社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

Stack Overflowと言語ドキュメントの紐づけ手法の検討

鬼塚　仙太郎†　神田　哲也†　眞鍋　雄貴††　肥後　芳樹†

† 大阪大学大学院情報科学研究科　〒565–0871 大阪府吹田市山田丘 1–5
†† 福知山公立大学情報学部情報学科　〒620–0886 京都府福知山市字堀 3370
E-mail: †{s-onizuk,t-kanda,higo}@ist.osaka-u.ac.jp, ††manabe-yuki@fukuchiyama.ac.jp

あらまし　プログラミング学習のリソースとして Stack Overflow があり，プログラミング学習中に直面した課題の解決において有益なリソースである．一方で，プログラミング学習者が投稿内容を十分に理解できない可能性がある．特に，プログラミング言語の基本機能が投稿の理解を阻む場合は，プログラミング学習者がその事象が原因であると気づけない場合がある．そこで，このような事項の理解を助けるために，Stack Overflow の投稿とプログラミング言語のドキュメント（以降言語ドキュメント）を紐づけることを考える．Stack Overflow の投稿には質問のトピックを説明するタグが存在する．もしタグが本文に出現する要素を特徴づけたものになっていれば，タグを用いて言語ドキュメントとの紐づけを行える可能性がある．本研究では，それを実現する手法の検討として，Stack Overflow におけるタグと記載内容間の関係について言語ドキュメントに登場する語を対象として調査した．その結果，本文を中心に解析して投稿に関連するプログラミング言語の基本機能を抽出することと，言語ドキュメントから紐づけに使用する単語を適切に選択することが課題となることがわかった．

キーワード　Stack Overflow，プログラミング言語，言語ドキュメント，プログラミング学習

An exploratory study of linking between Stack Overflow and Language Documentation

Sentaro ONIZUKA†, Tetsuya KANDA†, Yuki MANABE††, and Yoshiki HIGO†

† Graduate School of Information Science and Technology, Osaka University
1–5 Yamadaoka, Suita, Osaka, 565–0871 Japan
†† Faculty of Informatics, The University of Fukuchiyama　Hori, Fukuchiyama, Kyoto, 620–0886 Japan
E-mail: †{s-onizuk,t-kanda,higo}@ist.osaka-u.ac.jp, ††manabe-yuki@fukuchiyama.ac.jp

Abstract　Stack Overflow(SO) is a programming learning resource that is useful in solving issues faced while learning programming. On the other hand, there is a possibility that programming learners may not fully understand the posts on SO. In particular, if a basic feature of the programming language prevents the understanding of a post, the programming learner may not be able to recognize the cause of the event. Therefore, we consider linking posts on SO to programming language documentation (language documentation) to aid understanding of such matters. SO posts contain tags that describe the topic of the question. If the tags are characterized by elements that appear in the body of the text, it may be possible to use the tags to link them to the language documentation. In this study, we investigated the relationship between tags and content in SO, focusing on words that appear in language documentation. As a result, it was found that the main issues are analyzing the text and extracting the basic functions of the programming language related to the post, and appropriately selecting words from the language documentation for linking.

Key words　Stack Overflow, Programming language, Language documentation, Programming learning

1. ま え が き

プログラミング学習者は多くのウェブリソースに触れており，その例として Stack Overflow [1] がある．Stack Overflow はプログラミングに特化した Q&A サイトであり，そこにはプログラマからの質問に対して他のプログラマが投稿した解決策や例が回答として掲載されている．

Stack Overflow はプログラミング学習者にとって有益なリ

ソースとなっており，課題に直面した際に Stack Overflow を利用してその課題に対処することも多い．過去に行われた調査でも，学生が課題に役に立ちそうなリソースをウェブ上で探す際に Stack Overflow を利用する可能性が高いことがわかっている [2]〜[4].

