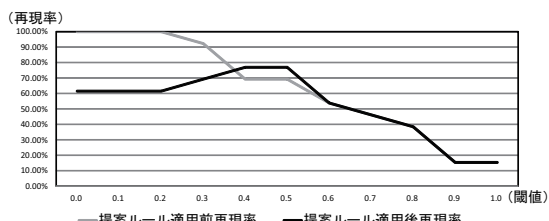


図 15 就職支援・自動判別共通・再現率

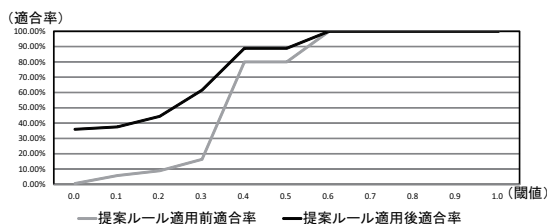


図 12 SNS・自動判別共通・適合率

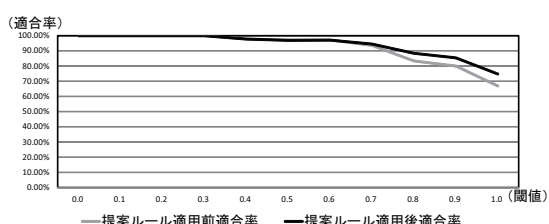


図 16 テレビ・自動判別可変・適合率

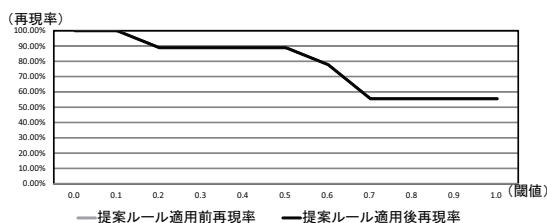


図 13 SNS・自動判別共通・再現率

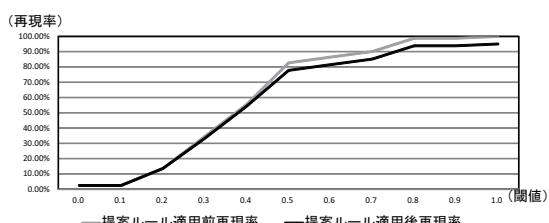


図 17 テレビ・自動判別可変・再現率

後を示す。自動判別共通ゴールについては「1 ゴール 1 共通」および「下位ゴール優先」の適用前後を、また自動判別可変ゴールについては「子ゴール共通性の反映」の適用前後を示している。

全ての場合において、提案ルール適用後には、適合率は上がることが多く、再現率は下がるが多い。そしてその逆になることはほとんどない(図 15 の「就職支援・自動判別共通」の再現率の一部のみ)。

次に各システムに対し、5.3.1 で述べた (1), (2),

および (3) の評価項目の値を表 2 に示す。評価項目の値とは、(1) と (2) では適合率が 100 % になる最大、あるいは最小の閾値における再現率のパーセント値を、(3) では共通ゴールランキングで正解共通ゴールが出現する順位の平均値を指す。表 2 において「前」「後」「差」は、(1) と (2) についてはそれぞれ提案ルールの適用前、適用後、および (適用後 - 適用前) の値を、また (3) についてはそれぞれ自動判別共通・可変ゴールの除外前、除外後、および (除外前 - 除外後)

