形態素解析で決める 2013 年の流行語

2014 年 1 月 20 日

理工学部　経営システム工学科

学籍番号：10x4025

氏名：加藤圭哲

指導教員：五島洋行　教授

論文要旨

本研究では，形態素解析と呼ばれる自然言語処理の技術を用いて技術を用いて，流行語を機械的に決定する．

現在人間が行っている流行語の決定は多数の人手が必要になり，主観が大きく関わる．しかしそれを機械的に行うことができれば，客観的に世間で流行っている言葉を知ることができるようになる．また，現在はインターネットに情報が溢れており，新聞記事やニュースデータのような文章データを複数読むためには時間が大量に必要である．そこで文章データの中で重要な役割を果たす単語を抽出できるようになれば，一目見るだけでも文章データの内容をおおよそ把握できるようになる．

本研究では流行語を調べるための元データとして，RSS と呼ばれる，ニュースタイトルや配信元の URL 等，サイトの更新情報をまとめたフォーマットで配信されているニュースデータを使用する．その中でもニュースの内容をよく表すと考えられるニュースタイトルに対して，形態素解析と呼ばれる自然言語処理の技術を用いて文章を品詞単位に分解し，出現回数や単語の重みなどを調べ，本来複数の審査委員によって決定される流行語を機械的に決定しようと試みる．なお，本研究で行う形態素解析で決める流行語では，実際に流行語を決める時と同じように 1 月から 11 月までのニュースデータの中で，形態素解析を用いて名詞と判断された単語から決定する．さらに流行語の多くは特定の期間に出現回数が増加する傾向が予測される．そこで，重要と思われる単語の中から出現回数の増加を判断し流行語を決定した結果，主観的に判断して広く大衆の目・口・耳をにぎわせた流行語として十分であろう単語を抽出することができた．

単語の重要度を判定する tfidf 法において，ニュースデータを月ごとにまとめたため，出現回数が多い単語は全ての月に出現しており，差がつかなかった．また，期間が 11 ヶ月分と少ないため，出現していない月が存在する単語に対して tfidf の値が与える影響が小さかった．加えて，今回は出現回数が極端に増加しているかどうかを，単純に tfidf の値が大きい順に 200 位までの単語しか調べなかったため，これを調べる範囲についても考える必要がある．さらに流行語の順位付けについては行っていない．最終的に抽出された 9 個の流行語は，tfidf 値の大きさと出現数がどれほど増加しているかによって決めた．そのため tfidf 値の大きさと出現数増加の大きさのどちらに優先順位を置くべきか決めかねたためである．今後は流行語の順位付けも行いたい．

目次

1. はじめに

2013 年のユーキャン新語・流行語大賞には，東進ハイスクールの CM で知られた「今でしょ!」や，国際オリンピック委員会(IOC)総会でのプレゼンで発言された「お・も・て・な・し」といった言葉が並んだ．「ユーキャン流行語大賞」のホームページに『1 年の間に発生したさまざまな「ことば」のなかで，軽妙に世相を衝いた表現とニュアンスをもって，広く大衆の目・口・耳をにぎわせた新語・流行語』とされているように，流行語とは世間で広く用いられた言葉のことであり，ユーキャン流行語は複数の人間によって，ある程度話題になった単語の中から決定されている．このように流行語の決定は人の主観による割合が大きい．

山川・馬青(2004)らは機械的に流行語を予測するために，形態素解析用の辞書に登録されていない単語であり，なおかつ，短期間に高頻度で出現した言葉や語句を「未知語」とし，その中から流行語の抽出を行った．しかし形態素解析後に現れる単語には，例えば 2012 年のユーキャン流行語である「竜巻」のように，単体でも十分に流行語としての役割を果たす単語が多く存在する．

1.1. 研究の背景

現在人間が行っている流行語の決定は多数の人手が必要になり，主観が大きく関わる．しかしそれを機械的に行うことができれば，客観的に世間で流行っている言葉を知ることができるようになると考えられる．このように流行語の予測をすることができれば，流行り始めに流行語を知ることができるようになり，例えば一般の人の興味を引くような広告をいち早く掲載できるようになる．またその他にも，現在はインターネットに情報が溢れており，新聞記事や，ニュースデータのような文章データを複数読むためには時間が大量に必要である．そこで文章データの中で重要な役割を果たす単語を抽出できるようになれば，一目見るだけでも文章データの内容をおおよそ把握できるようになり，すべて読むことなく大まかな内容を把握できるようになる．

このように文章内の重要な単語，流行語を抽出することでわずかな時間で，世間において話題のニュースを把握できるようになり，また流行をいち早く察知できるようになる．

そこで本研究では形態素解析と呼ばれる技術を用いて，いくつかのニュースサイトから配信される，ニュースデータ内の文章に対して形態素解析を行い現れる，品詞単位に分解された単語の中から流行語を抽出することを試みる．

1.2. 本研究の方針

本研究では流行語を調べるための元データとして, RSS と呼ばれる, ニュースタイトルや配信元の URL 等, サイトの更新情報をまとめたフォーマットで配信されているニュースデータを使用する. その中でもニュースの内容をよく表すと考えられるニュースタイトルに対して, 形態素解析と呼ばれる自然言語処理の技術を用いて文章を品詞単位に分解し, 出現回数や単語の重みなどを調べ, 本来複数の審査委員によって決定される流行語を機械的に決定しようと試みる. なお, 本研究で行う形態素解析で決める流行語では, 実際に流行語を決める時と同じように 1 月から 11 月までのニュースデータの中で, 形態素解析を用いて名詞と判断された単語から決定する. さらに流行語の多くは特定の期間に出現回数が増加する傾向が予測される. そこで, 重要と思われる単語の中から出現回数の増加を判断し, 流行語を決定する.

2.　関連研究

2.1.　関連技術

2.1.1　形態素解析

形態素解析とはコンピュータ等を用いた言語処理の基礎的な技術である．辞書内にある品詞についての情報を用いて，与えられた文章内にある単語を品詞単位まで分解する．このような機能を持った形態素解析ツールを用いると，例えば文章に対して以下のような判断をして解析する．

(1) 辞書に登録されている単語が存在しているか

(2) 検出された単語同士が文章になるか

「こんにちは」という短い文字列がある．(1)の作業を行う際はまず先頭の「こ」という単語が存在するかを判定する．すると「子」や「個」のような候補がでる．次に「こん」という二文字が単語として存在するかを判定する．さらに次は「こんに」を判定する．このように繰り返した後，今度は先頭の文字を除いた「んにちは」という四文字を解析の対象として同様の作業を繰り返す．このようにしてでてくる候補の単語を組み合わせると以下の図 1 のようになる．