しかし，Stack Overflow に掲載されている投稿内容を，投稿に含まれる説明が不十分であるためにプログラミング学習者が持っている知識では十分に理解できない可能性がある．そのような問題を解決するために Stack Overflow を関連する外部リソースと紐づける研究が存在する．具体的には，Stack Overflow の投稿から API ドキュメントへのリンクを作成する研究がある [5]．一方で，API 呼び出しそのものでなく，それに付随して使用されている制御構造やプログラミング言語標準のデータ構造などのプログラミング言語の基本機能が投稿を理解するために必要な知識であることも考えられる．

このような事項の理解を助けるために，Stack Overflow の投稿とプログラミング言語のドキュメント（以降言語ドキュメント）を紐づけることを考える．Stack Overflow の投稿にはコード片も含まれるが，質問に含まれるコード片は問題を抱えているものであったり，回答に含まれるコード片は修正内容のみを抽出した部分的なものであるなど，構文解析が容易ではない．また，Stack Overflow の投稿と言語ドキュメントを紐づけるという観点から，大量の投稿を処理するためには構文解析の成否に関係なく分析を行う仕組みを導入したい．そのため，Stack Overflow と言語ドキュメントに含まれる自然言語の文章を解析して紐づけする手法を検討する．

Stack Overflow 側に含まれる自然言語で書かれた文章としては，タイトル，質問，回答，質問のコメント，回答のコメントのコードを除くテキスト（以上をまとめて本稿では「本文」とする）のほかに，質問のトピックを説明するタグが存在する．もしタグが本文に出現する要素を特徴づけたものになっていれば，タグを用いて言語ドキュメントとの紐づけを行える可能性がある．そこで，本研究では Stack Overflow の投稿につけられているタグのうち，プログラミング言語の基本機能に関するタグは本文に出現する特徴的な要素であるか，およびプログラミング言語の基本機能に関するタグが本文を特徴づけるものになっているか，言語ドキュメントに登場する単語を対象として調査した．調査のために，以下の 2 つのリサーチクエスチョンを設定した．

RQ1：プログラミング言語の基本機能に関するタグは本文の頻出単語であるのか

プログラミング言語の基本機能に関するタグが本文の頻出単語である割合と，タグと関連性の低い単語が頻出単語である割合に関して調査する．これにより，タグは本文に出現する特徴的な要素であるかを明らかにする．

RQ2：本文のプログラミング言語の基本機能に関する頻出単語はタグとして付与されるのか

本文の頻出単語がタグとして付与される割合と，本文の頻出単語と関連性の低い単語がタグとして付与される割合に関して調査する．これにより，本文中のプログラミング言語の基本機能に関する頻出単語がタグとして選ばれているのか，タグが本文に出現する要素を特徴づけたものになっているのかを明らかにする．

RQ1 と RQ2 の結果から，言語ドキュメントに登場する単語において，Stack Overflow と言語ドキュメントの紐づけ手法に適用できるほどタグと本文では用語が一致していないことがわかった．RQ1 ではタグが本文の頻出単語である割合は，タグと関連性の低い単語が頻出単語である割合と比較して高いことがわかったが，そもそも頻出単語でなかった投稿もタグにより 4–8 割程度存在した．RQ2 では，本文の頻出単語がタグとして付与される割合は，本文の頻出単語と関連性の低い単語がタグとして付与される割合と比較して高いことがわかったが，その割合は最大でも 10.32% であり低かった．

以降，2 章では研究背景や関連研究について述べる．3 章では本研究の調査方法について述べ，4 章では得られた結果，5 章では考察を述べる．最後に 6 章では本研究のまとめと今後の課題を述べる．

2. 背　　景

プログラミングに特化した Q&A サイトとして Stack Overflow があり，プログラミング学習において課題に直面した際に Stack Overflow を利用する人は多い [2]〜[4]．ただし，プログラミング学習者は Stack Overflow の投稿内容を十分に理解できない可能性があり，その場合は投稿内容に関連する別のリソースをさらに参照する必要がある．特に，プログラミング言語の基本機能はプログラミング学習の初期段階で学ぶ事項であるため，そのような事項に関する記述が多い場合に，投稿を理解するための追加情報を自ら検索することも難しいと考えられる．

