

ソフトウェア機能提案のための新規要求に類似した既設計仕様の利活用

Utilization of existing design specifications similar to new requirements for software function proposals

山口 智也[†] 島川 博光[†]
Tomoya Yamaguchi Hiromitsu Shimakawa

1. はじめに

アジャイル開発プロセスの導入やテスト自動化の強化、コードレビューの効率化などにより、ソフトウェア開発の期間が短縮化されている。

開発者の労力を縮小化する動きがある一方で、顧客からの要求の獲得には労力を惜しんではならない。しかし、要求時にソフトウェア開発に精通していない顧客から開発者に対し、ソフトウェア要求として絵や図などの曖昧な情報を提示されることが多くある。そのような提示には、要求が明確に書かれていない。簡単な手書きの絵で顧客の要求が指定されたとき、作成すべき機能が詳細に書かれていない可能性が高い。さらに、曖昧な要求が指定された場合、ソフトウェア要求を作成する際に、開発すべき機能を顧客と開発者の間で合意していく必要が出てくるなど無駄な作業が生まれ、費用とコストがかかる。無駄な作業が増えるとソフトウェア開発の効率と生産性が低下する。

本研究では、要求時に提示される曖昧な情報から顧客の要求を抽出し、要求に類似した既設計仕様を活用することで、顧客の要求にない可能性が高いソフトウェア機能を提示する手法を提案する。既設計仕様を活用し、要求に無いソフトウェア機能を提案することで、開発者の負荷を軽減する。

2. 曖昧なソフトウェア要求

2.1 ソフトウェア開発における曖昧さ

顧客の曖昧な要求指定はソフトウェア開発において深刻な問題である。ソフトウェア開発者が要求の曖昧さを認識していたとしても、開発者の解釈次第で顧客の要求とは異なるものが決定されてしまう。この要求認識の差異は、ソフトウェア開発の失敗に繋がるため、曖昧な情報から適切に顧客の要求を汲み取る必要がある。

ソフトウェア開発における曖昧さの原因は「情報の欠如」と「コミュニケーションエラー」という 2 つの原因がある。これらの曖昧さの原因に対処する方法は 2 つある。1 つ目は、曖昧さの防止である。要求に曖昧さが生じないようにするために、正確で一意的意味を持つ文や図を使用して理解しやすい要求を記述できるようにする方法である。2 つ目は、曖昧さを検出し解決する方法である。初期の要求にある曖昧さを許容し、その後、機械学習などの適切な技術を使用して曖昧さを検出し解決する。

2.2 自然言語ソフトウェア要求の曖昧さ解決

ソフトウェア開発の成功は正確な要求仕様に依存する。しかし、曖昧さを含んでいるものは重大な問題を引き起こ

す可能性がある。中でも、言語的曖昧さを含んだソフトウェア要求仕様は、しばしば語彙的、構文的、意味的、語用的なあいまいさを含み、誤解やプロジェクトの失敗を引き起こす可能性がある。そのため、自然言語のソフトウェア要求の曖昧さ解決に向けて、曖昧さ自動検出に向けた手法が研究されている。しかし、自然言語だけでなく図や絵のような曖昧な情報を含んだソフトウェア要求を提示されることもある。

3. 要求抽出と機能提案手法

3.1 全体の流れ

本研究では、自然言語のソフトウェア要求ではなく、視覚情報であるソフトウェア画面から要求を抽出し、既設計仕様と比較することで、顧客の要求に類似した新規機能を提案する。

本研究では、要求が曖昧な手書きの絵や図で表現されたソフトウェア画面の画像データを物体認識モデルである Faster-RCNN を用いて機能の要素を検知し、顧客の要求に無い機能を提案する。物体認識とは、画像の中から特定の物体の位置、個数、ラベルを認識する技術である。本研究で、扱うソフトウェア画面の種類はログイン画面に限定し、要求抽出と新規機能を提案する。本研究の要求抽出と機能提案の手法概要図を図 1 に示す。

検知した要求であるソフトウェア機能を、WEB 上に存在する既設計のログイン画面と比較し、要求が曖昧な図や手書きの絵にはないソフトウェア機能を認識する。顧客の要求にはなく、他の類似したログイン画面にはある機能をテキスト変換することで、取り入れるべき機能を提案する。

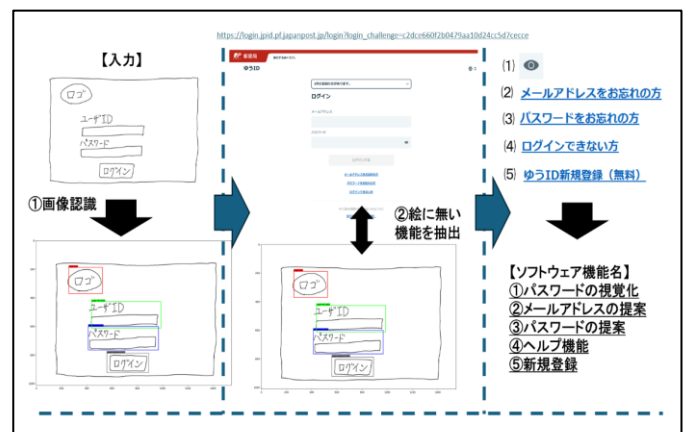


図 1 手法概要図

3.2 ログイン画面の認識

入力を手書きや図などの要求が曖昧なログイン画面の画像データとし、Faster-RCNN を用いてログイン画面を構成

[†]立命館大学 Ritsumeikan University

する各機能に相当する領域を検知する。今回扱うログイン画面の画像データの内、表 1 に示した既設計のログイン画面にある 9 つの機能を指定し、その機能を検出するように学習させる。