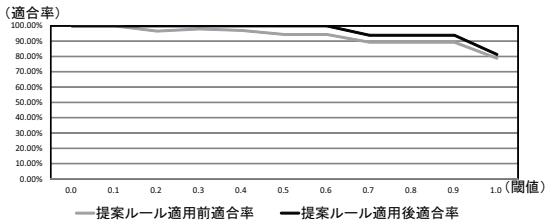


図 18 SNS・自動判別可変・適合率

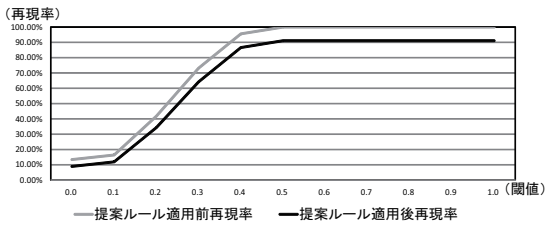


図 19 SNS・自動判別可変・再現率

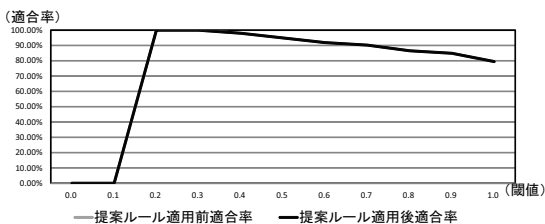


図 20 就職支援・自動判別可変・適合率

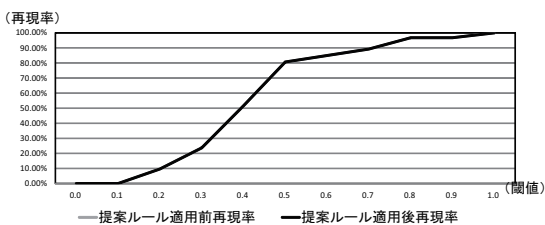


図 21 就職支援・自動判別可変・再現率

の値を、それぞれ表している。したがって、各「差」の値は、(1)と(2)ではルール適用前後での再現率の上昇ポイントを、(3)では自動判別共通・可変ゴール除外前後での正解共通ゴール出現順位の下降値を表すので、前者は正の方向に、後者は負の方向に大きい方が、提案手法の有効性を示すことになる。また(1)と(2)における「N/A」は、適合率が100%になる閾値が存在しないことを示す。

表 2 各システムに対する各評価項目の値

システム		テレビ	SNS	就職支援
(1)	前	N/A	77.78	38.46
	後	40.0	77.78	53.85
	差	+40.0	0	+15.39
(2)	前	34.57	16.42	23.66
	後	33.33	91.04	23.66
	差	-1.24	+74.62	0
(3)	前	1.60	1.00	2.31
	後	1.17	1.00	2.17
	差	-0.43	0	-0.14

5.4 カメラ

テレビ・SNS・就職支援システムに対するゴールモデルを用いた実験により、ゴールモデルの特徴が提案手法の精度に大きな影響を与えることがわかった。

そこで提案手法の精度が高い場合に、どれだけ効率よく共通ゴールを判別できるようになるのかを評価するためにカメラに対するゴールモデルを用いて実験を行った。

5.4.1 実験の方法

テレビ、SNS 同様に著者の1人が2つのカメラのゴールモデルを作成した。その際、提案手法の精度を高めるために最下位ゴールの分解粒度が粗くならないように注意した。

その後修士の学生8人をチームAとチームBの2チームに分け、全員にカメラ1とカメラ2のゴールモデルを配布した。

チームAの4人には、手作業で2つのゴールモデルから共通ゴールを判別させた。その際、作業時間を記録させた。

また、チームBの4人には提案手法を用いて共通ゴールを判別させた。まず、ゴールモデルの一部を用いて共通ゴールを判別させることで予備実験を行い、結果を提出させた。

閾値を変動させながらツールの出力と予備実験の結果を比較することで、予備実験のデータに対して適合率が100%になる閾値がわかる。この閾値を用いてツールにゴールモデルのすべてのゴールを入力することで、自動判別共通・可変ゴールが出力される。さ

表 3 カメラ実験の作業時間

	被験者A1	被験者A2	被験者A3	被験者A4	被験者B1	被験者B2	被験者B3	被験者B4
作業時間(分)	20	92	67	35	42(12+30)	43(16+27)	46(30+15)	26(14+12)

チームA作業時間平均(分)	53.5
チームB作業時間平均(分)	39.25

表 4 カメラ実験の実験結果

	被験者A1	被験者A2	被験者A3	被験者A4	被験者B1	被験者B2	被験者B3	被験者B4
正解共通ゴール数	23	23	23	23	23	23	23	23
判別した共通ゴール数	15	19	28	18	24	24	24	19
判別した正解共通ゴール数	14	16	20	14	22	22	22	18
適合率	93.33%	84.21%	71.43%	77.78%	91.67%	91.67%	91.67%	94.74%
再現率	60.87%	69.57%	86.96%	60.87%	95.65%	95.65%	95.65%	78.26%