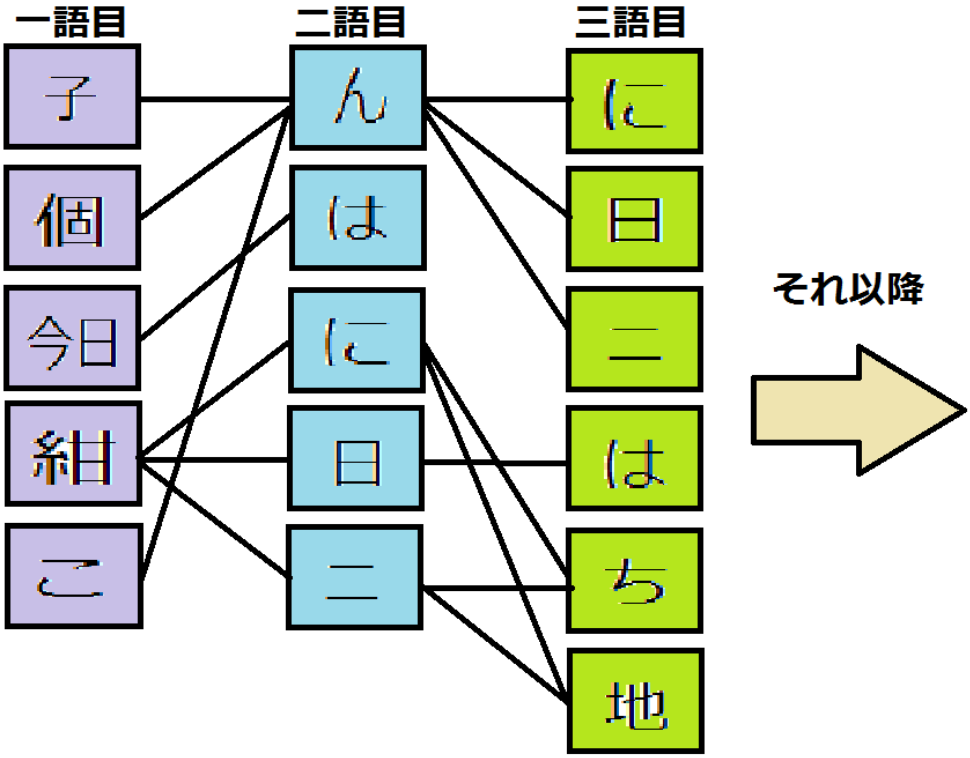


図 1　単語を切り出すときの構造

次に(2)ではｎ語目が前後の単語と接続可能であるかを判定する．辞書データにはあらかじめ単語ごとに品詞の情報があるため，それを利用する．例えば名詞+名詞では文章にならないため候補から外す等，ありえない品詞の組み合わせを削除する作業を行う．すると前後との繋がりが無い単語は削除され，組み合わせの候補が減少する．この方法以外にも組み合わせを絞る方法が研究されている．[1]そのうちのいくつかを紹介する．

2.1.2. 最長一致法

最長一致法は経験則を用いた方法で，必ずしも正しい結果を得られないが正しい結果に近くすることができる．さらに，処理が高速で使用する領域も少なくて済む．この方法では文字列を左から解析し単語を長いものから選択していく．再び「こんにちは」という文字列を例に挙げると，一語目の候補として「こ」，「個」，「子」，「今日」，「こんにちは」等があり，この中から最も長い「こんにちは」が選択される．また，最長一致法を用いる場合には辞書データ内の単語が長い順に並べられており，辞書データを検索する際の処理が短くなるようになっているため，処理が高速で使用する領域も少ない．[2]

2.1.3. 単語数最少法

もう一つの方法に単語数最少法がある．単語数最少法は全ての分割パターンの中から単語数が最も少なくなるような結果を求める．三度「こんにちは」という文字列を例に分割の候補をいくつか挙げる(表 1).

表１　単語数最少法を用いた単語分割

分割候補	単語数
今日/は	2
紺/日/は	3
紺/に/地/は	4
こんにちは	1

分割候補として単語数が最少になる「こんにちは」が選択される．[2]

2.1.4.　日本語解析 API

単語の切り出しにはヤフーデベロッパーネットワークで提供されている日本語解析 API を用いた．日本語解析 API では，日本語文を形態素に分割し，品詞，読み仮名の付与，統計情報を取得できる機能が提供される．例えば「家から近い犬公園で犬の散歩をしました．」という文章を，サンプルのコードを用いて解析した結果が図 2 である．

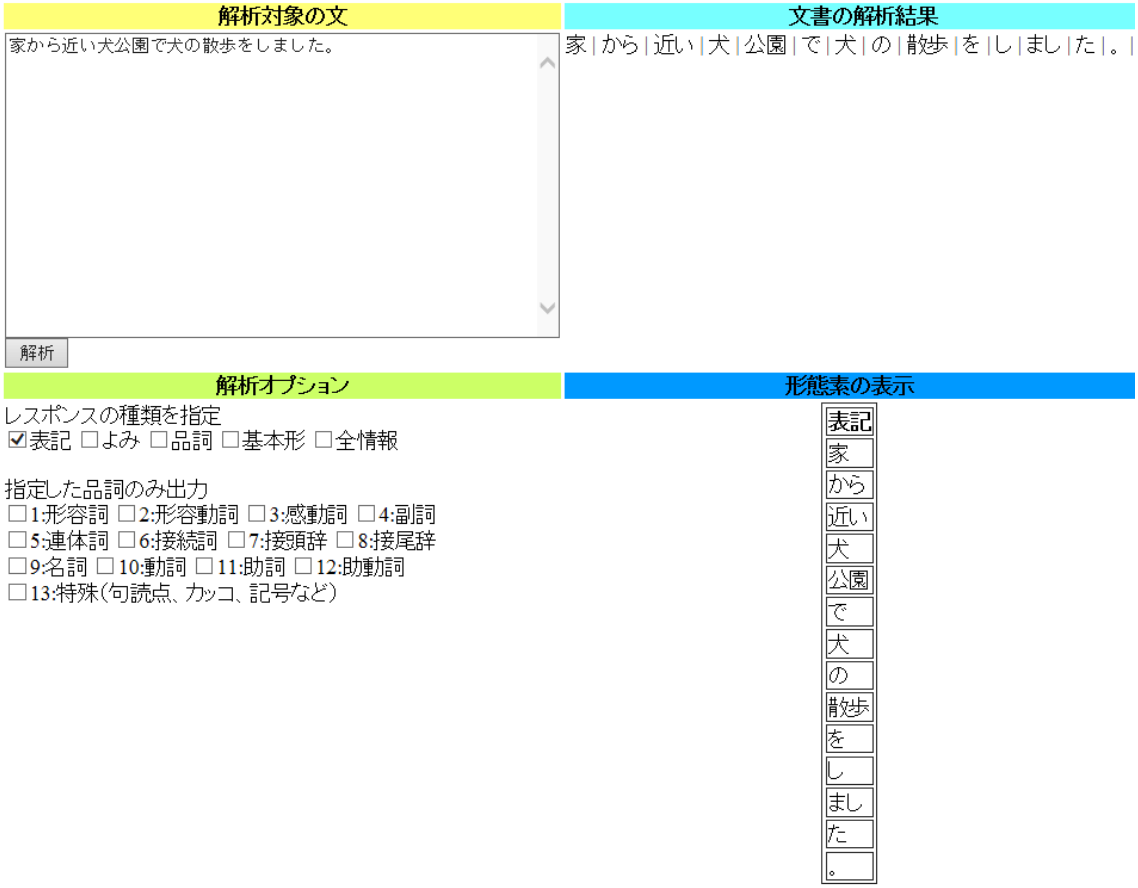


図 2　日本語解析 API を利用した単語の切り出し

日本語形態素解析 Web API は，24 時間以内で 1 つのアプリケーション ID につき 50000 件のリクエストが上限となっており，1 リクエストの最大サイズを 100KB に制限されている．このような制限の為か，一度に大量の文章を解析することはできないが，この機能を用いた単語の出現数をカウントするサイトがあるので紹介しておく．