本章では，まず Stack Overflow について説明し，さらに Stack Overflow 利用時における問題とそれに対処することを目的とした既存研究について紹介する．そして，それらをもとに本研究で調査する Stack Overflow と言語ドキュメントの紐づけについて既存研究とともに説明する．

2.1 Stack Overflow

Stack Overflow は 2008 年に設立されたプログラミングに関する質問とそれに対する回答を投稿・検索できる Q&A コミュニティサイトである．Q&A コミュニティサイトとは，ある質問者が行った質問に対して不特定多数の回答者が回答を行うサイトである．Stack Overflow における質問の例を図 1 に示す．Stack Overflow における投稿は，タイトル，質問，複数個のタグ，そして質問に対する複数の回答で構成される．また，各質問及び回答にはコメントを付与することができる．質問及び回答では，自然言語による記述だけでなくコードを組み合わせた記述が可能である．タグとは，質問のトピックを説明する単語またはフレーズであり興味のある質問を特定するためにも使用される．タグとして “Python” 等の言語名や，“If-statement” のような文法事項，“pandas” のようなライブラリ名などが登録されている．Stack Overflow は 2023 年 6 月現在，ユーザ数は約 2,100 万人，投稿は質問と回答を合わせて約 5,900 万件，そして 1 日に約 590 万人が閲覧しており広く普及していることがわかる [6].

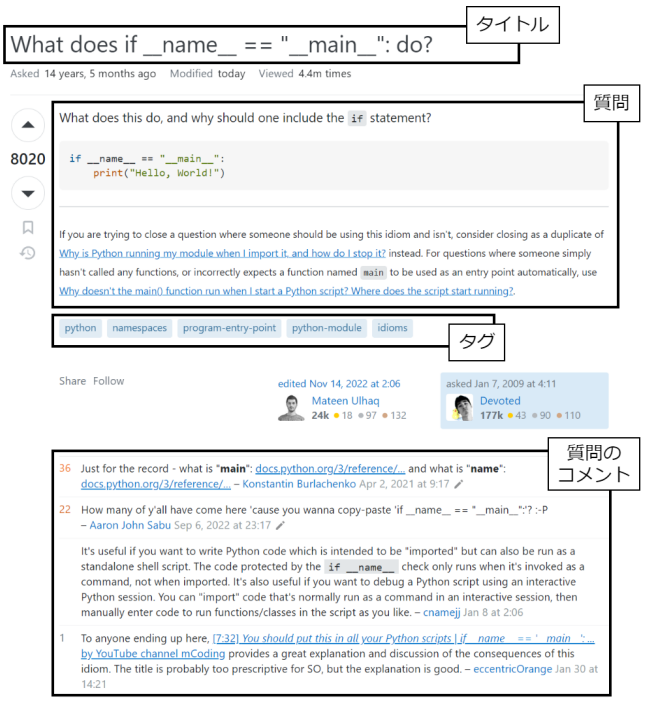

図 1　Stack Overflow における質問の例（ID：419163）[1]

Stack Overflow はプログラミング学習にも多く利用されている．Stack Overflow Developer Survey 2022 [2] によると，コーディング方法を学ぶためのオンラインリソースとして Stack Overflow を使用すると答えた人の割合は 86.14% であり，技術文書と並んで最も利用されているリソースの一つである．Parnin ら [3] は，プログラミングに関するクエリにおいて，Stack Overflow 上の投稿が Google の検索結果リストの上位に高い確率で表示されることを示している．Robinson [4] は，大学生が Stack Overflow に投稿した質問の割合が授業期間に大幅に増加し休暇期間に低下することを示している．

2.2　Stack Overflow 利用時における問題点

Stack Overflow はプログラミング学習者にとって有益なリソースである一方で，投稿に含まれる説明が不十分であるためにプログラミング学習者の知識ではその投稿内容を十分に理解できない可能性がある．そのため，Stack Overflow に加えて関連するリソースを活用する研究もいくつか行われている．例えば，Stack Overflow 上のコードスニペットから API 要素を特定し，API ドキュメントへのリンクを作成する研究がある [5]．また，Stack Overflow 上のコードスニペットから，非推奨となっている API を検出するフレームワークが提案されている [7]．これらは API に関する手法であるため自然言語の解析をせずコード部分のみの解析で実現できている．