チームA適合率平均	81.69%
チームB適合率平均	92.43%
チームA再現率平均	69.57%
チームB再現率平均	91.30%

らに、それらを除外した共通ゴールランキングが出力される。この除外ありの共通ゴールランキングをチーム B の 4 人に配布した。チーム A の 4 人にはそれを用いて共通ゴールを判別させ、結果を提出させた。

つまり、チーム B ではゴールモデルの一部を用いて予備実験を行うステップ 1 と共通ゴールランキングを用いて共通ゴールを判別するステップ 2 の 2 つのステップを行なった。時間を計測する際は、各ステップごとに計測し提出させた。

評価項目は、

- (1) 作業時間の平均 (チーム A, B 両方)
- (2) 判別した共通ゴールの適合率・再現率 (チーム A, B 両方)
- (3) ツールが判別した自動判別共通・可変ゴールの適合率・再現率 (チーム B のみ)
- (4) 共通ゴールランキングから判別した共通ゴールの適合率・再現率 (チーム B のみ)

である。

(1) と (2) の結果をチーム A と B で比較することにより、提案手法を用いることでどれだけ効率よく共通ゴールを判別できるようになるかを評価する。

また (3) の結果から、予備実験により最適な閾値を見つけることができるかを評価する。

さらに、(4) の結果とチーム A の (2) の結果を比較することで、共通ゴールランキングが効率化に与える影響を評価する。

加えて、(4) の結果とチーム B の (2) の結果と (3) の結果を比較することで、ツールの精度が効率化に与える影響を評価する。

5.4.2 実験結果

(1) 作業時間の平均

表 3 において、チーム B の括弧中の 2 つの数値は、左側が予備実験にかかった時間で、右側が共通ゴールランキングから共通ゴールを判別するためににかかった時間である。両チームの作業時間平均を見ると、結果として提案手法を用いた方が約 20 % の作業時間を削減できたことが分かる。

(2) 判別した共通ゴールの適合率・再現率

表 4 より、提案手法を用いた方が適合率が平均して約 10 % 上昇しており、また再現率が平均して約 20 % 上昇していることが分かる。

(3) ツールが判別した自動判別共通・可変ゴールの適合率・再現率

予備実験を行ったところ、4 つの共通ゴールのセットが判別された。これらのゴールを自動判別共通ゴールとしてツールが適合率 100 % で判別できたのは、閾値が 0.7 のときだった。また、これらのゴールを自動判別可変ゴールとして一切含めずにツールが判別できたのは、閾値が 0.3 のときだった。

これらの閾値でゴールモデル全体から自動判別共通・可変ゴールを判別したところ表 5 の結果になった。この表を見ると、自動判別共通・可変ゴールとも

表 5 カメラ実験でツールが判別したゴールの結果

カメラ・明らかに共通		カメラ・明らかに可変	
正解共通ゴール数	23	正解共通ゴール数	61
判別した共通ゴール数	14	判別した可変ゴール数	40
判別した正解共通ゴール数	14	判別した正解可変ゴール数	40
適合率	100.00%	適合率	100.00%
再現率	60.87%	再現率	65.57%

表 6 カメラ実験で共通ゴールランキングを用いて判別したゴールの結果

カメラ・共通ゴールランキングのみ	被験者B1	被験者B2	被験者B3	被験者B4
正解共通ゴール数	9	9	9	9
判別した共通ゴール数	10	10	10	5
判別した正解共通ゴール数	8	8	8	4
適合率	80.00%	80.00%	80.00%	80.00%
再現率	88.89%	88.89%	88.89%	44.44%

適合率平均	80.00%
再現率平均	77.78%

に適合率 100 %でゴールを判別できていることが分かる。このときの自動判別共通・可変ゴール療法の再現率を見ると、ゴールモデル全体から 60 %以上のゴールを自動で除外できたことが分かる。