ワード	品詞	出現回数
も	助詞	2
うち	名詞	1
すももも	名詞	1
の	助詞	1
もも	名詞	1

図 3　ヤフー形態素解析 API と単語出現数カウントプログラムを用いたサイト [3]

2.1.5. MeCab

形態素解析で有名なフリーソフトの一つに MeCab がある．これを用いてもヤフー形態素解析 API のような結果を出力することができる．また，他のフリーソフトよりも解析結果を求める時間が早いとされている．実際に解析すると以下のような結果が出力される．

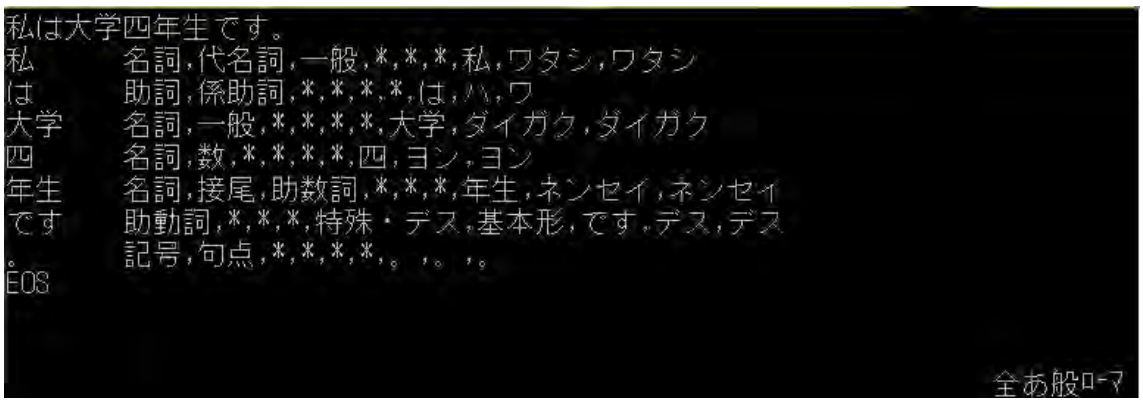


図 4　MeCab を用いた文章の解析例

2.1.6. MySQL

MySQL とはオープンソースの，リレーショナルデータベースを管理するシステムである．RDBMS(Relational DataBase Management System)と呼ばれ，リレーショナルデータベースではひとつのデータをいくつかの項目の集まりで表す．

表 2　MySQL 内のデータ例

番号	名前	性別	身長	体重
1	加藤	男	180	70
2	佐藤	女	150	40
3	藤原	男	160	60

表 2 のようにデータをデータベース内に複数，大量に管理することができる．この中から必要に応じて特定の項目だけを抽出することができ，例えば身長が 160 以下の人の名前だけを抽出する，といったようなことも可能である．

2.1.7. tf-idf 法

tf-idf 法とは tfidf を用いた単語の重みづけをするためのアルゴリズムのことである．これによってある単語の重みは以下のように表される．

$$tfidf=\mathrm{tf}\times\mathrm{idf} \qquad (a)$$

$$\mathrm{tf}=\frac{全ての文書で単語n\ の出現回数}{全ての文書中に出現した単語の総数} \qquad (b)$$

$$\mathrm{idf}=log\ \frac{総文書数}{単語n が出現する文書の数} \qquad (c)$$

tf とは単語の出現頻度のことであり，一つの文書中に多く単語 n が多く出現しているほど重要な単語であることを表している．これに対して idf では数多くの文書に出現している単語の重要度を下げる役割をしている．例えば接続詞などは文書中に多く用いられる．これらの単語を tf のみで判定すると文書にとって重要でない単語が多くなるため, idf を用いて多くの文書に出現している単語の重要度を下げる．そして tf と idf の積で単語の重みが表される．

tf-idf 法を「家から近い犬公園で犬の散歩をしました．」と「明日には家に家の置物が届きます．」，「8 時に起きた．」という三つの文章に用いてみる．まず形態素解析にかけると「家/から/近い/犬/公園/で/犬/の/散歩/を/し/まし/た/．」，「明日/に/は/家/に/置物/が/届き/ます/．」「8/時/に/起/き/た/．」というように分けられる．この中で「犬」と「家」という単語を(a)に当てはめると以下のようになる．

表 3　tfidf の計算

単語	出現数	出現した文書の数	tf	idf	tfidf
家	2	2	0.0645	0.176	0.0113
犬	2	1	0.0645	0.477	0.0153
.	3	3	1	0	0

表 3 を見ると「家」と「犬」という単語では出現数は同じであるが，出現した文書の数が違うため idf の値に差が出ている．つまり，多くの文書に出現した「犬」という単語よりも「家」という単語の方が重要でないということである．同様に全ての文書に出現している「．」は重要度が低いということがわかる．

2.2. 先行研究

形態素解析を用いた流行語の決定において山川・馬青(2004)が「ユーキャン流行語大賞」を先取りし，毎年の流行語をコンピュータで予測することは可能かといった内容の研究を行い，新聞データから流行語を抽出することに成功している．[4]

山川・馬青(2004)らは形態素解析用の辞書に登録されていない単語でありなおかつ，短期間に高頻度で出現した言葉や語句を「未知語」とし，その中から流行語の抽出を行った．このように対象を絞ることで，表 4 のような未知語を抽出している．

表 4　2003 年の「流行語」抽出結果の一部 [5]

順位	単語
1	SARS
2	NGO
3	アルカイダ
4	IT
5	ラムズフェルド

表 4 をみると，2003 年の流行語 TOP10 に入っている「SARS」のように新語かつ，流行した単語を抽出することに成功している．しかし形態素解析後に現れる単語には，単体でも十分に流行語としての役割を果たす単語が多く存在する．

そこで本研究では，形態素解析後の結果現れる細分化された品詞の中から流行語を決定する．さらに山川・馬青(2004)では単語の頻出度を調べる際，単語の順位を次のように表している．