Stack Overflow の投稿にはコード片も含まれるが，質問に含まれるコード片は問題を抱えているものであったり，回答に含まれるコード片は修正内容のみを抽出した部分的なものであるなど，構文解析が容易ではない．また，Stack Overflow の投稿と言語ドキュメントを紐づけるという観点から，大量の投稿を処理するためには構文解析の成否に関係なく分析を行う仕組みを導入したい．そのため，Stack Overflow の投稿と言語ドキュメントの紐づけでは，それぞれに含まれる自然言語の文章を解析して紐づけする手法を検討する．

2.3　プログラミング言語の基本機能に関する紐づけ

2.2 節で説明したように，Stack Overflow 上の投稿と言語ドキュメントを紐づけるには投稿に含まれる自然言語で書かれた文章を考慮する必要がある．

Stack Overflow 上の自然言語に基づいて関連するリソースとの紐づけを実現している研究として，タグとその説明文に基づいて Stack Overflow 上のタグを Wikipedia の記事にマッピングする手法を提案したものがある [8]．言語ドキュメントへの紐づけと Wikipedia の記事への紐づけは両者ともに外部リソースへの紐づけである点で似ているが，プログラミング言語の基本機能における紐づけは Wikipedia では不十分である．なぜなら，Wikipedia ではプログラミング言語の基本機能の概要を知ることはできても，各プログラミング言語特有の情報を知ることはできない．例えば，基本機能の一つである “Stack” に該当する Wikipedia の記事[2] には，プログラミング言語 LISP 及び Java のコードの例や PHP の Stack に該当するライブラリのリンクが含まれているが，Python の Stack に関する情報はない．また，Python のチュートリアルには “range 関数” という項目が存在するが，Wikipedia にはその関数に対応する記事が存在しない．

Stack Overflow 上の本文と言語ドキュメントの内容に注目した研究として，Python における For 文と If 文を対象に Stack Overflow と言語ドキュメントそれぞれに登場する名詞集合間の類似度を調査したものがある [9]．この研究では，Stack Overflow 上の “If-statement” 及び “Python” タグが付いた質問と Python チュートリアルの “4.1. if Statements” 項目，Stack Overflow 上の “For-loop” 及び “Python” タグが付いた質問の本文と Python チュートリアルの “4.2. for Statements” 項目を対象として，Stack Overflow と言語ドキュメントそれぞれの類似したトピック間で同じ名詞が登場しているかを調査している．そして，For 文の場合はどちらにも対応しうる名詞が現れたものの，If 文の場合は現れなかった結果となっている．この調査結果から，Stack Overflow と言語ドキュメントの紐づけにおいては単純なキーワードマッチングでは達成できない可能性が高いことが予想される．

3.　調　　査

3.1　概　　要

本研究では Stack Overflow と言語ドキュメントの紐づけ手法について調査する．プログラミング言語 Python を対象とし，言語ドキュメントとして “The Python Tutorial” [10] を対象とする．具体的には，紐づけ手法の検討として，Stack Overflow の投稿につけられているタグのうち，プログラミング言語の基本機能に関するタグは本文に出現する特徴的な要素であるか，およびプログラミング言語の基本機能に関するタグが本文を特徴づけ

(注1)：https://stackoverflow.com/questions/419163/what-does-if-name-main-do

(注2)：https://en.wikipedia.org/wiki/Stack_(abstract_data_type)

表 1　調査対象の項目と単語

Python チュートリアルの項目	調査対象の単語
4.2. for Statements	iteration
5.4. Sets	set
8.2. Exceptions	exception

るものになっているか，言語ドキュメントに登場する単語を対象として調査した．調査では，関連性の高い単語間での調査だけでなく関連性の低い単語間でもタグと本文の関係を調査することで，その関係が単語間の関連性によって異なる結果となるかを確かめる．本調査では以下の 2 つの RQ を設定した．