(4) 共通ゴールランキングから判別した共通ゴールの適合率・再現率

表 6 における、共通ゴールランキングから共通ゴールを判別した際の適合率と再現率の平均と、表 4 のチーム A の結果と比較すると、同じ手作業でも適合率の平均が約 2 %下がっていて、再現率の平均が約 8 %上昇している。

5.5 考察

5.5.1 提案ルールの有効性の評価

まず「1 ゴール 1 共通」,「下位ゴール優先」を用いる前後の「自動判別共通ゴール」の適合率 100 %時の再現率より,「1 ゴール 1 共通」,「下位ゴール優先」はテレビと就職支援システムでは有効で, SNS では有効ではないことが分かる。これは、テレビと就職支援システムのゴールモデル内のゴールの類似度が高かったことにより「類似度の高い非共通ゴール」が出現したためだと考える。テレビでは図 9 の「番組情報を見ることができる」と「番組の情報を見ることができる」のような「類似度の高い非共通ゴール」のセットが出現していた。また、就職支援システムでは「企業を職種から検索できる」と「企業を職種で検索でき

る」のような「類似度の高い非共通ゴール」のセットが出現していた。

それに比べ SNS のゴールモデルでは、ゴールモデル内のゴールの類似度が低かったため,「類似度の高い非共通ゴール」が出現することはなく提案ルールを用いる必要がなかった。

次に「子ゴール共通性の反映」を用いる前後の「自動判別可変ゴール」の適合率 100 %時の再現率より,「子ゴール共通性の反映」は SNS では有効で, テレビと就職支援システムでは有効ではないことが分かる。これは、サービス C とサービス D のゴールモデル間のゴールの類似度が低かったことにより「類似度の低い共通ゴール」が出現したためだと考える。図 1 の「searching friends」と「searching in the search field」のような「類似度の低い共通ゴール」のセットが出現していた。

それに比べテレビのゴールモデルでは、2 つのテレビのゴールモデル間のゴールの類似度が高かったため,「類似度の低い共通ゴール」が出現することはなく提案ルールを用いる必要がなかった。

就職支援システムの E 社システムと F 社システムのゴールモデルはそれぞれ別の作者によって作成されたため、2 つのゴールモデル間のゴールの類似度が低かった。また、E 社システムのゴールモデル内に,「企業を具体的日時から説明会をする検索できる」や「企業を詳細な業種を複数選択検索できる」という目

的語が複数あるゴールが存在していた。さらに、E 社システムの「企業を先輩から検索できる」と F 社システムの「企業を先輩に聞く仕事の魅力で複合検索できる」は共通ゴールなのだが、作成者が違うことによりゴール名の長さが異なっていた。これらが原因で、ジャカード係数の信頼性が下がり、「類似度の低い共通ゴール」が多く存在していた。

しかし、「子ゴール共通性の反映」を用いてもそれらを見分けることができなかった。その原因として以下の 2 つがあげられる。

- ・最下位ゴールの分解粒度が粗い
- ・最下位ゴールの分解粒度が統一されていない

実際に、E 社システムの「企業を詳細な業種を複数選択検索できる」と F 社システムの「業種を小分類で検索できる」のようにさらに「業種」で分解できるようなゴールを両ゴールモデルでは分解していなかった。

「類似度の低い共通ゴール」が多く存在していたとしてもそれらをより具体的なゴールへと分解できれば、「子ゴール共通性の反映」によりそれらを見分けることが可能であった。

この結果から、「子ゴール共通性の反映」は最下位ゴールの分解粒度が細かくかつ統一されている場合に有効であるということがわかった。

5.5.2 カメラに対するゴールモデルを用いた 提案手法の有効性の評価

最後に 5.4 の評価項目 (1)～(4) について考察する。

(1) と (2) の結果をチーム A と B で比較することにより、提案手法を用いることでより短時間で多くの共通ゴールを正しく判別できていることがわかる。しかし、手法を用いても適合率が 100 % になっていないため、手戻りが発生する可能性がある。

(3) の結果では、予備実験で最適な閾値を見つけることができたため、ツールは適合率 100 % でゴールを判別することができていた。

(2), (4) の結果から、手作業で判別を行うと離れたゴールを見逃したり、共通ゴールランキングの上位のゴールを誤って共通だと判別してしまうことがあり、それにより適合率が下がっていたと考えられる。