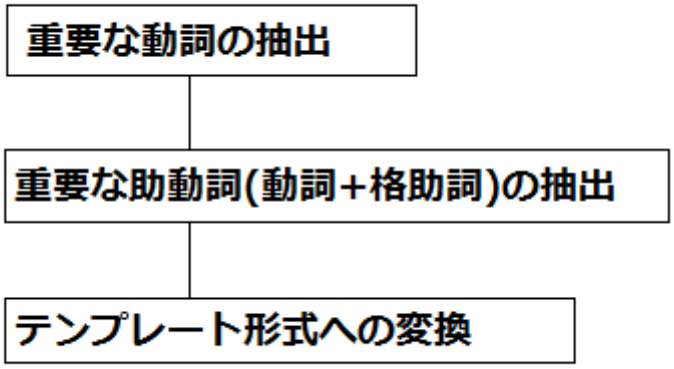
TF=総出現頻度

DF=その単語を含む総記事数

TF+2×DF=単語を順位付けする指標

このような単語の重要度を示す手法を調べているうちに，より資料の多い tf-idf 法を用いることにした．

単語の重要度の研究に関しては吉田・徳永・田中(1996)において，グループ情報を用いた重要度の計算を提案している．吉田・徳永・田中(1996)らは記事を要約するために，ある話題の記事に対して重要と考えられる項目をテンプレートとして作成する過程で，重要度を再計算することで表現の違いによる重要度の違いを少なくしており，処理手順は以下のようになっている．



重要な動詞の抽出において類似した動詞のグループ情報をもとに重要度の再計算を行っている．[6]

重要度を計算するために，記事A_i中の動詞v_k(k=1,2,…, N_{v_i})の重み w_{ik}を以下のような式で表している．[7]

$$w_{ik} = \frac{f_{ik}/(n_k + 1)}{\sqrt{\sum_{j=1}^{N_{v_i}} (f_{ij}/(n_j + 1))^2}}$$

f_{ik}: 単語T_kの記事A_iでの出現頻度

n_k: 単語T_kがサンプル記事で出現する記事数

ある話題の記事$(A_1, \ldots, A_{N_d})$の単語T_kの重要度i_kをこの重みの総和で表している．

$$i_k = \sum_{j=1}^{i_d} w_{jk}$$

そして同じような意味を表す動詞の重要度を足し合わせてグループ化している．以下はグループ情報を用いた再計算の例である．[7]

動詞	重要度			合計
起きる	3.64	→	起きる	
起こる	2.21		起こる	7.34
引き起こす	1.07		引き起こす	
起こす	0.42		起こす	

似た意味の動詞はまとめて重要度を足し合わせることで，上記のように重要度を表すことができる．

本研究では名詞を流行語の対象として調べる．名詞として出現した単語の中にもこのように似たような意味を持つ単語が存在している．

表 5　類似している単語の出現数

単語	出現数	単語	出現数
死亡	10,496	男	3,809
死去	2,546	男性	4,132

これらの単語を見ると，動詞に限らず意味の似ている単語を足し合わせる手法は，名詞のみを対象として抽出する場合でも有効であると考えられるが，似ている単語でありながらあるひとつの単語のみが流行語として頻出していた場合，足しあわせてしまうと流行語としての役割を果たしていない単語までカウントしてしまうため，主観で判断する必要がある．

2.3.　本研究の特徴

まず 2.1.5.で述べたように，tf-idf 法は単語の重みづけをするためのアルゴリズムである．それを利用して，本研究では流行語を決定するために(b)(c)の式を次のように用いた．

$$\mathrm{tf}=\frac{\textit{全ての月で単語}n\textit{ の出現回数}}{\textit{全ての文書中に出現した単語の総数}} \qquad \text{(d)}$$

$$\mathrm{idf}=log_{10}\frac{\textit{範囲すべての月数}}{\textit{単語}n\textit{が出現する月の数}}+1 \qquad \text{(e)}$$

(e)の範囲すべての月数は，十一ヶ月分のデータを使用するため 11 となる．また，log の底は計算機での計算のしやすさを考慮して 10 とした．そして(a)の式と同様に，(d)と(e)の積の大きさを基準に流行語の選択をする．

$$W=\mathrm{tf}\times\mathrm{idf} \qquad \text{(f)}$$

次に流行語として選ぶ単語の品詞を名詞のみとした．これは形態素解析を行うと文章が品詞単位に分解されるためである．

表 6　2013 年ユーキャン新語流行語の一部

PM2.5	NISA(ニーサ)	美文字	DJポリス
母さん助けて詐欺	弾丸登山	二刀流	倍返し
ビッグデータ	SNEP（スネップ）	ダークツーリズム	ご当地電力
バカッター	さとり世代	お･も･て･な･し	コントロールされている

表 6 を見ると流行語には文章も含まれていることがわかる．例えばこの中から「さとり世代」と「コントロールされている」という二つの流行語を「僕らさとり世代はコントロールされている．」文章にして形態素解析にかけると「僕ら/さとり/世代/は/コントロール/さ/れ/て/いる/．」となり，二つの流行語は品詞分解されて表示されてしまう．

表 7　文章になっている流行語

母さん助けて詐欺	ナチスの手口に学んだら	コントロールされている
今でしょ	スポーツの底力	引いたら負け

表 8　分解された後も流行語として判断できる単語

二刀流	倍返し	富士山
竜巻	野獣	絆

形態素解析を行い，文章を品詞単位に分解するため，表 7 のような文章になっている流行語を決定するのは難しい．ここで表 8 を見ると，分解された後も流行語として判断できる単語群が並んでいる．

2.1.1.で書いたように，形態素解析を行うツールには区切られた単語ごとに品詞の情報を提供する機能を持つものがある．今回使用したヤフーデベロッパーネットワークで提供されている日本語解析 API(図 2)にも表示されているように，任意の品詞のみを指定して出力することができる．この，形態素解析の機能を用いることで(表 8)のような主に名詞で構成された単語群を選ぶ．

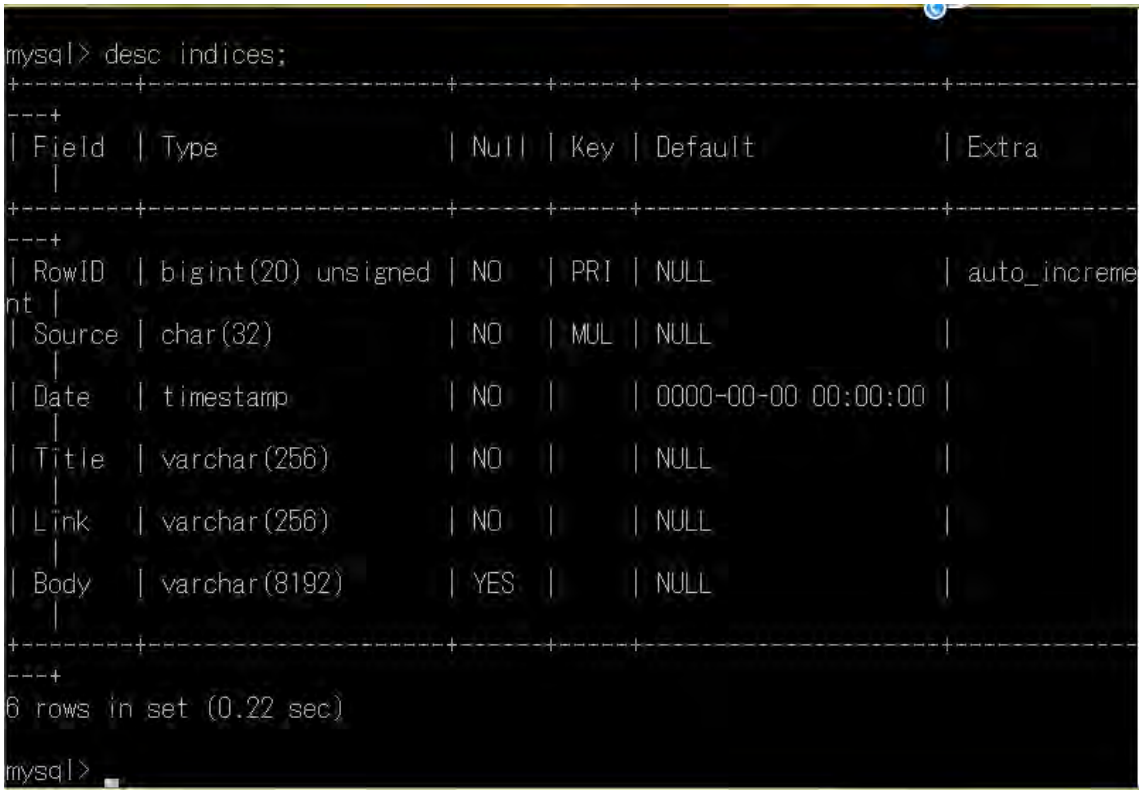
以上のことから，形態素解析で決める流行語は，形態素解析を用いて名詞と判断された単語から決定する．