- **RQ1：プログラミング言語の基本機能に関するタグは本文の頻出単語であるのか**

調査対象のタグが A であり，本文における調査対象の単語が B であるとする．Stack Overflow 上の “Python” タグと A タグが付いた投稿を対象として，B（または B 及び B の類義語）が本文に頻繁に登場するかを調べる．

- **RQ2：本文のプログラミング言語の基本機能に関する頻出単語はタグとして付与されるのか**

本文における調査対象の単語が A であり，調査対象のタグが B であるとする．Stack Overflow 上の “Python” タグが付いた質問を対象として，A が本文中の頻出単語である質問に B（または B 及び B の類義語）がタグとして付与されているかを調べる．

調査対象の言語ドキュメント中にある単語として，Python チュートリアルにおける 3 つの項目からそれぞれ 1 つ単語を選択した．選択した項目と単語を表 1 に示す．Stack Overflow と言語ドキュメントそれぞれに登場する名詞集合間の類似度を調査した既存研究 [9] に従い，調査対象の Python チュートリアルにおける項目として “4.2. for Statements” を選択した．そして，対象とする単語の種類を多様化するために，その他の項目としては別のカテゴリーに属しているものから 2 つ選択した．調査対象の単語は，各項目で使用されていた名詞からその項目に関連していて，かつ Stack Overflow のタグとして存在する単語を選択した．

調査において，以下の手順により StackOverflow 上の質問を分析する．Stack Overflow のタグや本文などのデータへのアクセスには SOTorrent [11] を利用した．SOTorrent は，Stack Overflow の公式データダンプをもとに構築されたオープンデータセットであり，変更履歴にテキストブロックやコードブロックの単位でアクセスすることができる．また，単語の登場回数を調べるには自然言語処理のツールキットである Python ライブラリの NLTK [12] を用いた．RQ1 と RQ2 に対する調査における具体的な操作に関しては，各 RQ について細部が異なるため，それぞれ 3.2 節，3.3 節に示す．

- STEP1：タグのついた投稿の抽出
- STEP2：各要素（質問，回答，質問のコメント，回答のコメント，タイトル）での調査対象の単語の登場回数をカウント
- STEP3：調査目的に該当する投稿を抽出

STEP2 において，語句の完全一致だけでなく，類義語を用いて回数をカウントすることも行う．類義語の判定には，大規模コーパスを用いた類似検索が可能な Python ライブラリである Gensim [13] と，事前学習済みモデルである “fasttext-wiki-news-subwords-300” を用いた．本モデルの選択理由は，このモデルがその他のモデルと比較して，プログラミングにおいて関連性の高い単語である “iteration” と “loop” の類似度が一番高かったためである．“iteration” と “loop” の類似度に関してその他のモデルと比較した表を表 2 に示す．本ライブラリとモデルによって算出された類似度が 0.5 以上の単語を類義語として選択する．これは，“iteration” と “loop” の類似度が 0.54 だったためである．また，類義語における登場回数は「類似度×出現回数」を加算した結果を用いる．例えば，“iteration and loop are important for loop.” という文において，“iteration” の登場回数は 1 であるが，類義語を含めて登場回数を計算する場合，類似度が 0.5 を超える “loop” も出現回数に加算され「$1.0 + 0.54 \times 2 = 2.08$」と計算される．

表 2　事前学習済みモデルの比較

モデル名	“iteration” と “loop” の類似度
fasttext-wiki-news-subwords-300	0.54
word2vec-google-news-300	0.31
word2vec-google-news-300	0.19
glove-twitter-200	0.15
conceptnet-numberbatch-17-06-300	×
word2vec-ruscorpora-300	×

本調査で選択した 3 つの単語から 2 つを選ぶ組み合わせ（タグとして 1 つと本文における調査対象の単語として 1 つ）は計 6 通りあり，類義語を含む場合と含まない場合でさらに 2 通りある．つまり，RQ1 に対応する調査と RQ2 に対応する調査ではそれぞれ計 12 通りの調査を行った．また，登場回数の閾値 n（n 回以上登場していれば頻出単語であると判定する）は 1 から 30 まで変更した．ただし，登場回数の閾値の変更による調査結果の傾向は大きく変わらなかったため，4. 章の調査結果では $n = 2$ の場合の結果のみを示す．