これらの結果から実用化のためには、ツールの精度

を上げ手作業でゴールを判別する部分を減らすことにより適合率を 100 % に近づけることが重要だと考える。

6 手法の限界と課題

6.1 ゴールモデルおよびゴール名の記述の前提

実験の結果から、「類似度の高い非共通ゴール」や「類似度の低い共通ゴール」が出現しても、提案ルールを用いることで手法の精度の低下をある程度防止できることが確認できた。しかし、提案ルールを用いたとしても手法の精度が下がる場合がある。以下に具体例を示す。

(1) 「1 ゴール 1 共通」・「下位ゴール優先」が効果を発揮できない場合

a. ゴールの分解粒度が統一されていない場合

ゴールの分解粒度が階層ごとに統一されていない場合、下位ゴールから優先して共通ゴールのセットを判別する際に、階層がずれ「共通ゴールの親子ゴール」を共通ゴールと判別してしまう可能性がある。提案ルールの効果を発揮させるためには、階層ごとに粒度を定め、それに従ってゴールモデルを作成する必要がある。

b. ゴール名に否定が使われている場合

両方のゴールモデルに否定が使われたゴールと否定が使われていないゴールが存在しそれらの類似度が高い場合、それらは「類似度の高い非共通ゴール」となる。例えば、「A ボタンを押すことができる」と「A ボタンを押さないようにすることができる」は類似度の高い非共通ゴールである。それらのゴールは提案ルールを用いても判別できない。提案ルールの効果を発揮させるためには、ゴール名に否定を用いない必要がある。否定を用いる必要がある場合は、そのゴールを OR 結合にしフィーチャモデル作成時に alternative 関係に変換されるようソフトゴールに記載する。

c. 子ゴールが親ゴールの動詞や目的語を踏まえて記述されている場合

子ゴールが親ゴールの動詞や目的語を踏まえて記述されているとそれらは「類似度の高い非共通ゴール」となる。例えば、「予約をすることができる」の子ゴー

ルに「ケータイから指示できる」が存在し、「予約を解除することができる」の子ゴールにも「ケータイから指示できる」が存在する場合は「ケータイから指示できる」同士が類似度の高い非共通ゴールとなる。

(2)「子ゴール共通性の反映」が効果を発揮できない場合

a. 最下位ゴールの分解粒度が粗く統一されていない場合

5.3.3 節の (2) のように最下位ゴールの分解粒度が粗くゴール名が抽象的な状態、もしくは粒度は細かいがゴールモデル間で粒度が統一されていない状態の場合、子ゴール共通性を反映できない可能性がある。提案ルールの効果を発揮させるためには、最下位ゴールをユースケースレベルまで分解するとともに、階層ごとに粒度を定め、それに従ってゴールモデルを作成する必要がある。

b. 「自動判別共通ゴール」の祖父母ゴールに「類似度の低い共通ゴール」が存在する場合

子ゴール共通性を反映することにより、例えば親ゴール同士が「類似度の低い共通ゴール」であってもそれらを正しく判別できる。しかし本ルールでは孫ゴールの共通性までは反映していないため、「自動判別共通ゴール」の祖父母ゴールに「類似度の低い共通ゴール」が存在する場合、それらを正しく判別できない。これにより「自動判別可変ゴール」の適合率が下がる可能性がある。

一方で本ルールを孫ゴールの共通性まで反映できるように拡張すると、今度は「自動判別可変ゴール」が可変でないと判別され「共通ゴールランキング」に出現するゴール数が増加し「自動判別可変ゴール」の適合率が下がる可能性がある。

適合率を 100 % に近づけるためには、例えば再現率が下がったとしても孫ゴールの共通性を反映できるように提案ルールを拡張する必要がある。

6.2 評価実験の妥当性

6.2.1 ゴールモデルの作成者に関する妥当性

本実験では、同一作者が作成したゴールモデル同士の場合においては手法が有効であり、別作者が作成したゴールモデル同士の場合においては手法があまり

有効ではないという結果になった。

このとき、同一作者が作成したゴールモデルは用語・記述方法・分解粒度が統一されており、本研究が想定しているシナリオに合致している。

こういった理由から、本実験においてゴールモデルの作成者を同一人物にしていることは妥当であり、その場合に手法が有効であったという実験結果も妥当なものであると考える。

