

# 識別子の読解を目的とした名詞辞書の作成方法の一試案

藤木 哲也<sup>†1</sup> 早瀬 康裕<sup>†1</sup> 井上 克郎<sup>†1</sup>

本稿では、識別子の読解を支援するために、識別子に用いられる単語の辞書を自動的に作成する手法を提案する。作成する辞書には、識別子に用いられる名詞の説明文が収録される。説明文の作成には、プログラム中に記述された識別子名とコメントを利用する。

## Towards Noun Lexicon for Identifier Comprehension

TETSUYA FUJIKI,<sup>†1</sup> YASUHIRO HAYASE<sup>†1</sup> and KATSURO INOUE<sup>†1</sup>

This paper proposes a method to generate a lexicon that helps comprehending identifier. The lexicon consists of descriptions of nouns which are used in identifiers. The description is generated from identifiers and comments in source code.

### 1. はじめに

プログラムの保守において、ソースコードの理解に掛かるコストは全体の中で大きな割合を占める<sup>1)</sup>。さらに、ソースコードの可読性及び開発者の知識等によって、理解に要するコストは大きく変化する。

一般に、ソフトウェア開発者はソースコードからソフトウェアの動作を理解する際に、識別子名から関数や変数の役割や振舞いを類推する。しかし、識別子に適切な命名がなされていない場合にはうまく類推が行えず、理解に要する労力が増えてしまう。また、識別子の命名規則や識別子中に現れる単語の用法は自然言語と異なる場合があるために、識別子に適切な命名を行うことは容易でない。そして、類推や適切な命名のために、開発者には多くの経験や開発するソフトウェア分野に対する知識が必要である。

その問題を解決するため、我々は、ソフトウェアに用いられている単語の種々の情報を収録した辞書を作成することで、ソフトウェアの理解を支援することを目標としている。

本論文は、識別子からの読解を支援するために、識別子に使用される名詞の説明文を、ソースコード中のコメントを利用することで、自動的に作成する手法を提案する。

名詞には変数の型、データの構造や関数の操作の対象などの情報が含まれている。多くの識別子名には名

詞が含まれているため、作成する辞書を利用することで類推を支援できる場面は多い。また、プログラム解析ツールの出力には多くの識別子が含まれるため、提案する辞書はプログラム解析技術の普及にも役立つと期待される。

### 2. 提案手法

提案手法の概要を図1に示す。提案手法では、Java ソースコード集合を入力とし、識別子に用いられる名詞の説明を収録した辞書を出力する。本手法では、各名詞に対して、ソースコード中にあるクラスや変数の説明として記述されているコメントから、名詞を説明する文を抽出することにより、説明を作成する。以降、各処理について説明する。

初めに、Java ソースコード集合から識別子とそれを説明するコメントのペアを取得する。また、ドキュメンテーションコメントを利用することで、識別子とコメントの対応付けを行う。

次に、識別子名を分解し名詞を抽出し、名詞とコメント集合のペアを作成する。まず、識別子名をキャメルケースやスネークケースを利用し、複数の単語に分解する<sup>2)</sup>。そして、一定回数以上ソースコード集合中に出現した名詞についてのみコメントを収集する。それから、識別子に対応するコメントを、識別子に含まれる名詞のコメント集合に追加する。