Faster-RCNN は各機能に対応するクラスラベルと確信度を表す Confidence Score、そしてバウンディングボックスの座標を出力する。この出力を使って、要求が曖昧なログイン画面の画像データに含まれるソフトウェア機能の領域を視覚的に捉える。図 2 は、要求が曖昧な手書きの絵に含まれるソフトウェア機能を Faster-RCNN を使って認識させたものである。この手書きの絵には、「ロゴの表示」と「ログイン ID 入力」、「パスワード入力」、「ログイン実行」という機能が含まれており、機能別に色分けされたバウンディングボックスの上部に表示されている数値が認識された領域がそれぞれの機能である確信度を表している。

クラス名	機能
Logo	ロゴの表示
Login ID	ログイン ID 入力ボックス
Password	パスワード入力ボックス
Login Button	ログイン実行ボタン
forget	パスワードを忘れた方への対処
Make Account	新規登録
Visualize Pass	パスワードの可視化
Help	ヘルプ機能
SNS Login	SNS によるログイン

表 1 ログイン画面上の機能

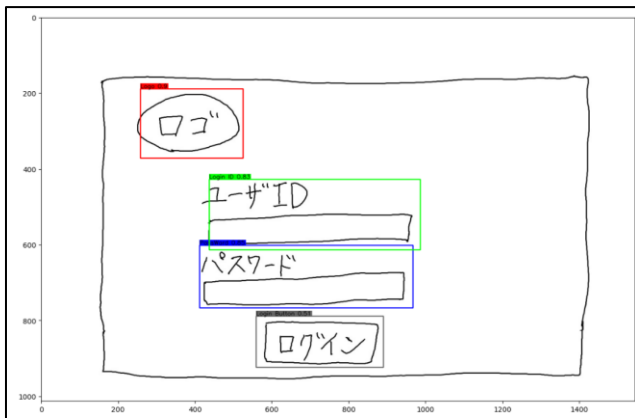


図 2 ログイン画面の機能認識

3.3 手書きの絵に無いソフトウェア機能を抽出

要求が曖昧なログイン画面と既設計仕様を比較し、手書きの絵に無いソフトウェア機能を抽出する。要求が曖昧なログイン画面画像にはないソフトウェア機能を抽出するには、Confidence Score に閾値を設定し、確信度の大きさをソフトウェア機能の有無を判定する。

Confidence Score の閾値を 0.5 と設定し、0.5 以上と認識されたソフトウェア機能は要求に含まれていると判定する。Confidence Score が 0.5 以下と認識されたものを要求に含ま

れていないソフトウェア機能とし、要求に含まれていないと判定した機能を新規機能として抽出する。

図 2 で認識されなかった機能を、既設計仕様には含まれていて、顧客の要求に含まれていない機能推定し抽出する。

3.4 ソフトウェア機能のテキスト変換

抽出した要求が曖昧なログイン画面画像に含まれていない機能をテキストとして出力する。機能名に関するラベルを顧客が理解しやすい具体的なテキストや要求仕様書に近いテキストに変換することで、要求に類似した新機能を提案する。

4. おわりに

本研究では、ソフトウェア開発における曖昧な情報から、既設計ソフトウェアを活用して、顧客の要求に類似した機能を提案することを考えている。

本研究により新規要求に類似した機能提案が可能になれば、その提案を基にソフトウェア開発を進めることができ、時間短縮や労力の軽減に繋がる。更に既設計ソフトウェアの画面ではなく、ソフトウェア仕様書を比較対象とすることで、過去の仕様書に基づいた機能提案を実現できる。

参考文献

- [1] Todd Sedano, Paul Ralph, and Cécile Péraire. 2017. Software development waste. In Proceedings of the 39th International Conference on Software Engineering (ICSE '17). IEEE Press, 130–140. <https://doi.org/10.1109/ICSE.2017.20>
- [2] Berry, Daniel M., Erik Kamsties and Michael Krieger. "From Contract Drafting to Software Specification: Linguistic Sources of Ambiguity." (2003).
- [3] Unnati S. Shah and Devesh C. Jinwala. 2015. Resolving Ambiguities in Natural Language Software Requirements: A Comprehensive Survey. SIGSOFT Softw. Eng. Notes 40, 5 (September 2015), 1–7. <https://doi.org/10.1145/2815021.2815032>
- [4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'15). MIT Press, Cambridge, MA, USA, 91–99.
- [5] Kamsties, Erik. (2001). Taming Ambiguity in Natural Language Requirements. Fraunhofer IESE.
- [6] Abdelhamid, Abdelaziz A., Sultan Refa Alotaibi and Abdel Salam Jad Bassiouni Mousa. "Deep learning-based prototyping of android GUI from hand-drawn mockups." IET Softw. 14 (2020): 816-824.
- [7] M Syauqi Haris and Tri Astoto Kurniawan. 2021. Automated requirement sentences extraction from software requirement specification document. In Proceedings of the 5th International Conference on Sustainable Information Engineering and Technology (SIET '20). Association for Computing Machinery, New York, NY, USA, 142–147. <https://doi.org/10.1145/3427423.3427450>
- [8] ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes --Requirements engineering," in ISO/IEC/IEEE 29148:2011(E), vol., no., pp.1-94, 1 Dec. 2011, doi: 10.1109/IEEESTD.2011.6146379.