6.2.2 実験の被験者に関する妥当性

5.3.1 で述べたように、本実験の被験者は著者も含め情報工学を専攻している修士課程の学生であり、妥当性には限界がある。実際の開発者による評価は今後の課題である。

また、ゴールモデルの作成者が著者も含めて 3 人しかいないため客観性の低い評価となっている。もっと多くの人にゴールモデルを作成してもらい実験を重ねることで評価の妥当性を高める必要があると考える。

7 まとめ

本研究では効率よく共通ゴールを判別するための手法を提案し、それを実装したサポートツールを作成した。本手法では 2 つのゴールモデル中からツールが自動で「自動判別共通ゴール」と「自動判別可変ゴール」を判別する。その後、共通か可変かの判断が難しいゴールを「共通ゴールランキング」として出力し、開発者はそれを用いて手作業で共通ゴールの判別を行う。

開発者の労力を削減させるためには、できるだけ多くの共通ゴールと可変ゴールをツールが自動でミスなく判別できる必要がある。ツールはゴール間の類似度に基づいて判別を行うが、ゴールモデル中に「類似度の高い非共通ゴール」と「類似度の低い共通ゴール」が存在するとうまく判別を行うことができない。しかし、それらのゴールはゴールモデルの構造を用いることで見分けることが可能である。そのため、「1 ゴール 1 共通」、「下位ゴール優先」、「子ゴール共通性の反映」というゴールモデルの構造に基づいた 3 つのルールを提案した。

提案ルール及びそれを含めた提案手法全体の有効性を評価するために、様々な既存システムを対象とした

ゴールモデルを用いて実験を行った。テレビ・SNS・就職支援システムのゴールモデルを用いた実験の結果から、提案ルールが有効な場合とそうでない場合を明確化した。またカメラのゴールモデルを用いた実験結果から、提案手法を使うと全て人手で判別を行うよりも早く正確に共通ゴールを判別できることが確認された。

今後、より多くの既存システムに対する実験や、より実現場に沿った方法で実験を行い手法の適用範囲を広げていきたい。

参考文献

- [1] Pohl, K., Bockle, G. and Van Der Linden, F.: *Software Product Line Engineering: Foundations, Principles And Techniques*, Springer-Verlag New York Inc., 2005.
- [2] Chen, L., Babar, M. A. and Lero, N. A.: Variability Management in Software Product Lines: A Systematic Review, in *SPLC2009*, pp. 81–90.
- [3] Asadi, M., Bagheri, E., Gaevi, D., Hatala, M. and Mohabbati, B.: Goal-Driven Software Product Line Engineering, in *SAC2011*, pp. 691–698.
- [4] van Lamsweerde, A.: *Requirements Engineering, From System Goals to UML Models to Software Specifications*, Wiley, 2009.
- [5] Uno, K., Hayashi, S. and Saeki, M.: Constructing Feature Models using Goal Oriented Analysis, in *QSIC2009*, pp. 412–417.
- [6] Yu, Y., Wang, Y., Mylopoulos, J., Liaskos, S., Lapouchnian, A., Cesar, J. and do Prado Leite, S.: Reverse Engineering Goal Models from Legacy Code, in *RE2009*, pp. 363–372.
- [7] Lee, B.-S. and Bryant, B. R.: Contextual natural language processing and DAML for understanding software requirements specifications, in *COLING 2002*, pp. 1–7.
- [8] Assawamekin, N.: An Ontology-Based Approach for Multiperspective Requirements Traceability between Analysis Models, in *ICIS 2010*, pp. 673–678.
- [9] Ilyas, M. and Kung, J.: A Similarity Measurement Framework for Requirements Engineering, in *ICCGI2009*, pp. 31–34.
- [10] Czarnecki, K., Helsen, S. and Eisenecker, U.: Formalizing Cardinality-based Feature Models and their Specialization, *Software Process, Improvement and Practice*, Vol. 10, No. 1(2005), pp. 7–24.
- [11] Kumaki, K., Tsuchiya, R., Washizaki, H. and Fukazawa, Y.: Supporting commonality and variability analysis of requirements and structural models, in *SPLC2012*, pp. 115–118.
- [12] Kang, K., Cohen, S., Hess, J., Novak, W. and Peterson, A.: Feature Oriented Domain Analysis (FODA), Feasibility Study, Technical Report CMU/SEI-90-TR-21, 1990.
- [13] Cheng, B. H. C. and Atlee, J. M.: Research Directions in Requirements Engineering, in *FOSE 2007*, pp. 285–303.
- [14] Berry, M., Daudjee, K., Dong, J., Finchtein, I., Nelson, A., Nelson, T. and Ou L.: User's Manual as a Requirements Specification: Case Studies, in *Requirements Engineering 9.1 2004*, pp. 67–82.
- [15] 中村祐貴, 本田耕三, 中川博之, 田原康之, 大須賀昭彦: ゴールモデルの構造に基づいた共通ゴール判別手法の提案, ソフトウェア工学の基礎 XIX FOSE2012, pp. 63–68.
- [16] 小林幹門:業務文書の品質向上を支援する文書間整合性診断, 東芝レビュー, Vol. 67, No. 8(2012), pp. 56–57.
- [17] 大崎人士, 岡本圭史, 北村崇師, 木下佳樹, 矢田部俊介:仕様書の統一様式の策定と仕様整合性検証システムの開発, SSV2008, pp. 82–85.
- [18] Carnegie Mellon University: Software Product Lines | Overview, <http://www.sei.cmu.edu/productlines/>, 2013.



