E-009

Webにおけるサンプルコード検索支援手法の検討
A Study of Sample Code Retrieval Method on Web

飯田　修平†　　平川　豊†

Shuhei Iida　　Yutaka Hirakawa

1. はじめに

現在、インターネットの普及により Web 上にはプログラミング言語に関するサンプルコードが数多く存在している。これら Web 上のサンプルコードにはプログラミング言語の文法に関するものをはじめ、オープンソースとして提供されているパッケージに関するものなど様々な種類のものがありとても便利である。しかし現在の検索エンジンはキーワード検索を行っているのでサンプルコードが欲しいのにパッケージやクラスの解説ページが出てくるという問題や、サンプルコードが出てきたとしても同じようなサンプルコードばかり出てくるなどの問題がある。また「使用したことのないクラスの使用方法を調べる」というケースではそのクラスに属するメソッドにはどのようなものがあるのかをまず調べ、次に調べたメソッドを基にサンプルコードを検索するという手順を踏むためにサンプルコード検索に時間と手間がかかる。

そこで本研究では「使用したことのないクラスの使用方法を調べる」というケースを想定した際のサンプルコード検索支援手法の検討を行う。このケースを想定した際のサンプルコード検索の手順は①クラスに属するメソッドを Web から抽出する②抽出したメソッドを基にサンプルコードを検索するというものになる。今回は①の Web からクラスに属するメソッドを抽出するところまでを範囲とし、後述のメソッド抽出方法を用いて抽出実験を行った際の抽出結果を述べる。また、今回対象とするプログラミング言語は Java とし、メソッド名を抽出するクラスに関しては Sun の SE に含まれる標準パッケージのものを対象とする。

2. 関連研究

本研究ではメソッド抽出に正規表現を利用したパターンマッチを用いているが、パターンマッチを用いた他の研究としては[1],[2]などが挙げられる。[1]では Web コーパスを用いた人物の呼称抽出を行っている。この研究では人物の呼称を述べる際に使用される “(呼称)こと(人物名)” というパターンを用いて人物の呼称抽出を行っている。また[2]では Web を対象にオブジェクトの外観情報の抽出を行っている。この研究ではオブジェクトを構成する構成要素をまず抽出し、この構成要素を用いて “(オブジェクト)の(外観情報)(構成要素)” というパターンを用いて外観情報の抽出を行う。これらの研究との差異であるが本研究はプログラムのサンプルコードというドメインであるという点、またメソッドを抽出する際に正規表現によるパターンマッチだけでなくメソッド一覧表を用いているという点も異なっている。

3. 本研究の提案

3.1. 概要

本研究では「使用したことのないクラスの使用方法を調べる」というケースを想定した際のサンプルコード検索支援手法の検討を行う。全体の流れとしては以下のようになる。図 1 に本提案のイメージを示す。

Step1.ユーザが調べたいクラス名を入力
Step2.入力されたクラスに属するメソッドを Web から抽出
Step3.抽出したメソッド毎のサンプルコードを Web から検索し、ユーザに提示

上記のとおり本研究の最終的な目標はサンプルコードをユーザに提示することであるが、今回は step2 の「入力されたクラスに属するメソッド抽出」までを範囲としメソッド抽出の方法に焦点を当てる。また本来は Web を対象にメソッド抽出を行うべきであるが、今回は Web からいくつかのプログラミングに関するサイトを選択し、この選択したサイトを全体集合とする検索システムを構築し本研究で提案するメソッド抽出方法を検証するという方法をとった。これは提案手法を評価する際には動的な Web を用いるよりも静的な環境下で評価を行ったほうが正確な数値が取れると考えたためである。

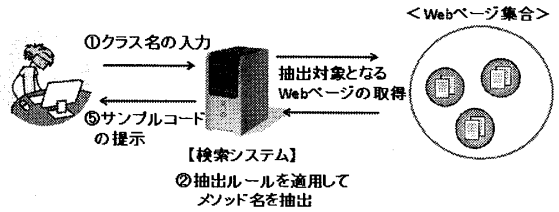

図 1. サンプルコード検索手法のイメージ

3.2. メソッド抽出方法

3.2.1.メソッド抽出へのアプローチ

あるクラスに属するメソッドを調べる方法には公開されている API のドキュメントを使用する方法が考えられるが、本研究ではドキュメントを使用したメソッド名抽出は行わない。API のドキュメントを使用すれば誤抽出も少なく正確にメソッド名を抽出することができると思われるが、使用するパッケージやクラスが変わるごとに API のドキュメントをそれぞれ用意しなければならなくなるためユーザの負担となってしまう。

そこで本研究では Web 上に存在するプログラミング関連のサイトからメソッドを抽出する手法を採用する。具体的にはメソッド名を記述する際に頻繁に利用されるパターン “（クラス名）～（メソッド名）メソッド” を用いた方法と、プログラミング関連の Web ページにはメソッド一覧表が頻出するということに着目したメソッド一覧表を用いた方法の 2 種類を用いてメソッド抽出を行う。

3.2.2 節では正規表現を利用したパターンマッチによる抽出、3.2.3 節ではメソッド一覧表を用いた抽出方法の手順を説明し 3.3 節ではこれら 2 つの抽出方法の実験結果を述べる。