3.2　RQ1 に対応する調査

調査対象のタグが A であり，本文における調査対象の単語が B である場合を例に各 STEP での操作について説明する．

STEP1：タグのついた投稿の抽出

SOTorrent から “Python” タグと A タグが付いた投稿のみを抽出する．

STEP2：各要素での調査対象の単語の登場回数をカウント

STEP1 で抽出された各投稿に対して，その各要素（質問，回答，質問のコメント，回答のコメント，タイトル）での B（または B 及び B の類義語）の登場回数をカウントする．

STEP3：調査目的に該当する投稿を抽出

STEP2 で得られた投稿ごとの各要素（質問，回答，質問のコメント，回答のコメント，タイトル）での調査対象の単語の登場回数のカウント結果から，登場回数が閾値 n 以上の投稿数をカウントする．結果として，各閾値 n に対する，質問のみを対象とした場合の投稿数，回答のみを対象とした場合の投稿数，

質問のコメントのみを対象とした場合の投稿数，回答のコメントのみを対象とした場合の投稿数，タイトルのみを対象とした場合の投稿数，すべての要素を対象とした場合の投稿数が得られる．

3.3 RQ2 に対応する調査

本文における調査対象の単語が A であり，調査対象のタグが B である場合を例に各 STEP での操作について説明する．

STEP1：タグのついた投稿の抽出

SOTorrent から “Python” タグが付いた投稿のみを抽出する．

STEP2：各要素での調査対象の単語の登場回数をカウント

STEP1 で抽出された各投稿に対して，その各要素（質問，回答，質問のコメント，回答のコメント，タイトル）での A の登場回数をカウントする．

STEP3：調査目的に該当する投稿を抽出

STEP2 で得られた投稿ごとの各要素（質問，回答，質問のコメント，回答のコメント，タイトル）での調査対象の単語の登場回数のカウント結果から，登場回数が閾値 n 以上の投稿を抽出し，その投稿に付与されているタグに B（または B 及び B の類義語）が含まれるかを調べる．結果として，各閾値 n に対する，質問のみを対象とした場合の投稿数，回答のみを対象とした場合の投稿数，質問のコメントのみを対象とした場合の投稿数，回答のコメントのみを対象とした場合の投稿数，タイトルのみを対象とした場合の投稿数，すべての要素を対象とした場合の投稿数が得られる．

4. 調 査 結 果

4.1 RQ1：プログラミング言語の基本機能に関するタグは本文の頻出単語であるのか

RQ1 に対応する調査について，類義語を含めず調査した結果を表 3 に，類義語を含めて調査した結果を表 4 に示す．

類義語を含まない場合の結果を見ると，タグが本文の頻出単語である割合が高く，タグと関連性の低い単語が頻出単語である割合が低いことがわかる．ただし，“iteration” においては 16.35% であり，“set” と “exception” と比較してその差は小さいことがわかる．類義語を含んだ場合の結果を見ると，タグが本文の頻出単語である割合が高くなる一方，タグと関連性の低い単語が頻出単語である割合も同程度に高くなることがわかる．場合によってはタグと関連性の低い単語のほうが割合が高いことがわかる．

つまり，タグが本文の頻出単語である割合は，タグと関連性の低い単語が頻出単語である割合と比較して高いことがわかる．一方，類義語も含めた場合には，タグと関連性の低い単語が頻出単語である割合が，タグが本文の頻出単語である割合と同程度であることがわかる．