3. 提案手法

ここでは, 流行語を決定するにあたり行った作業を書いていく.

3.1. 提案手法の概要

まずニュースデータは RSS というサイトの更新情報をまとめた文書フォーマットである. 今回使用した RSS では図 5 のように番号, 配信元, 配信日時, ニュースタイトル, 配信元 URL, 本文といった情報が入っている.



```
mysql> desc indices;
+--------+---------------------+------+-----+---------------------+-------------
---+
| Field  | Type                | Null | Key | Default             | Extra
   |
+--------+---------------------+------+-----+---------------------+-------------
---+
| RowID  | bigint(20) unsigned | NO   | PRI | NULL                | auto_increme
nt |
| Source | char(32)            | NO   | MUL | NULL                |
   |
| Date   | timestamp           | NO   |     | 0000-00-00 00:00:00 |
   |
| Title  | varchar(256)        | NO   |     | NULL                |
   |
| Link   | varchar(256)        | NO   |     | NULL                |
   |
| Body   | varchar(8192)       | YES  |     | NULL                |
   |
+--------+---------------------+------+-----+---------------------+-------------
---+
6 rows in set (0.22 sec)

mysql>
```

図 5　MySQL 内に取り込んだニュースデータ

このニュースデータを各ニュースサイトから定期的に受信する. ニュースサイトとサーバ, MySQL の関係は図 6 のようになっている.

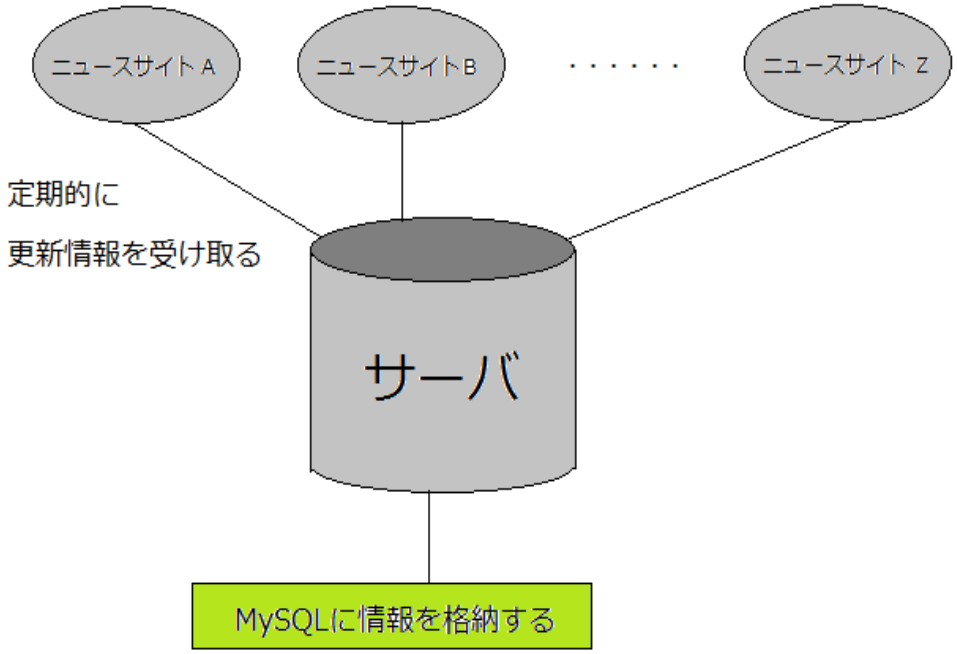


図 6　ニュースサイトとサーバ, MySQL の関係

このように MySQL 内に取り込んだニュースデータの中からニュースタイトルを抽出し，形態素解析にかけるためヤフー形態素解析 API を使用する．範囲は，実際に流行語を決める時と同じように 1 月から 11 月までの 11 ヶ月分行う．

次にニュースタイトルを品詞単位に切り分けるためにヤフーAPI を用いて形態素解析にかける．そして tf-idf 法を用いるために単語ごとの出現数や，出現している月を調べ，tf-idf 法を用いて検索する単語ごとに重みを出す．その際全ての単語について tfidf 法を行いたかったが，量が膨大であるため一部分について行う．さらにその中から数字が大きい順に流行語とする．ただし，名詞のみを抽出する．

3.2.　処理フロー

3.2.1.　全体フロー

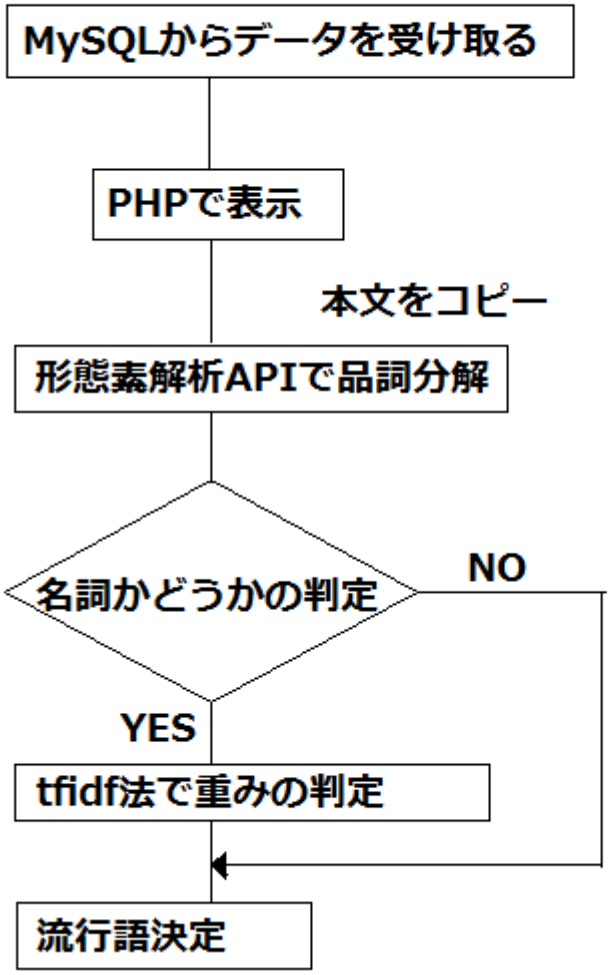


図 7　全体フロー

3.2.2.　MySQL からのデータ取得

まず MySQL に取り込んであるニュースデータからニュースデータを取得する. MySQL 上でニュースデータを全て表示しようとすると, 大量のデータであるため一部分しか表示されない. そこで全体を表示するために, PHP 上で MySQL からデータを取得することにした. そのために XAMPP をインストールし, Apache を動作させた. PHP 上で, 図 5 の RowID, Date, Title を取得し表示する.