最後に、自動要約技術を利用し、説明文を作成する。コメント集合を一つの文章とみなし、その文章に対する要約を作成し、その要約結果を名詞の説明文として

<sup>†1</sup> 大阪大学  
Osaka University

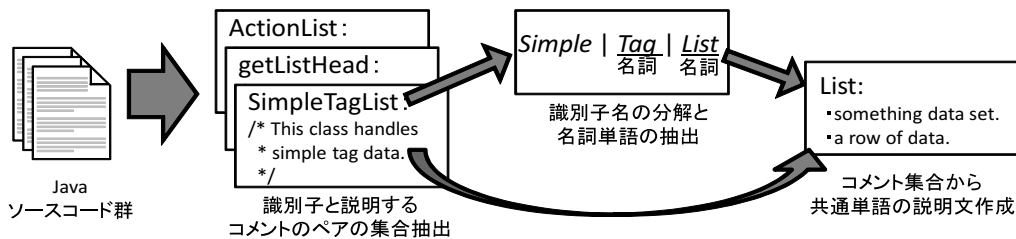


図 1 提案手法の概要

```

Copernic Summarizer による要約
Buffer:
A class for reading arbitrary numbers of bits from a
byte array.
StringBuffer is a variable size contiguous indexable
array of characters.
Tree:
Displays a tree showing all the elements in a text Doc-
ument.
Selecting a node will result in resetting the selection of
the JTextComponent.

```

図 2 実験結果の例

かれていることは少ないため、本研究の目的を達成するためには文を加工する必要がある。また、コメント中には名詞と関係の無い記述 (例: デバッグ用情報など) が含まれており、これらの記述が要約結果に影響を与えていることも分かった。さらに、tree に関係するコメントには、名詞の役割を説明する文が一切含まれていなかった。この原因は、コメントを記述した開発者が、名詞の意味を常識と考えたために説明を省略したことであると考えられる。

選択する。

### 3. 予備実験

本研究に適した説明文作成手法を調べるために予備実験を行った。実験では識別子に頻繁に用いられる 5 つの名詞 ( tree,node,page,buffer,table ) を説明する文を作成し、作成した説明文が名詞の説明文として適当であるかを主観的に評価した。また、コメント集合のコメントの内容を調査した。コメントの収集は、複数のソフトウェアのソースコードから手作業で行った。そして、収集したコメントは、クラス名に目的の名詞を含んでいるクラス、及びクラス名の末尾が目的の名詞であるクラスに対するコメントである。名詞の説明文作成には、商用の自動要約ツール Copernic Summarizer<sup>\*1</sup> と Intellexer Summarizer<sup>\*2</sup>を用いた。

実験結果の一部を図 2 に示す。

5 つの名詞のうち buffer に関しては、buffer の役割を連想できる文が得られたと考える。また、node,page,table に関しては、名詞に関連する文が得られた。しかし、tree に対する要約では、特定のクラスの振舞いを記述した文が出力されてしまった。

この実験から得られた知見を以下に記す。今回使用した 2 つの自動要約ツールは、入力の記事に含まれる文を無加工で抜き出し、要約文として出力する。しかし、コメント中に名詞の説明として適切な文が直接書

### 4. 今後の課題

実験の結果を踏まえて、名詞の説明としてより適当な説明文を作成することを目指す。

説明文の作成方法によって、得られる説明文は大きく変化する。そのため、実験を繰り返し行うことで、説明文作成に適したアルゴリズムを決定する必要がある。説明文の作成方法として、コメント集合中に複数回出現するフレーズを抽出し、組み合わせることで説明文を作成する方法を考えている。

次に、要約に与える入力を改善する。具体的には、単語の説明する文を多く、説明と無関係な文を少なくする。目的の名詞を含む識別子を説明するコメントだけでなく、親・子クラスのコメントや、識別子名に名詞が含まれていなくてもコメント中に名詞が出現するコメントなどをコメント集合に含める。また、コメントから、タイムスタンプ、デバッグ用情報、開発環境による自動挿入文などを削除する。

謝辞 本研究は科研費 ( 21700031 ) の助成を受けた。

### 参考文献

- 1) LaToza, T.D., Garlan, D., Herbsleb, J.D. and Myers, B.A.: Program comprehension as factfinding, ESEC-FSE '07, New York, NY, USA, ACM, pp.361-370 (2007).
- 2) Dave, Binkley., Marcia, ADAavis., Dawn, Lawire., and Chjristopher, Morrell. : To CamelCase or Under score, 2009 IEEE 17th ICPC, pp.158-167 (2009).

\*1 Copernic Summarizer: <http://www.copernic.com>

\*2 Intellexer Summarizer: <http://summarizer.intellexer.com>