4.2 RQ2：本文のプログラミング言語の基本機能に関する頻出単語はタグとして付与されるのか

RQ2 に対応する調査について，類義語を含めず調査した結果を表 5 に，類義語を含めて調査した結果を表 6 に示す．

類義語を含まない場合の結果を見ると，本文の頻出単語である単語は，本文の頻出単語である単語と関連性の低い単語と比較してタグとして付与される割合が高いことがわかる．ただし，本文の頻出単語である単語がタグとして付与される割合は最大で 10.32% である．類義語を含んだ場合の結果を見ると，本文の頻出単語である単語がタグとして付与される割合が高くなる一方で，本文の頻出単語と関連性の低い単語がタグとして付与される割合も高くなることがわかる．さらに，本文の頻出単語と関連性の低い単語のほうが割合が高くなることもある．

つまり，本文の頻出単語がタグとして付与される割合は，本文の頻出単語と関連性の低い単語がタグとして付与される割合と比較して高いが，その割合は最大で 10.32% である．類義語も含めた場合には，本文の頻出単語である単語がタグとして付与される割合が高くなる一方で，本文の頻出単語と関連性の低い単語のほうが割合が高くなることもある．

表 3　タグ A が付いた投稿で単語 B が本文の頻出単語である割合（%）（類義語含まない，$n = 2$）

タグ＼本文	iteration	set	exception
iteration	16.35	4.14	0.04
set	2.39	69.05	0.49
exception	0.81	0.62	65.95

表 4　タグ A が付いた投稿で単語 B が本文の頻出単語である割合（%）（類義語含む，$n = 2$）

タグ＼本文	iteration	set	exception
iteration	79.76	83.5	82.89
set	53.59	87.67	82.72
exception	36.84	77.83	88.16

表 5　単語 A が本文の頻出単語である場合に単語 B がタグとして付与される割合（%）（類義語含まない，$n = 2$）

タグ＼本文	iteration	set	exception
iteration	3.95	0.52	0.23
set	0.42	6.27	0.07
exception	0.13	0.06	10.32

表 6　単語 A が本文の頻出単語である場合に単語 B がタグとして付与される割合（%）（類義語含む，$n = 2$）

タグ＼本文	iteration	set	exception
iteration	21.25	1.76	0.47
set	10.37	7.21	0.65
exception	05.58	0.75	14.09

5. 考　　察

RQ1 の結果から，タグが本文の頻出単語である割合は，タグと関連性の低い単語が頻出単語である割合と比較して高いことがわかった．しかし，その割合は最大でも 65.95% でありタグが本文の頻出単語である割合は高いとはいえない．また，

“iteration” に関しては 16.35% であり他の単語と比較して非常に小さい結果となっていることから，タグが本文の頻出単語である割合が非常に低くなるケースも珍しくないことが予想される．さらに，類義語を含めて調査したところ，調査対象の単語の組み合わせによる結果の違いがみられなくなり，タグは本文に出現する特徴的な要素であるとはこの結果からは言えない．

RQ2 の結果から，本文の頻出単語がタグとして付与される割合は，本文の頻出単語と関連性の低い単語がタグとして付与される割合と比較して高いことがわかった．しかし，その割合は最大でも 10.32% であり本文の頻出単語がタグとして付与される割合は低い．類義語を含めて調査しても傾向は変わらなかったことから，本文に出現するプログラミング言語の基本機能に関する単語が，タグとして付与される可能性は低いと言える．そこで，他にどのようなタグが多く付けられているかを調べたところ，“pandas” や “numpy” といったライブラリに関するタグが上位を占めておりプログラミング言語の基本機能に関するタグが付けらることは少なかった．これが，ある質問がプログラミング言語の基本機能にどれほど深く関係しているかをタグから判別することを難しくしている可能性がある．

RQ1 と RQ2 の結果から，タグは Stack Overflow の投稿と言語ドキュメントを結びつける要素としては適切でないと言える．そのため，言語ドキュメントとの紐づけのためには，本文を中心に解析を行い，投稿に関連するプログラミング言語の基本機能を抽出することが必要であると考えられる．