2035909 2013-06-21 01:54:55 Gandolfini: The savior of TV drama?
2035908 2013-06-21 01:25:31 Brazil drops fare hike — but is it too late?
2035907 2013-06-21 01:38:32 Baby who fell 2 stories leaves hospital
2035906 2013-06-21 01:40:01 At their Doha HQ, Taliban make changes
2035905 2013-06-21 01:36:34 VIDEO: One-minute World News
2035903 2013-06-21 01:34:11 Teacher guilty of pupil abduction
2035904 2013-06-21 00:54:37 Musical stars to host radio station
2035902 2013-06-21 01:02:13 Brand cancels Middle East gigs
2035901 2013-06-20 18:41:21 Tributes flood in for Gandolfini
2035900 2013-06-21 01:29:58 Chinese injured in Beckham stampede
2035899 2013-06-21 01:26:07 Global markets fall on Fed comments
2035898 2013-06-21 01:27:40 Race to save India flood victims
2035897 2013-06-21 01:10:53 Taliban office row frustrates talks
2035896 2013-06-21 01:09:48 Brazil cities brace for new protests
2035895 2013-06-20 01:13:50 VIDEO: House of Commons
2035894 2013-06-21 01:10:53 Taliban office row frustrates talks
2035893 2013-06-20 23:28:52 Pub gunman given life sentence
2035892 2013-06-21 00:54:37 Musical stars to host radio station

図 8　RowID, Date, Title を PHP で表示した一部

このようにPHPを用いて表示し, MySQL上ではできなかったタイトル群をコピー&ペーストすることで, 大量の文章を図 2, 3 などの形態素解析 API を使用できるようになる.

3.2.3.　形態素解析 API を用いた品詞分解と品詞の判定

次に 3.2.2.でコピーしたデータを図 2, 3 のような形態素解析 API を用いて解析する. ただし, 2.1.2.で書いたように日本語形態素解析 Web-API は, 24 時間以内で 1 つのアプリケーション ID につき 50000 件のリクエストが上限となっており, 1 リクエストの最大サイズを 100KB に制限されているため, 大量のデータを扱う場合にはこの作業を複数回に分けて行う必要があり, 100KB では一度に約 50000 文字のみ解析が可能である. 加えて, 画面に表示する品詞を名詞に絞る作業も図 2, 名詞のチェックボックスをクリックするだけで可能である. しかし 1 リクエストの最大サイズが 100KB であるため, 作業を複数回行い, 形態素解析後の単語をまとめていくと, 同じ単語が複数存在してしまうことになるため, Microsoft Office Excel のピボットテーブル機能を用いて複数出現した単語と出現数を一つにまとめる. すると表 9 のように出現した名詞を出現数でソートしてまとめることができる.

表 9　11 ヶ月間に出現した上位 10 単語

単語	出現数	単語	出現数
in	49,248	of	19,126
to	40,263	on	16,134
日	21,947	中国	13,756
for	21,105	日本	12,642
円	20,985	万	11,066

3.2.4.　tfidf 法を用いた重みの判定

次に上位の単語について, tfidf 法を行う．ここでは表 9 の単語を参考にそれぞれの単語について重みを求める．そのために，それぞれの単語について(e), (d)を求める．
表 9 の単語について重みを求めるために必要な数字を調べたものが表 10 である．
表 10　表 9 の出現数上位 10 単語について式(e), (d), (f)を計算した結果

単語	出現数	tf	idf	tfidf
in	49,248	0.04586	0.30103	0.01381
to	40,263	0.03750	0.30103	0.01129
日	21,947	0.02044	0.30103	0.00615
for	21,105	0.01966	0.30103	0.00592
円	20,985	0.01954	0.30103	0.00588
of	19,126	0.01781	0.30103	0.00536
on	16,134	0.01503	0.30103	0.00452
中国	13,756	0.01281	0.30103	0.00386
日本	12,642	0.01177	0.30103	0.00354
万	11,066	0.01031	0.30103	0.00310

3.2.5.　流行語の選出

表 10 の計算した結果を見ると, idf の値に違いがないため出現数が多い順に tfidf の値が大きくなった．ここで上位の単語を見てみると日付に関する「in」や「to」等，流行語としてふさわしくない単語がある．そこでこのような単語を除き，残った単語の中から「日本」や「中国」等が形態素解析で決める流行語として選ばれる．

4. 分析結果

単語の出現数 50 位までは以下のようになった.

表 11　重みの大きい単語上位 50

順位	単語	出現数	tfidf*10^{-3}	順位	単語	出現数	tfidf*10^{-3}
1	in	49,248	13.81	26	女性	6441	1.81
2	to	40,263	11.29	27	new	6406	1.80
3	日	21,947	6.15	28	歳	6365	1.78
4	for	21,105	5.92	29	容疑	6245	1.75
5	円	20,985	5.88	30	億	6017	1.69
6	of	19,126	5.36	31	with	5984	1.68
7	on	16,134	4.52	32	運転	5979	1.68
8	中国	13,756	3.86	33	事故	5882	1.65
9	日本	12,642	3.54	34	女子	5847	1.64
10	万	11,066	3.10	35	police	5560	1.56
11	VIDEO	10590	2.97	36	世界	5536	1.55
12	死亡	10496	2.94	37	市場	5453	1.53
13	at	10110	2.83	38	Egypt	5192	1.46
14	首相	9508	2.67	39	選	5130	1.44
15	逮捕	9357	2.62	40	目	5126	1.44
16	US	9046	2.54	41	大統領	5002	1.40
17	Over	8604	2.41	42	市	4985	1.40
18	the	8580	2.41	43	and	4837	1.36
19	Syria	8327	2.33	44	says	4821	1.35
20	東京	7652	2.15	45	調査	4809	1.35
21	after	7544	2.11	46	from	4733	1.33
22	月	7480	2.10	47	監督	4725	1.32
23	野球	7106	1.99	48	World	4703	1.32
24	as	6811	1.91	49	talks	4693	1.32
25	ドル	6641	1.86	50	Iran	4637	1.30

