

技術解説記事の難易度分類による検索支援

日大生産工（学部） ○大川 英樹
日大生産工 関 亜紀子

1 まえがき

近年、はてな開発者ブログ¹⁾やQiita²⁾などプログラマ向けの技術情報共有サービスが増加している。これらのサービスでは30万件以上の技術解説記事が扱われており、検索キーワードの関連順や人気順などでソートすることができる。しかし、記事のレベルは多岐に渡っており、上位に自分のレベルに合う記事が存在するとは限らない。そこで、技術文書や学術書の内容や難易度を考慮した分類・推薦手法が提案されている。西原³⁾らは、文章中の特徴語と文章集合中の出現頻度を利用して特徴語の難しさの評価値を計算し、Web上の技術解析記事の難易度を推定する手法を提案している。また、舟木⁴⁾らは学術本を手動で難易度ごとに分類し、それらを学習モデルとすることで新規の学術本の難易度を分類する手法を提案している。

本稿では、技術解説記事の検索支援として記事のレベルに応じた分類手法を検討している。Qiitaの記事を対象に、記事のレベルによる特徴語の出現パターンの違いに着目し、入門と入門以外に分類する手法を述べる。

2 検索支援手法の提案

一般に情報検索では、ユーザが複数のキーワードを組み合わせながら、専門分野や記事の内容の絞り込みをしている。例えば、「python」を検索する際、図1のように第2キーワードとして「if文」や「例外処理」などの一般的な文法用語を用いることで基礎的な記事を得られ、「機械学習」や「深層学習」などの特定分野のキーワードを用いることで専門的な記事を得られる。しかし、「if文」の検索一覧の中には、コードの高速化や複雑な条件記述などのより応用的な記事も多く含まれ、これらが上位に含まれることがある。

そこで、本研究では、検索一覧の中でもより入門者向けの記事にユーザが求める情報が存在していると仮定し、これらを検索上位にすることによる検索支援の実現を目指す。ここでは、

記事の難易度を単語の出現頻度等より推定し、難易度の低い記事を推薦することで検索支援を実現する方法を提案する。

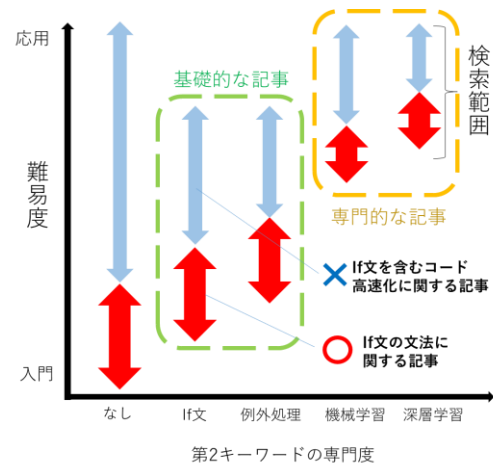


図1 第2キーワードと検索範囲の関係

3 技術解説記事に対する難易度推定

技術解説記事の難易度推定の手順を図2に沿って説明する。まず、(1)の固有名詞抽出では、キーワード検索結果の記事一覧のタイトルおよび本文を形態素解析し、固有名詞を抽出する。次に、(2)の特徴語抽出では、各固有名詞のTF-IDF値を計算し、上位10語を特徴語とする。(3)の特徴語難易度計算では、各特徴語の難易度 $Sp(t)$ ³⁾を計算する。その後、(4)の記事難易度計算では、特徴語10語の難易度の平均値を記事の難易度として算出する。

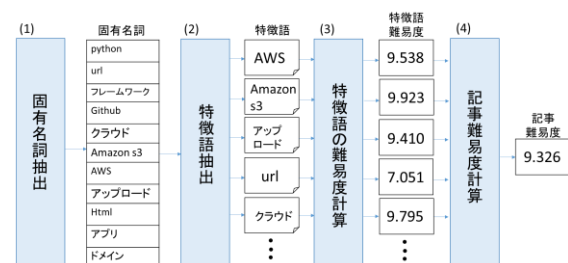


図2 単語頻度による難易度推定の流れ

A Study on Extracting Method of Technical Articles Based on Support Vector Machine

Hideki OKAWA and Akiko SEKI

表1 難易度ごとの頻出語比較

下位100記事			上位100記事		
単語	TF	難易度	単語	TF	難易度
python	0.1192	0.8074	raspberrypi	0.0238	9.8334
文字列	0.0376	8.6614	python	0.0266	0.8074
api	0.0261	8.9410	ウィジェット	0.0199	9.9645
for	0.0223	8.7496	ラズパイ	0.0182	9.9280
print	0.0173	9.4312	i	0.0135	9.5506
id	0.0129	9.4850	docker	0.0096	9.7850
str	0.0127	9.6484	レンダリング	0.0094	9.9624
c++	0.0098	9.7065	slack	0.0052	9.8688
windows	0.0092	9.2151	theta	0.0051	9.9559
opencv	0.0090	9.6753	numpy	0.0049	9.2055
django	0.0084	9.6291	api	0.0049	8.9410
if	0.0079	9.4807	pc	0.0049	9.7054
numpy	0.0069	9.2055	py	0.0047	8.6754
path	0.0069	9.5248	クエリ	0.0044	9.8936
i	0.0067	9.5506	自己関連	0.0044	9.9935
難易度平均		8.7808	難易度平均		9.0714