本調査では表 1 に示した言語ドキュメントの 3 つの項目から調査対象の単語を各 1 個選択した．ただし，調査対象の単語の選択条件として，項目に関連していてかつ Stack Overflow のタグとして存在する必要がある．そのため，本調査で選択した項目のようにどの項目においても調査対象となる単語を抽出できるわけではない．本調査では “4.1. if Statements” や “4.5. pass Statements” において調査対象となる単語を選択することができなかった．これは，言語ドキュメントの項目には Stack Overflow との紐づけにおいて役に立つ単語が出現しない可能性があることを示唆しており，言語ドキュメントの解析には工夫が必要であることが予想される．ただし，本調査においては名詞のみを対象としたため他の品詞も考慮することで改善される可能性がある．

6. まとめと今後の課題

本稿では，Stack Overflow の投稿につけられているタグのうち，プログラミング言語の基本機能に関するタグは本文に出現する特徴的な要素であるか，およびプログラミング言語の基本機能に関するタグが本文を特徴づけるものになっているか，言語ドキュメントに登場する単語を対象として調査した．RQ1 ではタグが本文の頻出単語である割合は，タグと関連性の低い単語が頻出単語である割合と比較して高いことがわかったが，そもそも頻出単語でなかった投稿もタグにより 4–8 割程度存在した．RQ2 では，本文の頻出単語がタグとして付与される割合は，本文の頻出単語と関連性の低い単語がタグとして付与される割合と比較して高いことがわかったが，その割合は最大でも 10.32% であり低かった．このような結果から，言語ドキュメントに登場する単語において，Stack Overflow と言語ドキュメントの紐づけ手法に適用できるほどタグと本文では用語が一致していないことがわかった．そのため，Stack Overflow 上のタグや本文に登場する単語をそのまま用いて言語ドキュメントに紐づけるような単純な方法では実現することが難しいと考えられ，工夫が必要であると考えられる．

今後の課題としては，本文を中心に解析し，投稿に関連するプログラミング言語の基本機能を抽出することが挙げられる．また，言語ドキュメントから紐づけに使用する単語を適切に選択する手法も課題である．

謝辞 本研究は JSPS 科研費（JP19K20239，JP21K02862, JP20H04166, JP21K18302, JP21K11829, JP21H04877, JP22H03567, JP22K11985, JP23H03375）の助成を得て行われた．

文　献

[1] “Stack overflow,” https://stackoverflow.com/, Accessed: 2023-06-21.
[2] StackOverflow, “Stack overflow developer survey 2022,” https://survey.stackoverflow.co/2022, Accessed: 2023-06-15.
[3] C. Parnin and C. Treude, “Measuring api documentation on the web,” Proceedings of the 2nd International Workshop on Web 2.0 for Software Engineering, pp.25–30, 2011.
[4] D. Robinson, “How do students use stack overflow? - stack overflow blog,” https://stackoverflow.blog/2017/02/15/how-do-students-usestack-overflow/, Accessed: 2023-06-08.
[5] S. Subramanian, L. Inozemtseva, and R. Holmes, “Live api documentation,” Proceedings of the 36th International Conference on Software Engineering, pp.643–652, 2014.
[6] “All sites - stack exchange,” https://stackexchange.com/sites?view=list#traffic, Accessed: 2023-06-15.
[7] J. Zhou and R.J. Walker, “Api deprecation: A retrospective analysis and detection method for code examples on the web,” Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp.266–277, 2016.
[8] A. Joorabchi, M. English, and A.E. Mahdi, “Automatic mapping of user tags to wikipedia concepts,” J. Inf. Sci., vol.41, no.5, pp.570–583, oct 2015.
[9] 眞鍋雄貴，“プログラミング学習における状況に応じた学習要素の提示方法の検討,” 信学技報，第 122 巻，pp.43–48，jul 2022.
[10] “The python tutorial — python 3.11.4 documentation,” https://docs.python.org/3/tutorial/，Accessed: 2023-06-08.
[11] S. Baltes, L. Dumani, C. Treude, and S. Diehl, “Sotorrent: Reconstructing and analyzing the evolution of stack overflow posts,” Proceedings of the 15th International Conference on Mining Software Repositories, pp.319–330, 2018.
[12] S. Bird, E. Klein, and E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, O’Reilly Media Inc, 2009.
[13] R. Řehůřek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp.45–50, 2010.