日常的に出現する単語が多くの割合を占めた．ここで，重みを比較するためにいくつかの 2013 年の流行について調べる．

表 12　流行語の出現数

月	1 月	2 月	3 月	4 月	5 月	6 月	7 月	8 月	9 月	10 月	11 月
おもてなし	1	1	5	5	5	5	2	0	25	13	20
富士山	60	25	7	44	124	194	121	58	41	32	29
NISA	0	0	0	5	1	1	6	13	15	8	5

出現数 50 位までの単語と比較すると出現数が 101 と少ないことがわかる．出現回数が 0 の月を持つ「おもてなし」と「NISA」について，重みを計算する．

表 13

単語	出現数	tfidf
おもてなし	101	0.000098
NISA	54	0.000053

表 13 のように出現数 50 位までの単語と比較すると，重みが小さい値となった．出現数上位 50 の単語の多くが日常的に出現する単語であるため出現回数が多く，idf の値に差がでなかった．また，日常的に用いられる単語の出現回数が多すぎるため「おもてなし」のように idf の値に差が出た場合であっても，単純に出現回数の多い単語の重みが大きくなった．そこで，次に「富士山」について調べる．

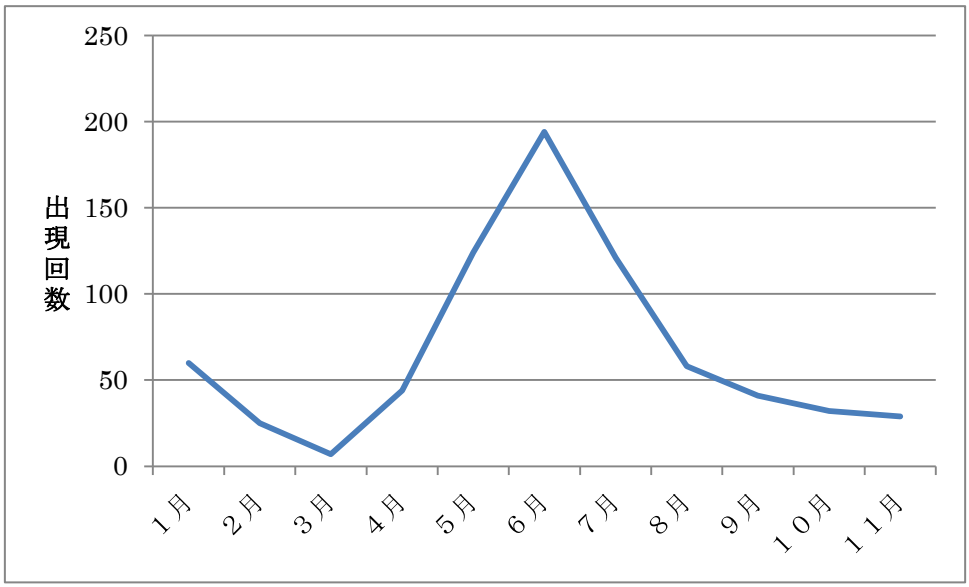


図 9　「富士山」の出現数の推移

世界文化遺産への登録がされた頃から，出現数が急激に増え6月には最も出現数が少ない月の数十倍の出現回数となっている．このように流行語は，日常的に用いられる単語とは違い，突然出現数が多くなる傾向がある．そこで tfidf の値が大きい単語上位 200 件の中から，出現回数が最も少ない月と多い月の比が大きい単語を調べると以下のような結果になった．

表 14　出現数が極端に増えている単語の出現数

単語	1 月	2 月	3 月	4 月	5 月	6 月	7 月	8 月	9 月	10 月	11 月	max/min
シリア (syria)	393	405	611	439	1,192	870	318	2,062	3,930	945	610	12.36
野球	171	123	347	693	625	653	979	1,304	755	586	520	10.60
Egypt	283	241	202	150	123	322	1,344	1,155	394	430	415	10.93
核(nuclear)	296	937	273	440	118	250	91	123	293	435	1,005	11.04
Iran	149	324	178	245	176	343	67	140	511	448	1,102	16.45
五輪	214	304	234	167	268	189	174	186	1,399	311	297	8.38
北朝鮮	265	550	455	871	400	207	209	125	159	104	134	8.38
楽天	36	74	63	113	87	109	167	264	364	494	538	14.94
汚染	47	232	70	266	67	110	135	515	640	383	104	13.62

「シリア(syria)」と「核(nuclear)」は，日本語と英語それぞれが出現数上位であったため，それらをまとめた値である．また，max/min は，すべての月の中で最も多い出現回数を最も少ない出現回数で割ったものである．出現回数に差はあるが，どの単語も今年話題に上がった単語を抽出することができた．

図 11 をみると，どの単語も出現回数が極端に増えている期間があることがわかる．その中でも「楽天」は，出現回数こそ少ないが，1 月に比べて日本一が決まる 12 月に近い11月の出現回数は約15倍にもなった．これらの単語は，日常的に用いられる単語の，月ごとの出現数の推移をみると明らかである．