†芝浦工業大学
SHIBAURA INSTITUTE OF TECHNOLOGY

3.2.2.パターンマッチによる抽出

パターンマッチによる抽出とは、メソッドが記述される際に頻繁に利用されるパターンを利用しこれを正規表現に変換してメソッドを抽出する方法である。今回採用したパターンは“（クラス名）～（メソッド名）メソッド”である。このパターンを使用することによって以下のような文からメソッド名を抽出することが可能になる。

・String クラスの equals メソッド
・String クラスに用意されている equals メソッド

この方法によるメソッド抽出の手順を以下に載せる。

Step1.パターンに対応する正規表現を生成
正規表現：“(クラス名).+?(メソッド)”
Step2.Web ページ上で上記正規表現にマッチするものをメソッド名として抽出

これらの処理を抽出対象の Web ページに対して繰り返す。

3.2.3.メソッド一覧表を用いた抽出

メソッド一覧表を用いた抽出とは、Web ページ上のメソッド一覧表からメソッド名を抽出するという方法である。この方法を用いたメソッド抽出の手順を以下に載せる。

Step1.HTML のタグを解析して一覧表を取得
Step2.取得した一覧表からメソッド名を抽出

これらの処理を抽出対象の Web ページに対して繰り返す。

3.3. 実験

3.3.1.実験概要

前節ではパターンマッチとメソッド一覧表を用いたメソッド抽出方法を述べた。本節ではこれら抽出方法の有効性を検証するため、まず Web から 8 つのサイトを選びそのサイトを構成する HTML ページ(全 130 ページ)をコピーしこれらサイトを全体集合とする検索システムを構築する。この検索システムを用いた実験の流れを以下に記す。

Step1.調べたいクラス名をユーザが入力
Step2.入力されたクラス名が出現する Web ページを検索システムにより取得
Step3.取得した Web ページから前節の抽出方法を適用してメソッド名を抽出
Step4.抽出したメソッドを正誤表と照らし合わせ評価

Step4 の正誤表に関してはあらかじめ上記 8 つのサイト全てから Web ページに出現する「クラス名」と「メソッド名」のペアを人手により作成しておく。また今回構築した検索システムには Java のオープンソース全文検索システムである Apache Lucene[3]を使用し、Web ページからメソッド一覧表を取得する際には JerichoHTML[4]という Java の HTML パーサを用いた。

3.3.2.実験結果と考察

パターンによる抽出とメソッド一覧表による抽出結果を表 1 に載せる。表中の「全クラス数」は 8 つのサイト中に出現するクラスの数を、「抽出クラス数」はメソッド名を 1 つでも抽出できたクラスの数を、「適合率」と「再現率」に関しては各クラスをクエリとして入力した際のメソッド抽出結果の適合率と再現率の平均の値である。

まずパターンマッチを用いた抽出に関しては、メソッド名を抽出できたクラス数が 15 と全体の半分以下のクラス数しか抽出できなかった。しかし抽出したメソッドの再現率は 100%で、この方法による抽出は抽出できる数(再現率)は少ないもののメソッドを正確に抽出できるという結果になった。

次にメソッド一覧表による抽出結果に関してであるが、全てのクラスに関して何らかのメソッドを抽出できている。再現率が 85%と比較的高いものの、適合率が 29%と低く関係のないメソッドまで抽出してしまっていることがわかる。この原因であるが 1 つの Web ページに複数の異なるクラスのメソッド一覧表がある場合、ユーザが入力したクラスとは関係のないクラスのメソッド一覧表まで抽出対象にして抽出してしまうためこのような結果になった。この方法によるメソッド抽出に関しては何らかの形で誤抽出してしまったメソッドをフィルタリングする必要がある。

最後にパターンマッチによる抽出とメソッド一覧表を用いた抽出方法を統合した結果についてであるが、適合率と再現率が 1%上昇に留まってしまった。この結果より、メソッド一覧表を用いた抽出のみでパターンマッチを利用した抽出結果をほぼ出せるということが言える。

表 1. メソッド抽出結果

抽出方法	全クラス数	抽出クラス数	適合率	再現率
パターンマッチ	36	15	1	0.47
一覧表		36	0.29	0.85
両者を統合		36	0.3	0.86

4. まとめと今後の課題

本稿では Web 上のプログラムサンプルコード検索支援手法の前段階であるメソッド抽出の手法とその結果について述べた。メソッド抽出の方法として 3.2.2 節のパターンマッチを用いた手法と 3.2.3 節のメソッド一覧表による 2 つの抽出方法を用いたが、実験結果よりパターンマッチによる抽出結果はメソッド一覧表による手法のみでほぼ出せるということがわかった。今後はメソッド一覧表を用いた抽出方法に対してノイズをフィルタリングし適合率を上げていく方法を検討する。適合率をあげる方法の一つとして、今回の実験では Web ページ上に記載されているサンプルコードの活用は行わなかったので、このサンプルコードを利用してノイズの除去ができるのではないかと考えている。

また今回は Web を対象にした実験ではなく、Web からサンプルとなる 8 つのサイトを対象とした実験を行ったので、今後 Web を対象とした実験も行う予定である。

参考文献

[1] 外間　智子, 北川博之, “Web コーパスを用いた人物の呼称抽”, 日本データベース学会 Letters, Vol. 5, No. 2, 2006 年

[2] 服部　峻, 手塚　太郎, 田中　克己, “オブジェクトの外観情報の Web マイニング”, DEWS2007 L4-6

[3] Apache Lucene：http://lucene.apache.org/

[4] JerichoHTML：http://jericho.htmlparser.net/docs/index.html