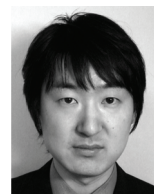
中村 祐貴

1989 年生。2013 年電気通信大学大学院情報システム学研究科社会知能情報学専攻博士前期課程修了。



本田 耕三

1953 年生。1976 年九州大学工学部電気工学科卒業。同年日本電気(株)に入社。2011 年電気通信大学大学院情報システム学研究科博士前期課程修了, 現在, 電気通信大学大学院情報システム学研究科博士後期課程に在学。ゴール指向要求分析・定義などの研究に従事。情報処理学会学生会員。



中川 博之

1974 年生。1997 年大阪大学基礎工学部情報工学科卒業。同年鹿島建設(株)に入社。2007 年東京大学大学院情報理工学系研究科修士課程修了, 2008 年同大学院博士課程中退。同年より電気通信大学助教。現在に至る。博士(工学)(早稲田大学)。要求分析, 形式手法, エージェントおよび自己適応システ

ム開発手法の研究に従事。情報処理学会，電子情報通信学会，IEEE CS 各会員。



田原 康之

1966 年生。1991 年東京大学大学院理学系研究科数学専攻修士課程修了。同年 (株) 東芝入社。1993～1996 年情報処理振興事業協会に出向。1996～1997 年英国 City 大学客員研究員。1997～1998 年英国 Imperial College 客員研究員。2003 年国立情報学研究所着任。2008 年より電気通信大学大学院情報システム学研究科 准教授。現在に至る。博士 (情報科学)(早稲田大学)。エージェント技術，およびソフトウェア工学などの研究に従事。情報処理学会，日本ソフトウェア科学会各会員。



大須賀 昭彦

1958 年生。1981 年上智大学理工学部数学科卒。同年 (株) 東芝入社。同社研究開発センター，ソフトウェア技術センター等に所属。1985～1989 年 (財) 新世代コンピュータ技術開発機構 (ICOT) 出向。2007 年より，電気通信大学大学院情報システム学研究科教授。2012 年より，国立情報学研究所客員教授兼任。工学博士 (早稲田大学)。主としてソフトウェアのためのフォーマルメソッド，エージェント技術の研究に従事。1986 年度情報処理学会論文賞受賞。IEEE Computer Society Japan Chapter Chair，人工知能学会理事，日本ソフトウェア科学会理事を歴任。情報処理学会，電子情報通信学会，人工知能学会，日本ソフトウェア科学会，IEEE Computer Society 各会員。