表1に、「python」に関して記事の難易度推定により得られた難易度下位100記事と上位100記事の頻出語上位15語のTF値と難易度を示す。網掛け部分は、両記事集合に現れる共通語のTF値の差が2倍を超える単語である。例えば、表1の「python」のTF値は下位100記事が0.1192であり、上位100記事の0.0266よりも4倍以上大きいことが分かる。また、下位100記事の場合は、「for」や「print」などの基礎的な文法に関する単語の出現頻度が高いことが分かる。一方、上位100記事では、「raspberrypi」や「ウィジェット」などの専門的な単語の出現頻度が高いことが分かる。さらに、各記事集合で現れる頻出語の難易度の平均は、下位100記事のほうが上位100記事よりも約0.29低いことから、基礎的な単語が多く解説に用いられていることが分かる。

4 線形SVMによる難易度分類

C言語やpythonなど入門者向けとされているプログラミング言語8種類を対象に、入門記事に共通してみられる文法的な特徴語の性質を利用し、入門と入門以外の二つに分類する。使用データは、表2に示すようにタイトルに「入門」と記載されている基礎的な文法や環境設定に関する解説記事421件を入門記事とした。また、各言語を対象に図2の手順により記事の難易度を計算した上位約53件を入門以外の記事とし、表3のような記事を計421件用意した。これらの20%に当たる約85記事をテスト用データとし、各記事内の固有名詞のTF-IDF値からなるベクトルを作成し、Support Vector Machine (SVM) による分類モデルを作成し分類をした。

分類結果は表4に示すように、正解率は94%、入門記事の再現率は92%、入門以外の記事の再現率は96%となっている。入門記事を誤認識した7件中2件は、Go言語特有の文法や機能を説明

表2 入門記事のタイトルと難易度の一例

タイトル	記事難易度
Python言語入門	8.5870
【Ruby】Ruby入門:オブジェクト、メソッド、クラス	9.5523
JavaScriptに再入門しようと思った関数編	9.6797
Java入門 配列と繰り返し	8.9276
【PHP超入門】HTTP(GET・POST)について	9.7480
文字列で学ぶC++入門	8.9686
Swift入門 勉強記録 その6 数値を文字列とするには	9.1958
【Go言語入門】構造体とポインタについて	9.5237

表3 入門以外のタイトルと難易度の一例

タイトル	記事難易度
【Swift】Calendarに日付操作のメソッドを追加してみた	9.9945
非プログラマー向け-個人投稿家のpythonデータ分析:2.分析用データの選択/作成編	9.9937
Yahoo! JavaScriptマップAPI [Circle]でスマホを使うとイベントが発生しない	9.9936
【Swift4】ArrayをShuffleする	9.9925
pythonでGDSファイル形式のデバイスレイアウトを描く方法	9.9925
c++ で少し便利かもしれない行列ライブラリと超複素数ライブラリを作ってみた。	9.9911
JavaScriptでMIDIファイルを解析してみる 1	9.9898
nanowalletに学ぶ、JavaScriptだけでBIP32準拠のNEMアカウントを生成する方法	9.9892

表4 線形SVMによる分類結果

正解\予測	入門	入門以外
入門	77	7
入門以外	3	82

する入門記事であった。Go言語は、全体として入門記事が20件と少ないことが原因と考えられる。また、他の2件はフレームワークに関する記事であり、本来は入門以外の記事とすべき記事を、入門記事のテストデータとしていた。

5 まとめと今後の課題

本稿では、プログラミング言語の最も基礎的な記事を対象に、特徴語の単語頻度を基に記事難易度を推定し、結果をSVMの学習データとして利用した。その結果、入門記事と入門以外の記事の分類として、9割以上の精度を確認した。

今後は、「pythonによる機械学習入門」などの専門的な記事についても入門レベルと応用レベルを分類するための拡張を検討する。また入門記事の中の更なる絞り込みを検討する。

「参考文献」

- 1) はてな開発者ブログ
<http://staff.hatenablog.com/>
- 2) Qiita <https://qiita.com>
- 3) 西原陽子, 砂山渡, 谷内田正彦, Web ページの難易度と学習順序に基づく情報理解支援システム, 電子情報通信学会論文誌, Vol. J89-D No. 9, (2006) p.1963-1975
- 4) 舟木 類佳, 黒田 久泰, 難易度及び類似度を用いたコンピューター関連書籍推薦システムの開発, 情報処理学会研究報告, 2014-NL-215 (8), (2014) p. 1-6