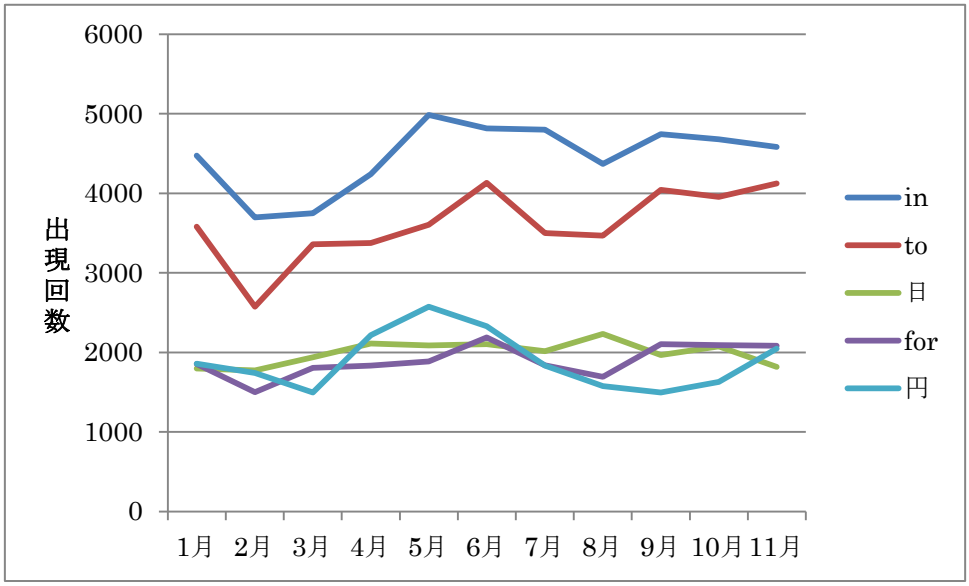


図 10　日常的に用いられる単語の出現数の推移

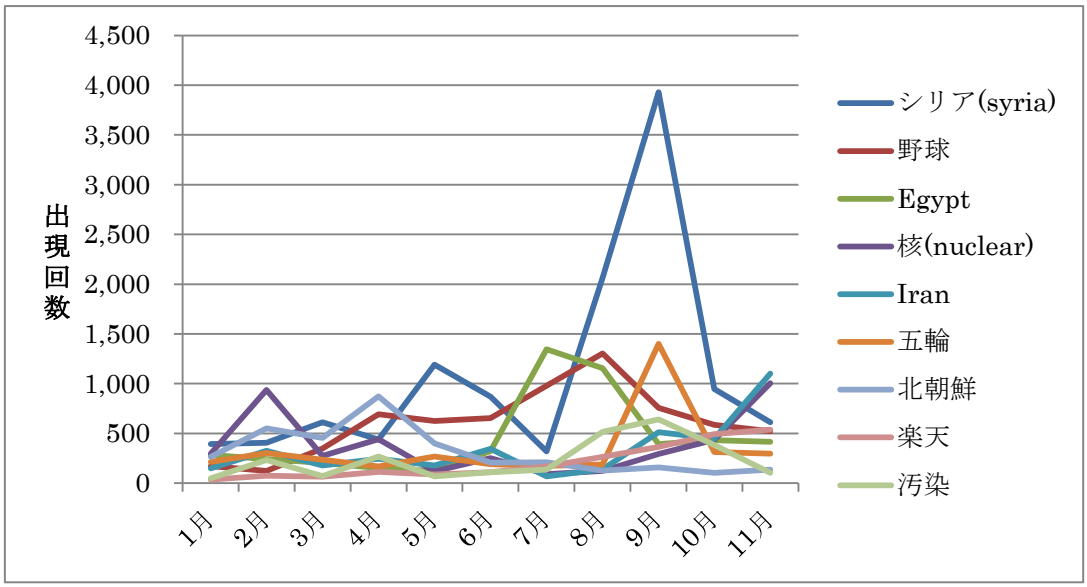


図 11　流行語として選ばれた単語の出現数の推移

ここまでの分析結果から，形態素解析で決める 2013 年の流行語と，その出現数は表 15 のようになった.

表 15　形態素解析で決める 2013 年の流行語と出現数

単語	シリア(syria)	野球	Egypt	核(nuclear)	Iran
出現数	11,775	6,756	5,059	4,261	3,683
単語	五輪	北朝鮮	楽天	汚染	
出現数	3,743	3,479	2,309	2,569	

表 15 をみると，2013 年「お・も・て・な・し」で話題になった「五輪」を始め，野球で日本一になった「楽天」や，内戦で話題になった「シリア(Syria)」などが選ばれた．しかし「核(nuclear)」や「野球」など，2013 年に限らず，何度も話題に上がる単語も同時に選ばれる結果となった．

ここで一度，山川・馬青ら(2004)で決定された流行語(表 4)と 2003 年の流行語 Top10(表 16)を比較すると SARS という単語が一致していることがわかる．

表 16　2003 年の流行語

毒まんじゅう	なんでだろう～	マニフェスト	勝ちたいんや！	コメ泥棒
SARS	年収 300 万円	バカの壁	ビフォーアフター	へぇ～

本研究によって決まった流行語は山川・馬青ら(2004)の結果に比べて，流行語そのものよりも世間で流行した話題をひとことで表すような流行語を抽出することができた．そのため，範囲をいくつかのニュースデータに絞ることで文章データの重要な内容をより把握できるようになると考えられる．

5. 結論

5.1. まとめ

2013 年の 1 月から 11 月の期間に配信されたニュースタイトルに対して，ヤフー形態素解析 API を使用して形態素解析を行い，切り出された名詞について tfidf 法を用いて重みを求めた．さらに，その中から重みの大きい上位 200 の名詞について，出現回数が突然増えているかどうかを調べると，2013 年に話題に上った単語を抽出することができた．表 16 にあがった単語のほとんどが 2013 年話題に上がった単語であり，主観的に判断して，広く大衆の目・口・耳をにぎわせた流行語として十分であろう単語を抽出することができた．

単語の特徴として，どの単語もある月に突然出現数が増えているものであり，どの単語もそれぞれの話題を語る上で欠かせない単語ばかりである．

5.2. 今後の課題

表をみるとほとんどが 2013 年話題に上がった単語であるが，「核(nuclear)」や「野球」など，2013 年に限らず，何度も話題に上がる単語も同時に選ばれる結果となった．流行語を決めるために，ある単語について一年間に限らず他の年での出現回数を調べる必要がある．また，主にニュースサイトから情報を取得しているため，出現回数上位の単語に，政治や経済に関わる単語が多くなった．

重みの大きい単語上位 50 の中に，あるニュースサイトにおいて，タイトルに必ず含まれる単語が存在しており，そのような単語が誤って流行語に選出されないようにするために，排除しておく必要がある．同じように，名詞の中でも流行語にはほぼなりえないような単語などもあらかじめ排除しておく必要がある．

tfidf の値について，ニュースタイトルを月ごとにまとめたため，出現回数が多い単語は全ての月に出現しており，差がつかなかった．また，期間が 11 ヶ月分と少ないため，出現していない月が存在する単語に対して tfidf の値が与える影響が小さかった．加えて，今回は出現回数が極端に増加しているかどうかを，単純に tfidf の値が大きい順に 200 位までの単語しか調べなかったため，これを調べる範囲についても考える必要がある．

流行語の順位付けについては行っていない．最終的に抽出された 9 個の流行語は，tfidf 値の大きさと出現数がどれほど増加しているかによって決めた．そのため tfidf 値の大きさと出現数増加の大きさのどちらに優先順位を置くべきか決めかねたためである．

5.3. おわりに

実際に流行語を決める時と同様に, 2013 年の 1 月から 11 月の期間に配信された RSS 形式のニュースデータを MySQL というリレーショナルデータベース管理システムに入れてまとめた. その中でも文章の内容を表すニュースタイトルに対して, ヤフー形態素解析 API を使用して形態素解析を行い, 切り出された品詞の中から, 名詞について tfidf 法を用いて単語の重みを求めた. さらに, その中から重みの大きい上位 200 の名詞について, 出現回数が突然増えているかどうかを調べることで, 重要であると思われる単語の中で, 日常的に用いられている単語を排除することができ, 2013 年に話題に上った単語を抽出することができた. しかし, いくつかの単語は元々出現回数に波があるものであったため, そのような流行語とは言えない単語も同時に抽出されてしまった.

参考文献

1　静岡理工科大学総合情報学部人間情報デザイン学科・知能インタラクション研究室：形態素解析の方法 <http://www.sist.ac.jp/~kanakubo/research/natural_language_processing/morphological_analysis.html>(2013/1/10 アクセス)

2　荒木健治：自然言語処理ことはじめ，森北出版株式会社(2004)

3　村上研究室：Yahoo!ディベロッパーネットワーク　日本語形態素解析 <http://murakami.media.osaka-cu.ac.jp/~mitsuhashi/02_problem1.php> (2013/1/10 アクセス)

4　山川　侑吾，馬青：新聞データからの「流行語」自動発見―「コンピュータ流行語大賞」を目指して―，（2004 年）

5　前掲 3，4 頁

6　吉田　和広，徳永　健身，田中　穂積：新聞記事要約のためのテンプレートの自動抽出，言語処理学会　第 2 回年次大会　発表論文集，（1996 年 3 月）

7　前掲 6，106 頁